Migrate Roles and Features to Windows Server 2012 R2 or Windows Server 2012

Step-by-Step

Migrate Roles and Features to Windows Server 2012 R2 or Windows Server 2012

Summary: This E-Book includes guidance to help you migrate server roles and features to Windows Server 2012 R2 or Windows Server 2012. Also included is an installation and operations guide for Windows Server Migration Tools, a set of five Windows PowerShell cmdlets that can be used to migrate some roles and features to Windows Server 2012 R2 or Windows Server 2012. This E-Book might not include the most up-to-date content about Windows Server migration, and is not guaranteed to be complete. To view the most up-to-date Windows Server migration content, see Migrate Roles and Features to Windows Server on the Microsoft TechNet website.

Category: Step-by-Step Guides

Applies to: Windows Server 2012 R2, Windows Server 2012

Source: Migrate Roles and Features to Windows Server

E-book publication date: January 2014

Contents

Migrate Roles and Features to Windows Server

Migration documentation and tools ease the process of migrating server roles, features, operating system settings, and data from an existing server that is running Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2 to a computer that is running Windows Server 2012 R2. By using migration guides linked to on this page (and where appropriate, Windows Server Migration Tools) to migrate roles, role services, and features, you can simplify deployment of new servers (including those that are running the Server Core installation option of Windows Server 2012 R2 or Windows Server 2012, and virtual servers), reduce migration downtime, increase accuracy of the migration process, and help eliminate conflicts that could otherwise occur during the migration process.

Most of the migration documentation and tools featured in this section support cross-architecture migrations (x86-based to x64-based computing platforms), migrations between physical and virtual environments, and migrations between both the full and Server Core installation options of the Windows Server operating system, where available.

In Windows Server 2012 and later releases of Windows Server, Windows Server Migration Tools supports cross-subnet migrations.

Migration guides

The following are available resources for migrating roles to Windows Server 2012 or Windows Server 2012 R2.

Windows Server roles, role services, and features

Windows Server Migration guides provide you with instructions for migrating a single role, role service, or feature to a server that is running Windows Server 2012 or Windows Server 2012 R2. Guides do not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, it is recommended that you design a custom migration procedure specific to your server environment, based on the information provided in other migration guides.

- Migrate Roles and Features to Windows Server 2012 R2
- Migrate Roles and Features to Windows Server 2012

Windows Server Migration Tools

Windows Server Migration Tools, available as a feature in Windows Server 2012 R2 and Windows Server 2012, allows an administrator to migrate some server roles, features, operating system settings, shares, and other data from computers that are running certain editions of

Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2 to computers that are running Windows Server 2012 or Windows Server 2012 R2.

Not all migrations require or use Windows Server Migration Tools. Guides for migrations that require Windows Server Migration Tools clearly state that Windows Server Migration Tools setup is part of the migration process, and provide specific instructions for how to use Windows Server Migration Tools.

To use Windows Server Migration Tools, the feature must be installed on both source and destination computers as described in the following guide.

Install, Use, and Remove Windows Server Migration Tools

See Also

Migrating Roles and Features to Windows Server

Migrate Roles and Features to Windows Server 2012 R2

Migration documentation and tools ease the process of migrating server roles, features, operating system settings, and data from an existing server that is running Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2 to a computer that is running Windows Server 2012 R2. By using migration guides linked to on this page (and where appropriate, Windows Server Migration Tools) to migrate roles, role services, and features, you can simplify deployment of new servers (including those that are running the Server Core installation option of Windows Server 2012 or Windows Server 2012 R2, and virtual servers), reduce migration downtime, increase accuracy of the migration process, and help eliminate conflicts that could otherwise occur during the migration process.

In this section

- Active Directory Certificate Services Migration Guide for Windows Server 2012 R2
- Migrating Active Directory Federation Services Role Service to Windows Server 2012 R2
- Migrate DHCP Server to Windows Server 2012 R2
- Migrate Hyper-V to Windows Server 2012 R2 from Windows Server 2012
- Migrate File and Storage Services to Windows Server 2012 R2
- File and Storage Services: Migrate an iSCSI Software Target
- Migrate Remote Desktop Services to Windows Server 2012 R2
- Migrate Cluster Roles to Windows Server 2012 R2
- Migrate Network Policy Server to Windows Server 2012 R2

See Also

Migrating Roles and Features to Windows Server

Active Directory Certificate Services Migration Guide for Windows Server 2012 R2

About this guide

This document provides guidance for migrating a certification authority (CA) to a server that is running Windows Server 2012 R2 from a server that is running Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, Windows Server 2003 R2, or Windows Server 2003.

Target audience

- Administrators or IT operations engineers responsible for planning and performing CA migration.
- Administrators or IT operations engineers responsible for the day-to-day management and troubleshooting of networks, servers, client computers, operating systems, or applications.
- IT operations managers accountable for network and server management.
- IT architects responsible for computer management and security throughout an organization.

Supported migration scenarios

This guide provides you with instructions for migrating an existing server that is running Active Directory® Certificate Services (AD CS) to a server that is running Windows Server 2008 R2 or Windows Server 2012 R2. This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, you should design a custom migration procedure that is specific to your server environment, based on the information provided in other role migration guides. To view migration guides for other server roles, see Migrate Roles and Features to Windows Server 2012 R2.

Note

This guide can be used to migrate a CA from a source server that is also a domain controller to a destination server with a different name. However, migration of a domain controller is not covered by this guide. For information about Active Directory Domain Services (AD DS) migration, see Active Directory Domain Services and DNS Server Migration Guide (http://go.microsoft.com/fwlink/?LinkId=179357).

Supported operating systems

This guide supports migrations from source servers running the operating system versions and service packs listed in the following table. All migrations described in this document assume that the destination server is running Windows Server 2012 R2 as specified in the following table.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x64-based	Windows Server 2012 R2	Windows Server 2012 R2, Server with a GUI only (not Server Core or Minimal Server Interface)	x64-based
x64-based	Windows Server 2012	Windows Server 2012 R2 or Windows Server 2012, Server with a GUI only (not Server Core or Minimal Server Interface)	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2012 R2or Windows Server 2012, Server with a GUI only (not Server Core or Minimal Server Interface) or Windows Server 2008 R2, both full and Server Core installation options	x64-based
x86-based or x64-based	Windows Server 2008	Windows Server 2012 R2or Windows Server 2012, Server with a GUI only (not Server Core or Minimal Server Interface) or Windows Server 2008 R2, both full and Server Core installation options	x64-based
x86-based or x64-based	Windows Server 2003 R2	Windows Server 2012 R2or Windows Server 2012, Server with a GUI only (not Server Core or	x64-based

Source server processor	Source server operating system	Destination server operating system	Destination server processor
		Minimal Server Interface) or Windows Server 2008 R2, both full and Server Core installation options	
x86-based or x64-based	Windows Server 2003 with Service Pack 2	Windows Server 2012 R2or Windows Server 2012, Server with a GUI only (not Server Core or Minimal Server Interface) or Windows Server 2008 R2, both full and Server Core installation options	x64-based

Note

In-place upgrades directly from Windows Server 2003 with Service Pack 2 or Windows Server 2003 R2 to Windows Server 2012 R2 are not supported. If you are running an x64-based computer, you can upgrade the CA role service from Windows Server 2003 with Service Pack 2 or Windows Server 2003 R2 to Windows Server 2008 or Windows Server 2008 R2 first and then upgrade to Windows Server 2012 R2 or Windows Server 2012.

What this guide does not provide

- Procedures to upgrade to Windows Server 2012 R2, Windows Server 2012, or Windows Server 2008 R2
- Procedures to migrate additional server roles
- Procedures to migrate additional AD CS role services

In general, migration is not required for the following AD CS role services. Instead, you can install and configure these role services on computers running Windows Server 2008 R2 or Windows Server 2012 by completing the role service installation procedures. For information about the impact of CA migration on other AD CS role services, see Impact of migration on other computers in the enterprise.

- CA Web Enrollment (http://go.microsoft.com/fwlink/?LinkId=179360)
- Online Responder (http://go.microsoft.com/fwlink/?LinkId=143098)
- Network Device Enrollment (http://go.microsoft.com/fwlink/?LinkId=179362)
- Certificate Enrollment Web Services (http://go.microsoft.com/fwlink/?LinkId=179363)

CA migration overview

Certification authority (CA) migration involves several procedures, which are covered in the following sections.

Warning

During the migration procedure, you are asked to turn off your existing CA (either the computer or at least the CA service). You are asked to name the destination CA with the same name that you used for the original CA. The computer name, (hostname or NetBIOS name), does not have to match that of the original CA. However, the destination CA name must match that of the source CA. Further, the destination CA name must not be identical to the destination computer name.

Note

It is possible to install a new PKI hierarchy while still leveraging an existing PKI hierarchy. However, doing so requires designing a new PKI, which is not covered in this guide. For an informal overview of how a dual PKI could work for an organization, see the following Ask DS blog post: Moving Your Organization from a Single Microsoft CA to a Microsoft Recommended PKI.

Preparing to migrate

- Preparing your destination server
- Backing up your source server
- Preparing your source server

Migrating the certification authority

- Backing up a CA database and private key
- Backing up CA registry settings
- Backing up CAPolicy.inf
- Removing the CA role service from the source server
- Removing the source server from the domain
- Joining the destination server to the domain
- Adding the CA role service to the destination server
- Restoring the CA database and configuration on the destination server
- Granting permissions on AIA and CDP containers
- Additional procedures for failover clustering (optional)

Verifying the migration

- Verifying certificate enrollment
- Verifying CRL publishing

Post-migration tasks

- Upgrading certificate templates in Active Directory Domain Services (AD DS)
- Retrieving certificates after a host name change
- Restoring Active Directory Certificate Services (AD CS) to the source server in the event of migration failure
- Troubleshooting migration

Impact of migration

Impact of migration on the source server

The CA migration procedures described in this guide include decommissioning the source server after migration is completed and CA functionality on the destination server has been verified. If the source server is not decommissioned, then the source server and destination server must have different names. Additional steps are required to update the CA configuration on the destination server if the name of the destination server is different from the name of the source server.

Impact of migration on other computers in the enterprise

During migration, the CA cannot issue certificates or publish CRLs.

To ensure that revocation status checking can be performed by domain members during CA migration, it is important to publish a CRL that is valid beyond the planned duration of the migration.

Because the authority identification access and CRL distribution point extensions of previously issued certificates may reference the name of the source CA, it is important to either continue to publish CA certificates and CRLs to the same location or provide a redirection solution. For an example of configuring IIS redirection, see Redirecting Web Sites in IIS 6.0.

Permissions required to complete the migration

To install an enterprise CA or a standalone CA on a domain member computer, you must be a member of the Enterprise Admins group or Domain Admins group in the domain. To install a standalone CA on a server that is not a domain member, you must be a member of the local Administrators group. Removal of the CA role service from the source server has the same group membership requirements as installation.

Estimated duration

The simplest CA migration can typically be completed within one to two hours. The actual duration of CA migration depends on the number of CAs and the sizes of CA databases.

See also

- Prepare to Migrate
- Migrating the Certification Authority
- Verifying the Certification Authority Migration
- Post-Migration Tasks
- Migrating Roles and Features in Windows Server

Prepare to Migrate

To reduce the duration of the migration process, you can complete the procedures detailed in this topic before beginning the migration process and taking the certification authority (CA) offline.

- Preparing your destination server
- Backing up your source server
- Preparing your source server

Preparing your destination server

Hardware requirements for the destination server

The hardware requirements to install any of the Active Directory Certificate Services (AD CS) role services are the same as the minimum and recommended configurations for installation of Windows Server 2012 R2. This section includes the general hardware recommendations for Windows Server 2012 R2. For detailed requirements, see System Requirements and Installation Information for Windows Server 2012 R2.

Hardware requirements for AD CS

In addition to the hardware requirements for the operating system, consider these storage and performance requirements for optimal CA performance and availability:

- The disk space requirements for a CA database depend on the number of certificates that the CA issues. Because a CA stores certificate requests, the issued certificates, and optionally, archived key material, 64 KB of database space per certificate is recommended.
- The operating system, the CA database, and the CA log files should be stored on separate physical disk drives in a multidisk configuration. For optimal CA performance and reliability, consider a redundant array of independent disks (RAID) system, such as RAID 5 for the CA database and log files and RAID 1 or RAID 0+1 for the operating system. A recommended minimum hard disk speed is 10,000 RPM.
- Processor power is generally more important to CA performance than system memory capacity.

- Failover clusters have additional hardware, software, and networking requirements. For more information, see Failover Cluster Requirements (http://go.microsoft.com/fwlink/?LinkId=179369).
- If a hardware security module (HSM) is used by the CA, consult with your HSM vendor to verify compatibility with Windows Server 2012 R2.

Software requirements for the destination server

Enterprise CAs can be installed on computers running any version of Windows Server 2012 R2.

When AD CS in Windows Server 2012 R2 is installed in an Active Directory Domain Services (AD DS) domain, the AD DS schema version must be at least 30 and all domain controllers in the domain must be running one of the following operating systems:

- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 with Service Pack 1 (SP1)
- Windows Server 2008
- Windows Server 2003 R2
- Windows Server 2003 with Service Pack 2 (SP2)
- Windows Server 2003 with SP1
- Windows Server 2003

Note

Domain controllers running Windows 2000 Server with Service Pack 4 (SP4) or Windows 2000 Server with Service Pack 3 (SP3) are technically compatible with AD CS deployments. However, the use of Windows 2000 Server is not recommended because Mainstream Support is no longer available for this operating system. For more information, see Microsoft Support Lifecycle (http://go.microsoft.com/fwlink/?LinkId=117347).

If an HSM is used by the CA, consult your HSM vendor to verify cryptographic service provider (CSP) and key service provider (KSP) compatibility with Windows Server 2012 R2 depending on the operating system to be used.

Installing the Operating System

To reduce the duration of the migration process, you can prepare the destination server by completing the following procedures before beginning the migration process and taking the source CA offline.

- Review the hardware and software requirements in the previous sections.
- Install Windows Server 2012 R2. For more information, see System Requirements and Installation Information for Windows Server 2012 R2.
- Install updates by using Windows Update.

- (Optional) Install failover clustering by reviewing the Active Directory Certificate Services (AD CS) Clustering documentation.

If you are migrating to a Server Core installation you should configure the server for remote management, which is disabled by default.

Configure remote management on Server Core

1. Log on as an administrator.
2. Type **sconfig.cmd** and press ENTER.
3. Perform the following tasks by completing the procedures described in Configuring a Server Core installation with Sconfig.cmd:
   a. Configure network settings as required for your environment.
   b. Join the server to your domain. This step is required if you are setting up an enterprise CA and optional if you are setting up a standalone CA.
   c. Configure Remote Management to enable **MMC Remote Management** or **Server Manager Remote Management**.
   d. Enable **Remote Desktop** (optional).
4. Type **13** and press ENTER to close sconfig.cmd.

Note

For information on configuring remote management in see Configure Remote Management in Server Manager.

Backing up your source server

Back up your source server to prepare for recovery of the source CA in the event of migration failure.

For information about backing up Windows Server 2012 R2 or Windows Server 2012, see Windows Server Backup.

For more information about creating backups in Windows Server 2008, see the Windows Server Backup Step-by-Step Guide for Windows Server 2008 (http://go.microsoft.com/fwlink/?LinkId=119141).

For more information about creating system state backups in Windows Server 2003, see article 326216 in the Microsoft Knowledge Base (http://go.microsoft.com/fwlink/?LinkId=117369).

Detailed procedures for backing up the source CA database, private key, and registry settings are provided in the topic Migrating the Certification Authority.

Preparing your source server

To reduce the duration and impact of CA migration, the following procedures should be completed before you begin migration:

- Back up the CA templates list (required only for enterprise CAs).

- Record the CA's CSP and signature algorithm.
- Publish a CRL with an extended validity period.

Backing up a CA templates list

An enterprise CA can have certificate templates assigned to it. You should record the assigned certificate templates before beginning the CA migration. The information is not backed up with the CA database or registry settings backup. This is because certificate templates and their association with enterprise CAs are stored in AD DS. You will need to add the same list of templates to the destination server to complete CA migration.

Note
It is important that the certificate templates assigned to the source CA are not changed after this procedure is completed.

You can determine the certificate templates assigned to a CA by using the Certification Authority snap-in or the **Certutil.exe –catemplates** command.

To record a CA templates list by using the Certification Authority snap-in

1. Log on with local administrative credentials to the CA computer.
2. Open the Certification Authority snap-in.
3. In the console tree, expand **Certification Authority**, and click **Certificate Templates**.
4. Record the list of certificate templates by taking a screen shot or by typing the list into a text file.

To record a CA templates list by using Certutil.exe

1. Log on with local administrative credentials to the CA computer.
2. Open a Command Prompt window.
3. Type the following command and press ENTER.

```
certutil.exe -catemplates > catemplates.txt
```

4. Verify that the catemplates.txt file contains the templates list.

Note
If no certificate templates are assigned to the CA, the file contains an error message: 0x80070490 (Element not found).

Recording a CA's signature algorithm and CSP

During CA installation on the destination server, you can specify the signature algorithm and CSP used by the CA, or accept the default configuration. If your source CA is not using the default configuration, then you should complete the following procedure to record the CSP and signature algorithm.

Note

If an HSM is used by the source CA, follow procedures provided by the HSM vendor to determine the HSM CSP.

To record a CA's CSP by using Certutil.exe

1. Log on with local administrative credentials to the CA computer.
2. Open a Command Prompt window.
3. Type the following command and press ENTER.

```
certutil.exe -getreg ca\csp\* > csp.txt
```

4. Verify that the csp.txt file contains the CSP details.

Publishing a CRL with an extended validity period

Before beginning CA migration, it is a good practice to publish a CRL with a validity period that extends beyond the planned migration period. The validity period of the CRL should be at least the length of time that is planned for the migration. This is necessary to enable certificate validation processes on client computers to continue during the migration period.

You should publish a CRL with an extended validity period for each CA being migrated. This procedure is particularly important in the case of a root CA because of the potentially large number of certificates that would be affected by the unavailability of a CRL.

By default, the CRL validity period is equal to the CRL publishing period plus 10 percent. After determining an appropriate CRL validity period, set the CRL publishing interval and manually publish the CRL by completing the following procedures:

Important

Record the value of the CRL publishing period before changing it. After migration is complete, the CRL publishing period should be reset to its previous value.

- Schedule the publication of the certificate revocation list
- Manually publish the certificate revocation list

Caution

Client computers download a new CRL only after the validity period of a locally cached CRL expires. Therefore, you should not use a CRL validity period that is excessively long.

Next steps

After completing the procedures to prepare the source and destination servers, you should review the topic Migrating the Certification Authority and complete the procedures appropriate for your specific migration scenario.

See also

- Active Directory Certificate Services Migration Guide for Windows Server 2012 R2
- Migrating the Certification Authority
- Verifying the Certification Authority Migration
- Post-Migration Tasks
- Migrate Roles and Features to Windows Server 2012 R2

Migrating the Certification Authority

Review all procedures in this topic and complete only the procedures that are required for your migration scenario.

- Backing up a CA database and private key
- Backing up CA registry settings
- Backing up CAPolicy.inf
- Removing the CA role service from the source server
- Removing the source server from the domain
- Joining the destination server to the domain
- Adding the CA role service to the destination server
- Restoring the CA database and configuration on the destination server
- Granting permissions on AIA and CDP containers
- Additional procedures for failover clustering

  This is an optional set of steps if you are migrating to a failover cluster.

Backing up a CA database and private key

You can back up the CA database and private key by using the Certification Authority snap-in or by using Certutil.exe at a command prompt. Complete either one of the backup procedures described in this section.

Note

If a hardware security module (HSM) is used by the CA, back up the private keys by following procedures provided by the HSM vendor.

After completing backup steps, the Active Directory Certificate Services service (Certsvc) should be stopped to prevent issuance of additional certificates. Before adding the CA role service to the destination server, the CA role service should be removed from the source server.

The backup files created during these procedures should be stored in the same location to simplify the migration. The location should be accessible from the destination server; for example, removable media or a shared folder on the destination server or another domain member.

Backing up a CA database and private key by using the Certification Authority snap-in

The following procedure describes the steps to back up the CA database and private key by using the Certification Authority snap-in while logged on to the source CA.

Note

If you prefer, you can use the certutil application to back up the CA database and private key. Using certutil for CA backup is covered in the next section.

You must use an account that is a CA administrator. On an enterprise CA, the default configuration for CA administrators includes the local Administrators group, the Enterprise Admins group, and the Domain Admins group. On a standalone CA, the default configuration for CA administrators includes the local Administrators group.

To back up a CA database and private key by using the Certification Authority snap-in

1. Choose a backup location and attach media, if necessary.
2. Log on to the source CA.
3. Open the **Certification Authority** snap-in.
4. Right-click the node with the CA name, point to **All Tasks**, and then click **Back Up CA**.
5. On the **Welcome** page of the CA Backup wizard, click **Next**.
6. On the **Items to Back Up** page, select the **Private key and CA certificate** and **Certificate database and certificate database log** check boxes, specify the backup location, and then click **Next**.
7. On the **Select a Password** page, type a password to protect the CA private key, and click **Next**.

   Security

   Use a strong password; for example, at least eight characters long with a combination of uppercase and lowercase characters, numbers, and punctuation characters.
8. On the **Completing the Backup Wizard** page, click **Finish**.
9. After the backup completes, verify the following files in the location you specified:
   - *CAName*.p12 containing the CA certificate and private key
   - Database folder containing files certbkxp.dat, edb#####.log, and *CAName*.edb
10. Open a Command Prompt window, and type **net stop certsvc** to stop the Active Directory Certificate Services service.

    Important

    The service should be stopped to prevent issuance of additional certificates. If certificates are issued by the source CA after a database backup is completed, repeat the CA database backup procedure to ensure the database backup contains all issued certificates.
11. Copy all backup files to a location that is accessible from the destination server; for

example, a network share or removable media.

Security

The private key must be protected against compromise. Protect a shared folder by limiting its access control list to authorized CA administrators. Protect removable media against unauthorized access and damage.

Backing up a CA database and private key by using Windows PowerShell

The following procedure describes the steps to back up the CA database and private key by using the Backup-CARoleService cmdlet while logged on to the source CA.

Important

You must use an account that is a CA administrator. On an enterprise CA, the default configuration for CA administrators includes the local Administrators group, the Enterprise Admins group, and the Domain Admins group. On a standalone CA, the default configuration for CA administrators includes the local Administrators group.

To back up a CA database and private key by using Windows PowerShell

1. Log on with local administrative credentials to the CA computer.
2. Right-click **Windows PowerShell** and click **Run as Administrator**.
3. Type the following command and press ENTER:

   ```
   Backup-CARoleService –path <BackupDirectory>
   ```

   Note

   BackupDirectory specifies the directory in which the backup files are created. The specified value can be a relative or absolute path. If the specified directory does not exist, it is created. The backup files are created in a subdirectory named Database.
4. The service must be stopped to prevent issuance of additional certificates. Type the following command and press ENTER:

   ```
   Stop-service certsvc
   ```
5. After the backup completes, verify the following files in the location you specified:
   - *CAName*.p12 containing the CA certificate and private key
   - Database folder containing files certbkxp.dat, edb#####.log, and *CAName*.edb
6. Copy all backup files to a location that is accessible from the destination server; for example, a network share or removable media.

   Security

   The private key must be protected against compromise. Protect a shared folder by granting permission to only authorized CA administrators. Protect removable

media against unauthorized access and damage.

Backing up a CA database and private key by using Certutil.exe

The following procedure describes the steps to back up the CA database and private key by using Certutil.exe while logged on to the source CA.

Important

You must use an account that is a CA administrator. On an enterprise CA, the default configuration for CA administrators includes the local Administrators group, the Enterprise Admins group, and the Domain Admins group. On a standalone CA, the default configuration for CA administrators includes the local Administrators group.

To back up a CA database and private key by using Certutil.exe

1. Log on with local administrative credentials to the CA computer.
2. Open a Command Prompt window.
3. Type **Certutil.exe –backupdb** <*BackupDirectory*> and press ENTER.
4. Type **Certutil.exe –backupkey** <*BackupDirectory*> and press ENTER.

   Note

   BackupDirectory specifies the directory in which the backup files are created. The specified value can be a relative or absolute path. If the specified directory does not exist, it is created. The backup files are created in a subdirectory named Database.
5. Type a password at the prompt, and press ENTER. You must retain a copy of the password to access the key during CA installation on the destination server.

   Security

   Use a strong password; for example, at least eight characters with a combination of uppercase and lowercase characters, numbers, and symbols.
6. Type **net stop certsvc** and press ENTER to stop the Active Directory Certificate Services service. The service must be stopped to prevent issuance of additional certificates.
7. After the backup completes, verify the following files in the location you specified:
   - *CAName*.p12 containing the CA certificate and private key
   - Database folder containing files certbkxp.dat, edb#####.log, and *CAName*.edb
8. Copy all backup files to a location that is accessible from the destination server; for example, a network share or removable media.

   Security

   The private key must be protected against compromise. Protect a shared folder by granting permission to only authorized CA administrators. Protect removable media against unauthorized access and damage.

Backing up CA registry settings

Complete one of the following procedures to back up the CA registry settings.

The files created during the backup procedure should be stored in the same location as the database and private key backup files to simplify the migration. The location should be accessible from the destination server; for example, removable media or a shared folder on the destination server or another domain member.

You must be logged on to the source CA using an account that is a member of the local Administrators group.

To back up CA registry settings by using Regedit.exe

1. Click **Start**, point to **Run**, and type **regedit** to open the Registry Editor.
2. In `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CertSvc`, right-click **Configuration**, and then click **Export**.
3. Specify a location and file name, and then click **Save**. This creates a registry file containing CA configuration data from the source CA.
4. Copy the registry file to a location that is accessible from the destination server; for example, a shared folder or removable media.

To back up CA registry settings by using Reg.exe

1. Open a Command Prompt window.
2. Type **reg export HKLM\SYSTEM\CurrentControlSet\Services\CertSvc\Configuration** <*output file*>.reg and press ENTER.
3. Copy the registry file to a location that is accessible from the destination server; for example, a shared folder or removable media.

Backing up CAPolicy.inf

If your source CA is using a custom CAPolicy.inf file, you should copy the file to the same location as the source CA backup files.

The CAPolicy.inf file is located in the %SystemRoot% directory, which is usually C:\Windows.

Removing the CA role service from the source server

It is important to remove the CA role service from the source server after completing backup procedures and before installing the CA role service on the destination server. Enterprise CAs and standalone CAs that are domain members store in Active Directory Domain Services (AD DS) configuration data that is associated with the common name of the CA. Removing the CA role service also removes the CA's configuration data from AD DS. Because the source CA

and destination CA share the same common name, removing the CA role service from the source server after installing the CA role service on the destination server removes configuration data that is required by destination CA and interferes with its operation.

The CA database, private key, and certificate are not removed from the source server by removing the CA role service. Therefore, reinstalling the CA role service on the source server restores the source CA if migration fails and performing a rollback is required. See Restoring Active Directory Certificate Services (AD CS) to the source server in the event of migration failure.

Warning

Although it is not recommended, some administrators may choose to leave the CA role service installed on the source server to enable the source CA to be brought online quickly in the case of migration failure. If you choose not to remove the CA role service from the source server before installing the CA role service on the destination server, it is important that you disable the Active Directory Certificate Services service (Certsvc) and shut down the source server before installing the CA role service on the destination server. Do not remove the CA role service from the source server after completing the migration to the destination server. Removing the CA role service from the source server after migrating to the destination server interferes with the operation of the destination CA.

- To remove the CA on a computer running Windows Server 2003, use the Add/Remove Windows Components wizard.
- To remove the CA on a computer running Windows Server 2008 or later, use the Remove Roles and Features Wizard in Server Manager.

Removing the source server from the domain

Because computer names must be unique within an Active Directory domain, it is necessary to remove the source server from its domain and delete the associated computer account from Active Directory before joining the destination server to the domain.

If you have access to a domain member computer running Windows Server 2008 or later , complete the following procedure to remove the source server from the domain by using Netdom.exe.

If you do not have access to a computer running Windows Server 2008 or later, then complete the procedure Join a Workgroup (http://go.microsoft.com/fwlink/?LinkId=207683). Joining a workgroup also removes a domain member computer from its domain.

To remove the source server from the domain by using Netdom.exe

1. On a domain member computer running Windows Server 2008 or later, open an elevated Command Prompt window.
2. Type **netdom remove** <*source server name*> /d:<*domain name*> /ud:<*domain user account*> /pd:* and press ENTER. For additional command-line options, see Netdom

remove syntax (http://go.microsoft.com/fwlink/?LinkID=207681).

Tip

Using Windows PowerShell, you can run the command: **remove-computer** <*computer name*>

For more information, see Remove-Computer (http://technet.microsoft.com/en-us/library/dd347703.aspx).

3. Shut down the source server.

After removing the source server from its domain, delete the source server's computer account from AD DS by completing the procedure Delete a Computer Account (http://go.microsoft.com/fwlink/?LinkID=138386).

Tip

You can also use Windows PowerShell to remove the computer account from AD DS. For more information, see Remove-ADComputer (http://technet.microsoft.com/library/hh852313).

Joining the destination server to the domain

Before joining the destination server to the domain, change the computer name to the same name as the source server. Then complete the procedure to join the destination server to the domain.

If your destination server is running on the Server Core installation option, you must use the command-line procedure.

To rename the destination server, you must be a member of the local Administrators group. To join the server to the domain, you must be a member of the Domain Admins or Enterprise Admins groups, or have delegated permissions to join the destination server to an organizational unit (OU) in the domain.

Important

If you are migrating a standalone CA that is not a domain member, complete only the steps to rename the destination server and do not join the destination server to the domain.

To join the destination server to the domain by using Netdom.exe

1. On the destination server, open an elevated Command Prompt window.
2. Type **netdom renamecomputer** <*computer name*> /newname:<*new computer name*>

   Tip

   Using Windows PowerShell, you can run the command: **rename-computer** <*new computer name*>
3. Restart the destination server.

4. After the destination server restarts, log on by using an account that has permission to join computers to the domain.
5. Open an elevated Command Prompt window, type **netdom join** <*computer name*> /d:<*domain name*> /ud:<*domain user account*> /pd:* [/ou:<*OU name*>] and press ENTER. For additional command-line options, see Netdom join syntax (http://go.microsoft.com/fwlink/?LinkID=207680).

   Tip

   Using Windows PowerShell, you can run the command: **add-computer -DomainName** <*domain name*>

   For more information, see Add-Computer (http://technet.microsoft.com/en-us/library/dd347556.aspx).
6. Restart the destination server.

Adding the CA role service to the destination server

This section describes two different procedures for adding the CA role service to the destination server, including special instructions for using failover clustering.

Review the following statements to determine which procedures to complete.

- If your destination server is running the Server Core installation option, you can use Windows PowerShell to install the CA. See Install-AdcsCertificationAuthority for more information.
- If you are migrating to a CA that uses failover clustering, you must review the section "Special instructions for migrating to a failover cluster" and complete the procedures Importing the CA certificate and Adding the CA role service by using Server Manager.
- If you are migrating to a CA that uses an HSM, you must complete the procedures Importing the CA certificate and Adding the CA role service by using Server Manager.
- If none of the above statements describes your migration scenario, you can use the following procedure to add the CA role service: Adding the CA role service by using Server Manager. If you use Server Manager, you must also complete the procedure Importing the CA certificate.

Special instructions for migrating to a failover cluster

If you are migrating to a failover cluster, the procedures to import the CA certificate and add the CA role service must be completed on each cluster node. After the CA role service is added to each node, you should stop the Active Directory Certificate Services service (Certsvc).

Additionally, it is important to ensure that the shared storage used by the CA is online and assigned to the node you are adding the CA role service to.

The CA database and log files must be located on shared storage. Specify the shared storage location during step 12 of the CA installation procedure.

To verify shared storage is online

1. Log on to the destination server.
2. Start Server Manager.
3. In the console tree, double-click **Storage**, and click **Disk Management**.
4. Ensure that the shared storage is online and assigned to the node you are logged on to.

Importing the CA certificate

If you are adding the CA role service by using Server Manager, you must complete the following procedure to import the CA certificate.

To import the CA certificate

1. Start the Certificates snap-in for the local computer account.
2. In the console tree, double-click **Certificates (Local Computer)**, and click **Personal**.
3. On the **Action** menu, click **All Tasks**, and then click **Import** to open the Certificate Import Wizard. Click **Next**.
4. Locate the <*CAName*>.p12 file created by the CA certificate and private key backup on the source CA, and click **Open**.
5. Type the password, and click **OK**.
6. Click **Place all certificates in the following store**.
7. Verify **Personal** is displayed in **Certificate store**. If it is not, click **Browse**, click **Personal**, click **OK**.

   Note
   If you are using a network HSM, complete steps 8 through 10 to repair the association between the imported CA certificate and the private key that is stored in the HSM. Otherwise, click **Finish** to complete the wizard and click **OK** to confirm that the certificate was imported successfully.
8. In the console tree, double-click **Personal Certificates**, and click the imported CA certificate.
9. On the **Action** menu, click **Open**. Click the **Details** tab, copy the serial number to the Clipboard, and then click **OK**.
10. Open a Command Prompt window, type **certutil –repairstore My "{Serialnumber}"** and then press ENTER.

Adding the CA role service by using Server Manager

If your destination server is a domain member, you must use an account that is a member of the Domain Admins or Enterprise Admins group in order for the installation wizard to access objects in AD DS.

Important

If you made a backup CAPolicy.inf file from the source CA, review the settings and make adjustments, if necessary. Copy the CAPolicy.inf file to the %windir% folder (C:\Windows by default) of the destination CA before adding the CA role service.

To add the CA role service by using Server Manager

1. Log on to the destination server, and start Server Manager.
2. In the console tree, click **Roles**.
3. On the **Action** menu, click **Add Roles**.
4. If the **Before you Begin** page appears, click **Next**.
5. On the **Select Server Roles** page, select the **Active Directory Certificate Services** check box, and click **Next**.
6. On the **Introduction to AD CS** page, click **Next**.
7. On the **Role Services** page, click the **Certification Authority** check box, and click **Next**.

   Note
   If you plan to install other role services on the destination server, you should complete the CA installation first, and then install other role services separately. Installation procedures for other AD CS role services are not described in this guide.
8. On the **Specify Setup Type** page, specify either **Enterprise** or **Standalone**, to match the source CA, and click **Next**.
9. On the **Specify CA Type** page, specify either **Root CA** or **Subordinate CA**, to match the source CA, and click **Next**.
10. On the **Set Up Private Key** page, select **Use existing private key** and **Select a certificate and use its associated private key**.

    Note
    If an HSM is used by the CA, select the private key by following procedures provided by the HSM vendor.
11. In the **Certificates** list, click the imported CA certificate, and then click **Next**.

    Note
    If you are using a custom CSP that requires strong private key protection, click **Allow administrator interaction when the private key is accessed by the CA**. The CSPs included with Windows Server do not require this setting to be enabled.
12. On the **CA Database** page, specify the locations for the CA database and log files.

    Note
    If you are migrating the CA to a failover cluster, the specified locations for database and log files must be on shared storage that is attached to all nodes. Because the location is common to cluster nodes, click **Yes** to overwrite the

existing CA database as you add the CA role service to other nodes.

Important
If you specify locations that are different from the locations used on the source CA, then you must also edit the registry settings backup file before the CA is restored. If the locations specified during setup are different from the locations specified in the registry settings, the CA cannot start.

13. On the **Confirmation** page, review the messages, and then click **Configure**.
14. If you are migrating to a failover cluster, stop the Active Directory Certificate Services service (Certsvc) and HSM service if your CA uses an HSM. Then repeat the procedures to import the CA certificate and add the CA role service on other cluster nodes.

Adding the CA role service by using Windows PowerShell

Use the following procedure to add the CA role service by using the Install-ADCSCertificateAuthority cmdlet with the ExistingCertificateParameterSet:

```
Install-AdcsCertificationAuthority [-AllowAdministratorInteraction [<SwitchParameter>]]
[-CAType <CAType>]
[-CertFile <String>] [-CertFilePassword <SecureString>] [-CertificateID <String>] [-
Credential <PSCredential>]
[-DatabaseDirectory <String>] [-Force [<SwitchParameter>]] [-LogDirectory <String>] [-
OverwriteExistingDatabase
[<SwitchParameter>]] [-OverwriteExistingKey [<SwitchParameter>]] [-Confirm
[<SwitchParameter>]] [-WhatIf
[<SwitchParameter>]] [<CommonParameters>]
```

The ExistingCertificateParameterSet is the preferred for migration because you can use the `-CertificateID` parameter to identify the CA certificate from the Importing the CA certificate section in order to configure for the CA. The value for `-CertificateID` can be either the thumbprint or the serial number of the imported certificate.

To add the CA role service by using Windows PowerShell

1. Right-click **Windows PowerShell** and click **Run as Administrator**.
2. To install the CA role service binaries with the Certification Authority and Certificate Templates MMC tools, type the following command and press ENTER:

```
Add-WindowsFeature ADCS-Cert-Authority -
IncludeManagementTools
```

3. Type the `Install-AdcsCertificationAuthority` cmdlet with the appropriate parameters and press ENTER. For example, to restore an Enterprise Subordinate CA by using the

certificate you imported in the Importing the CA certificate section, type the following command and press ENTER:

```
Install-AdcsCertificationAuthority –CAType
EnterpriseSubordinateCA –CertificateID "YourCertSerialNumber
or YourCertThumbprint" –credential (get-credential
domain\administrator)
```

Type the password for a member of the Enterprise Admins group or Domain Admins group as needed.

For more information about using Windows PowerShell to install other AD CS role services, see Deploying AD CS Using Windows PowerShell. For more general information about Windows PowerShell cmdlets for AD CS, see AD CS Deployment Cmdlets in Windows PowerShell.

Restoring the CA database and configuration on the destination server

The procedures in this section should be completed only after the CA role service has been installed on the destination server.

If you are migrating to a failover cluster, add the CA role service to all cluster nodes before restoring the CA database. The CA database should be restored on only one cluster node and must be located on shared storage.

Restoring the source CA backup includes the following tasks:

- Restoring the source CA database on the destination server
- Restoring the source CA registry settings on the destination server
- Verifying certificate extensions on the destination CA
- Restoring the certificate templates list (required only for enterprise CAs)

Restoring the source CA database on the destination server

This section describes two different procedures for restoring the source CA database backup on the destination server.

If you are migrating to a Server Core installation, you must use the procedure "To restore the CA database by using Certutil.exe" or "To restore the CA database by using Windows PowerShell." In general, it is possible to remotely manage a CA running on a Server Core installation by using the Certification Authority snap-in and Server Manager; however, it is only possible to restore a CA database remotely by using Windows PowerShell.

If you are migrating to a failover cluster, ensure that shared storage is online and restore the CA database on only one cluster node.

To restore the CA database by using the Certification Authority snap-in

1. Log on to the destination server by using an account that is a CA administrator.
2. Start the Certification Authority snap-in.
3. Right-click the node with the CA name, point to **All Tasks**, and then click **Restore CA**.
4. On the **Welcome** page, click **Next**.
5. On the **Items to Restore** page, select **Certificate database and certificate database log**.
6. Click **Browse**. Navigate to the parent folder that holds the **Database** folder (the folder that contains the CA database files created during the CA database backup).

   Caution
   Do not select the **Database** folder. Select its parent folder.
7. Click **Next** and then click **Finish**.
8. Click **Yes** to start the CA service (certsvc).

To restore only the CA database by using Windows PowerShell

1. Log on to the destination server by using an account that is a CA administrator.
2. Right-click **Windows PowerShell** and click **Run as Administrator**.
3. Type the following command to stop the CA service and press ENTER:

   ```
   Stop-service certsvc
   ```

4. Type the following command and press ENTER:

   ```
   Restore-CARoleService –path < CA Database Backup Directory> -
   DatabaseOnly - Force
   ```

   Note
   The value of <*CA Database Backup Directory*> is the parent directory of the **Database** directory. For example, if the CA database backup files are located in C:\Temp\Database, then the value of <*CA Database Backup Directory*> is C:\Temp. Include the force flag because an empty CA database will already be present after you perform the steps in Adding the CA role service by using Server Manager.
5. Type the following command to restart the CA service and press ENTER:

   ```
   Start-service certsvc
   ```

To restore the only CA database by using Certutil.exe

1. Log on to the destination server by using an account that is a CA administrator.
2. Open a Command Prompt window.

3. Type the following command to stop the CA service and press ENTER:

```
Net stop certsvc
```

4. Type **certutil.exe -f -restoredb** *<CA Database Backup Directory>* and press ENTER.

   Note
   The value of *<CA Database Backup Directory>* is the parent directory of the **Database** directory. For example, if the CA database backup files are located in C:\Temp\Database, then the value of *<CA Database Backup Directory>* is C:\Temp.

5. Type the following command to restart the CA service and press ENTER:

```
Net start certsvc
```

Restoring the source CA registry settings on the destination server

The CA configuration information is stored in the registry in:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CertSvc

Before importing the registry settings from the source CA to the target CA, create a backup of the default target CA registry configuration by using the procedure Backing up CA registry settings. Be sure to perform these steps on the target CA and to name the registry file a name such as "DefaultRegCfgBackup.reg" to avoid confusion.

Important
Some registry parameters should be migrated without changes from the source CA computer, and some should not be migrated. If they are migrated, they should be updated in the target system after migration because some values are associated with the CA itself, whereas others are associated with the domain environment, the physical host, the Windows version, or other factors that may be different in the target system.

A suggested way of performing the registry configuration import is first to open the registry file you exported from the source CA in a text editor and analyze it for settings that may need to be changed or removed. The following table shows the configuration parameters that should be transferred from the source CA to the target CA.

Registry location	Configuration parameter
HKEY_LOCAL_MACHINE\system\currentcontrolset\services\certsvc\Configuration	LDAPFlags
HKEY_LOCAL_MACHINE\system\currentcontrolset\services\certsvc\Configuration*CAname*	DSConfigDN

Registry location	Configuration parameter
	ForceTeletex CRLEditFlags CRLFlags InterfaceFlags (required only if has been changed manually) EnforceX500NameLengths SubjectTemplate ValidityPeriod ValidityPeriodUnits KRACertHash KRACertCount KRAFlags CRLPublicationURLs CRLPeriod CRLPeriodUnits CRLOverlapPeriod CRLOverlapUnits CRLDeltaPeriod CRLDeltaPeriodUnits CRLDeltaOverlapPeriod CRLDeltaOverlapUnits CACertPublicationURLs (check for custom entries with hard-coded host names or other data specific

Registry location	Configuration parameter
	to the source CA) CACertHash
HKEY_LOCAL_MACHINE\system\currentcontrolset\services\certsvc\Configuration*CAname*\ExitModules\CertificateAuthority_MicrosoftDefault.Exit	PublishCertFlags
HKEY_LOCAL_MACHINE\system\currentcontrolset\services\certsvc\Configuration*CAname*\PolicyModules\CertificateAuthority_MicrosoftDefault.Policy	EnableRequestExtensionList EnableEnrolleeRequestExtensionList DisableExtensionList SubjectAltName SubjectAltName2 RequestDisposition EditFlags

To analyze the registry file

1. Right-click the .reg text file created by exporting the settings from the source CA.
2. Click **Edit** to open the file in a text editor.
3. If the target CA's computer name is different from the source CA's computer name, search the file for the host name of the source CA computer. For each instance of the host name found, ensure that it is the appropriate value for the target environment. Change the host name, if necessary. Update the **CAServerName** value.

   Important
   If the host name is located in the .reg file as part of the CA name, such as in the **Active** value within the **Configuration** key or the **CommonName** value within the **CAName** key, do not change the setting. The CA name must not be changed as part of the migration. This means the new target CA must have the old CA's name, even if part of that name is the old CA's host name.
4. Check any registry values that indicate local file paths, such as the following, to ensure drive letter names and paths are correct for the target CA. If there is a mismatch between the source and the target CA, either update the values in the file or remove them from the file so that the default settings are preserved on the target CA.

   These storage location settings are elected during CA setup. They exist under the Configuration registry key:

- DBDirectory
- DBLogDirectory
- DBSystemDirectory
- DBTempDirectory

The following settings under the Configuration\{*CA Name*} registry key contain, in their default values, a local path. (Alternatively, you can update these values after importing them by using the Certification Authority snap-in. The values are located on the CA properties **Extensions** tab.)

- CACertPublicationURLs
- CRLPublicationURLs

Warning

Some registry values are associated with the CA, while others are associated with the domain environment, the physical host computer, the Windows version, or even other role services. Consequently, some registry parameters should be migrated without changes from the source CA computer and others should not. Any value that is not listed in the .reg text file that is restored on the target CA retains its existing setting or default value. An issue that can occur, if the registry values are not properly verified, is explained in the following TechNet Wiki article: AD: Certification Authority Web Enrollment Configuration Failed 0x80070057 (WIN32: 87).

Remove any registry values that you do not want to import into the target CA. Once the .reg text file is edited, it can be imported into the target CA. By importing the source server registry settings backup into the destination server, the source CA configuration is migrated to the destination server.

To import the source CA registry backup on the destination CA

1. Log on to the destination server as a member of the local Administrators group.
2. Open a Command Prompt window.
3. Type **net stop certsvc** and press ENTER.
4. Type **reg import** *<Registry Settings Backup.reg>* and press ENTER.

To edit the CA registry settings

1. Click **Start**, type **regedit.exe** in the **Search programs and files** box, and press ENTER to open the Registry Editor.
2. In the console tree, locate the key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\CertSvc\Configuration`, and click **Configuration**.
3. In the details pane, double-click **DBSessionCount**.
4. Click **Hexadecimal**. In **Value data**, type **64**, and then click **OK**.
5. Verify the locations specified in the following settings are correct for your destination server, and change them as needed to indicate the location of the CA database and log

files.

- DBDirectory
- DBLogDirectory
- DBSystemDirectory
- DBTempDirectory

Important
Complete steps 6 through 8 only if the name of your destination server is different from the name of your source server.

6. In the console tree of the registry editor, expand **Configuration**, and click your CA name.
7. Modify the values of the following registry settings by replacing the source server name with the destination server name.

Note
In the following list, CACertFileName and ConfigurationDirectory values are created only when certain CA installation options are specified. If these two settings are not displayed, you can proceed to the next step.

- CAServerName
- CACertFileName
- ConfigurationDirectory – This value should appear in Windows Registry under the following location:
  `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\CertSvc\Configuration`.

Verifying certificate extensions on the destination CA

The steps described for importing the source CA registry settings and editing the registry in case of a server name change are intended to retain the network locations that were used by the source CA to publish CRLs and CA certificates. If the source CA was published to default Active Directory locations, after completing the previous procedure, there should be an extension with publishing options enabled and an LDAP URL that references the source server's NetBIOS name; for example,
`ldap:///CN=<CATruncatedName><CRLNameSuffix>,CN=<ServerShortName>,CN=CDP,CN=Public Key Services,CN=Services,<ConfigurationContainer><CDPObjectClass>`.

Because many administrators configure extensions that are customized for their network environment, it is not possible to provide exact instructions for configuring CRL distribution point and authority information access extensions.

Carefully review the configured locations and publishing options, and ensure that the extensions are correct according to your organization's requirements.

To verify extensions by using the Certification Authority snap-in

1. Review and modify the CRL distribution point and authority information access extensions and publishing options by following example procedures described in Specify

CRL Distribution Points (http://go.microsoft.com/fwlink/?LinkID=145848).

2. If the destination server name is different from the source server name, add an LDAP URL specifying a location that references the destination server's NetBIOS name with the substitution variable <*ServerShortName*>; for example `ldap:///CN=<CATruncatedName><CRLNameSuffix>,CN=<ServerShortName>,CN=CDP,CN=Public Key Services,CN=Services,<ConfigurationContainer><CDPObjectClass>`.
3. Ensure that the CDP options are set so that the former CDP location is not included in the CDP extension of newly issued certificates or in the Freshest CRL extension of CRLs.

Restoring the certificate templates list

The following procedure is required only for an enterprise CA. A standalone CA does not have certificate templates.

To assign certificate templates to the destination CA

1. Log on with administrative credentials to the destination CA.
2. Open a command prompt window.
3. Type **certutil -setcatemplates +** <*templatelist*> and press ENTER.

   Note
   Replace <*templatelist*> with a comma-separated list of the template names that are listed in the catemplates.txt file created during the procedure "To record a CA templates list by using Certutil.exe." For example, **certutil -setcatemplates +Administrator,User,DomainController**. Review the list of templates created during Backing up a CA templates list.

Granting permissions on AIA and CDP containers

If the name of the destination server is different from the source server, the destination server must be granted permissions on the source server's CDP and AIA containers in AD DS to publish CRLs and CA certificates. Complete the following procedure in the case of a server name change.

To grant permissions on the AIA and CDP containers

1. Log on as a member of the Enterprise Admins group to a computer on which the Active Directory Sites and Services snap-in is installed. Open Active Directory Sites and Services (dssite.msc).
2. In the console tree, click the top node.
3. On the **View** menu, click **Show services node**.
4. In the console tree, expand **Services**, expand **Public Key Services**, and then click **AIA**.
5. In the details pane, right-click the name of the CA, and then click **Properties**.
6. Click the **Security** tab, and then click **Add**.

7. Click **Object Types**, click **Computers**, and then click **OK**.
8. Type the name of the CA, and click **OK**.
9. In the **Allow** column, click **Full Control**, and click **Apply**.
10. The previous CA computer object is displayed (as **Account Unknown** with a security identifier following it) in **Group or user names**. You can remove that account. To do so, select it and then click **Remove**. Click **OK**.
11. In the console tree, expand **CDP**, and then click the folder with the same name as the CA.
12. In the details pane, right-click the **cRLDistributionPoint** item at the top of the list, and then click **Properties**.
13. Click the **Security** tab, and then click **Add**.
14. Click **Object Types**, click **Computers**, and then click **OK**.
15. Type the name of the destination server, and click **OK**.
16. In the **Allow** column, click **Full Control**, and click **Apply**.
17. The previous CA computer object is displayed (as **Account Unknown** with a security identifier following it) in **Group or user names**. You can remove that account. To do so, select it and then click **Remove**. Click **OK**.
18. Repeat steps 13 through 18 for each **cRLDistributionPoint** item.

Notes

If you are using file//\\computer\share syntax in the CDP Extensions for publishing the CRL to a shared folder location, then you may need to adjust the permissions to that shared folder so that the destination CA has the ability to write to that location.

If you are hosting the CDP on the destination server and using a AIA or CDP path that includes an alias name (for example, pki.contoso.com) for the destination, you may need to adjust the DNS record so that it points to the correct destination IP address.

Additional procedures for failover clustering

If you are migrating to a failover cluster, complete the following procedures after the CA database and registry settings have been migrated to the destination server.

- Configuring failover clustering for the destination CA
- Granting permissions on public key containers
- Editing the DNS name for a clustered CA in AD DS
- Configuring CRL distribution points for failover clusters

Note

Migration of a CA to a failover cluster running on the Server Core installation option of Windows Server 2008 R2 is not described in this guide.

Configuring failover clustering for the destination CA

If you are migrating to a failover cluster, complete the following procedures to configure failover clustering for AD CS.

To configure AD CS as a cluster resource

1. Click **Start**, point to **Run**, type **Cluadmin.msc**, and then click **OK**.
2. In the console tree of the Failover Cluster Management snap-in, click **Services and Applications**.
3. On the **Action** menu, click **Configure a service or Application**. If the **Before you begin** page appears, click **Next**.
4. In the list of services and applications, select **Generic Service**, and click **Next**.
5. In the list of services, select **Active Directory Certificate Services**, and click **Next**.
6. Specify a service name, and click **Next**.
7. Select the disk storage that is still mounted to the node, and click **Next**.
8. To configure a shared registry hive, click **Add**, type **SYSTEM\CurrentControlSet\Services\CertSvc**, and then click **OK**. Click **Next** twice.
9. Click **Finish** to complete the failover configuration for AD CS.
10. In the console tree, double-click **Services and Applications**, and select the newly created clustered service.
11. In the details pane, click **Generic Service**. On the **Action** menu, click **Properties**.
12. Change **Resource Name** to **Certification Authority**, and click **OK**.

If you use a hardware security module (HSM) for your CA, complete the following procedure.

To create a dependency between a CA and the network HSM service

1. Open the Failover Cluster Management snap-in. In the console tree, click **Services and Applications**.
2. In the details pane, select the previously created name of the clustered service.
3. On the **Action** menu, click **Add a resource**, and then click **Generic Service**.
4. In the list of available services displayed by the **New Resource** wizard, click the name of the service that was installed to connect to your network HSM. Click **Next** twice, and then click **Finish**.
5. Under **Services and Applications** in the console tree, click the name of the clustered services.
6. In the details pane, select the newly created **Generic Service**. On the **Action** menu, click **Properties**.
7. On the **General** tab, change the service name if desired, and click **OK**. Verify that the service is online.
8. In the details pane, select the service previously named **Certification Authority**. On the

Action menu, click **Properties**.

9. On the **Dependencies** tab, click **Insert**, select the network HSM service from the list, and click **OK**.

Granting permissions on public key containers

If you are migrating to a failover cluster, complete the following procedures to grant all cluster nodes permissions to on the following AD DS containers:

- The AIA container
- The Enrollment container
- The KRA container

To grant permissions on public key containers in AD DS

1. Log on to a domain member computer as a member of the Domain Admins group or Enterprise Admins group.
2. Click **Start**, point to **Run**, type **dssite.msc**, and then click **OK**.
3. In the console tree, click the top node.
4. On the **View** menu, click **Show services node**.
5. In the console tree, expand **Services**, then **Public Key Services**, and then click **AIA**.
6. In the details pane, right-click the name of the source CA, and then click **Properties**.
7. Click the **Security** tab, and then click **Add**.
8. Click **Object Types**, click **Computers**, and then click **OK**.
9. Type the computer account names of all cluster nodes, and click **OK**.
10. In the **Allow** column, select the **Full Control** check box next to each cluster node, and click **OK**.
11. In the console tree, click **Enrollment Services**.
12. In the details pane, right-click the name of the source CA, and then click **Properties**.
13. Click the **Security** tab, and then click **Add**.
14. Click **Object Types**, click **Computers**, and then click **OK**.
15. Type the computer account names of all cluster nodes, and click **OK**.
16. In the **Allow** column, select the **Full Control** check box next to each cluster node, and click **OK**.
17. In the console tree, click **KRA**.
18. In the details pane, right-click the name of the source CA, then click **Properties**.
19. Click the **Security** tab, and then click **Add**.
20. Click **Object Types**, click **Computers**, and then click **OK**.
21. Type the names of all cluster nodes, and click **OK**.
22. In the **Allow** column, select the **Full Control** check box next to each cluster node, and click **OK**.

Editing the DNS name for a clustered CA in AD DS

When the CA service was installed on the first cluster node, the Enrollment Services object was created and the DNS name of that cluster node was added to the dNSHostName attribute of the Enrollment Services object. Because the CA must operate on all cluster nodes, the value of the dNSHostName attribute of the Enrollment Services object must be the service name specified in step 6 of the procedure "To configure AD CS as a cluster resource."

If you are migrating to a clustered CA, complete the following procedure on the active cluster node. It is necessary to complete the procedure on only one cluster node.

To edit the DNS name for a clustered CA in AD DS

1. Log on to the active cluster node as a member of the Enterprise Admins group.
2. Click **Start**, point to **Run**, type **adsiedit.msc**, and then click **OK**.
3. In the console tree, click **ADSI Edit**.
4. On the **Action** menu, click **Connect to**.
5. In the list of well-known naming contexts, click **Configuration**, and click **OK**.
6. In the console tree, expand **Configuration**, **Services**, and **Public Key Services**, and click **Enrollment Services**.
7. In the details pane, right-click the name of the cluster CA, and click **Properties**.
8. Click **dNSHostName**, and click **Edit**.
9. Type the service name of the CA as displayed under **Failover Cluster Management** in the Failover Cluster Manager snap-in, and click **OK**.
10. Click **OK** to save changes.

Configuring CRL distribution points for failover clusters

In a CA's default configuration, the server's short name is used as part of the CRL distribution point and authority information access locations.

When a CA is running on a failover cluster, the server's short name must be replaced with the cluster's short name in the CRL distribution point and authority information access locations. To publish the CRL in AD DS, the CRL distribution point container must be added manually.

Important
The following procedures must be performed on the active cluster node.

To change the configured CRL distribution points

1. Log on to the active cluster node as a member of the local Administrators group.
2. Click **Start**, click **Run**, type **regedit**, and then click **OK**.
3. Locate the registry key **\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CertSvc\Configuration**.

4. Click the name of the CA.
5. In the right pane, double-click **CRLPublicationURLs**.
6. In the second line, replace **%2** with the service name specified in step 6 of the procedure "To configure AD CS as a cluster resource."

   Tip
   The service name also appears in the Failover Cluster Management snap-in under **Services and Applications**.
7. Restart the CA service.
8. Open a command prompt, type **certutil -CRL**, and press ENTER.

   Note
   If a "Directory object not found" error message is displayed, complete the following procedure to create the CRL distribution point container in AD DS.

To create the CRL distribution point container in AD DS

1. At a command prompt, type **cd %windir%\System32\CertSrv\CertEnroll**, and press ENTER. The CRL file created by the certutil –CRL command should be located in this directory.
2. To publish the CRL in AD DS, type **certutil -f -dspublish** *"CRLFile.crl"* and press ENTER.

Next steps

After completing the procedures to migrate the CA, you should complete the procedures described in Verifying the Certification Authority Migration.

See also

- Active Directory Certificate Services Migration Guide for Windows Server 2012 R2
- Prepare to Migrate
- Verifying the Certification Authority Migration
- Post-Migration Tasks
- Migrate Roles and Features to Windows Server

Verifying the Certification Authority Migration

Complete the following procedures to verify the operation of the destination certification authority (CA).

- Verifying certificate enrollment
- Verifying CRL publishing

Verifying certificate enrollment

To verify migration to an enterprise CA, complete the procedure Request a Certificate (http://go.microsoft.com/fwlink/?LinkId=179367).

You can start autoenrollment for user certificates by completing the following procedure or by running the following command: **certutil.exe -pulse**.

To verify autoenrollment

1. Log on to a domain member computer by using an account that has Autoenroll, Enroll, and Read permissions for the certificate templates that are assigned to the destination CA.
2. Click **Start**, and then click **Run**.
3. Type **certmgr.msc**, and then click **OK** to open the Certificates snap-in.
4. In the console tree, right-click **Certificates – Current User**, click **All Tasks**, and then click **Automatically Enroll and Retrieve Certificates** to start the Certificate Enrollment wizard.
5. On the **Before You Begin** page, click **Next**.
6. On the **Request Certificates** page, a list of one or more certificate templates should be displayed. Select the check box next to each certificate template that you want to request, and then click **Enroll**.

   Note
   If the correct certificate templates are not displayed, click **Show all templates** to display all certificate templates that are assigned to the issuing CA. A status of **Unavailable** indicates the user account does not have permission to autoenroll for a certificate. Follow the steps in the "To configure certificate templates for autoenrollment" procedure earlier in this topic.
7. Click **Finish** to complete the enrollment process.
8. In the console tree, double-click **Personal**, and then click **Certificates** to display a list of installed user certificates and to verify that the certificate that you requested is displayed.

To verify migration to a standalone CA, complete the following procedure.

To verify manual enrollment by using Certreq.exe

1. Create a certificate request, and save it to a file by completing the procedure Create a Custom Certificate Request (http://go.microsoft.com/fwlink/?LinkId=179368).
2. Open a Command Prompt window.
3. Type **certreq -submit -config "<DestinationServerName\CAName>" "<CertificateRequestInput>" "<CertificateResponseOutput>"** and press ENTER.

   Note
   If a message is displayed indicating that the certificate request is pending, the certificate must be issued by a certificate manager or CA administrator by using

the Certification Authority snap-in. After the certificate is issued, it must be retrieved by using the command in step 4. If the certificate is issued immediately by the CA, the file specified in <CertificateResponseOutput> contains the certificate. Use the command in step 5 to install the certificate into the certificate store.

4. Type **certreq –retrieve -config "<DestinationServerName\CAName>" <RequestID> <CertificateResponseOutput>** and press ENTER.
5. Type **certreq –accept -config "<DestinationServerName\CAName>" <CertificateResponseOutput>** and press ENTER.

Option	Description	Example
-config	The –config option is followed by a string specifying a host name and CA name in the format HostName\CAName.	Certreq.exe –submit –config Server1\CA1 C:\RequestFile.txt C:\ResponseFile.cer
DestinationServerName	The host name of the destination server.	
CAName	The CA name being migrated.	
CertificateRequestInput	The path and name of the file containing the certificate request that was created by using the procedure "Create a Custom Certificate Request."	
CertificateResponseOutput	The path and name of the file receiving the issued certificate from the CA. If the certificate request is pending, the file contains a message from the CA indicating the status of the request and the request ID. The request ID is used to retrieve the certificate after it is issued by a certificate manager or CA administrator.	

Verifying CRL publishing

If you published a certificate revocation list (CRL) with an extended validity period before beginning migration, you should change the CRL publishing period back to its pre-migration value by completing the procedure Schedule the publication of the certificate revocation list.

Manually publish a CRL by completing one of the procedures described in Manually Publish a CRL.

Next steps

After completing verification steps, you should review the topic Post-Migration Tasks and complete the procedures appropriate for your environment.

See also

- Active Directory Certificate Services Migration Guide for Windows Server 2012 R2
- Prepare to Migrate
- Migrating the Certification Authority
- Post-Migration Tasks
- Migrate Roles and Features to Windows Server

Post-Migration Tasks

Post-migration steps can be performed after migration has been completed and the operation of the destination CA has been verified.

If verification steps have failed, review the Troubleshooting section in this topic.

- Upgrading certificate templates in Active Directory Domain Services (AD DS)
- Retrieving certificates after a host name change
- Restoring Active Directory Certificate Services (AD CS) to the source server in the event of migration failure
- Troubleshooting migration

Upgrading certificate templates in Active Directory Domain Services (AD DS)

Review the post-migration steps below and perform only those that are appropriate for your environment and migration scenario.

The following additional default certificate templates are included in enterprise certification authorities (CAs) running on Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2 and Windows Server 2008 but are not included in Windows Server 2003:

- OCSP Response Signing
- Kerberos Authentication

These certificate templates are not required for CA operation. OCSP Response Signing certificates are required if you are deploying the Online Responder role service.

If you require these additional certificate templates, complete the following procedure.

To upgrade certificate templates in AD DS by using the Certificate Templates snap-in

1. Log on to the destination server as a member of the Enterprise Admins group.
2. Open the Certificate Templates snap-in. The snap-in automatically adds the default certificate templates to AD DS.

Retrieving certificates after a host name change

If the destination server name is different from the source server name, you might need to manually retrieve any certificates that were issued by the source CA and had not been retrieved before migration.

Complete this procedure on the computer that was used to submit the certificate request to the source CA.

To retrieve a certificate by using Certreq.exe

1. Open a Command Prompt window.
2. Type **certreq –retrieve -config "<DestinationServerName\CAName>" <RequestID> <CertificateResponseOutput>** and press ENTER.
3. Type **certreq –accept <CertificateResponseOutput>** and press ENTER.

Option	Description	Example
-config	The –config option is followed by a string specifying a host name and CA name in the format HostName\CAName.	Certreq.exe –submit –config Server1\CA1 C:\RequestFile.txt C:\ResponseFile.cer
DestinationServerName	The host name of the destination server.	
CAName	The CA name being migrated.	
CertificateRequestInput	The path and name of the file containing the certificate request that was created by using the procedure "Create a Custom Certificate Request."	

Option	Description	Example
CertificateResponseOutput	The path and name of the file receiving the issued certificate from the CA. If the certificate request is pending, the file contains a message from the CA indicating the status of the request and the request ID. The request ID is used to retrieve the certificate after it is issued by a certificate manager or CA administrator.	
RequestID	The Request ID value returned by a CA in response to a certificate request. The Request ID value is displayed in command output and written to the CertificateResponseOutput file.	

Restoring Active Directory Certificate Services (AD CS) to the source server in the event of migration failure

If you removed the CA role service from the source server as described in the procedure Removing the CA role service from the source server, you can restore the source CA by reinstalling the CA role service on the source server. It is important to remove the CA role service from the destination server before reinstalling the CA role service on the source server.

If you did not remove the CA role service from the source server, you should not remove the CA role service from the destination server. Simply shut down the destination CA and start the source CA.

Rollback procedures can be completed in less than one hour.

To remove the CA role service from the destination server, use the Remove Roles Wizard in Server Manager.

To add the CA role service to a source server running Windows Server 2003, use the Add/Remove Windows Components wizard.

To add the CA role service to a source server running Windows Server 2008 or later, use the Add Roles Wizard in Server Manager.

Troubleshooting migration

If you encounter errors during verification procedures, use Event Viewer to review the Application log on the destination CA. View an Error event in the preview pane or event properties, and click **Event Log Online Help** to open a Web page with troubleshooting procedures for that event.

For the full collection of documented AD CS events, see AD CS Events and Errors.

See also

- Active Directory Certificate Services Migration Guide for Windows Server 2012 R2
- Prepare to Migrate
- Migrating the Certification Authority
- Verifying the Certification Authority Migration
- Migrate Roles and Features to Windows Server

Migrating Active Directory Federation Services Role Service to Windows Server 2012 R2

About this guide

This guide provides instructions to migrate the following role services to Active Directory Federation Services (AD FS) that is installed with Windows Server® 2012 R2:

- AD FS 2.0 federation server installed on Windows Server 2008 or Windows Server 2008 R2
- AD FS federation server installed on Windows Server 2012

Target audience

- IT architects who are responsible for computer management and security throughout an organization
- IT operations engineers who are responsible for the day-to-day management and troubleshooting of networks, servers, client computers, operating systems, or applications
- IT operations managers who are accountable for network and server management

Supported migration scenarios

The migration instructions in this guide consist of the following tasks:

- Exporting the AD FS 2.0 configuration data from your server that is running Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012

- Performing an in-place upgrade of the operating system of this server from Windows Server 2008, Windows Server 2008 R2 or Windows Server 2012 to Windows Server® 2012 R2
- Recreating the original AD FS configuration and restoring the remaining AD FS service settings on this server, which is now running the AD FS server role that is installed with Windows Server® 2012 R2.

This guide does not include instructions to migrate a server that is running multiple roles. If your server is running multiple roles, we recommend that you design a custom migration process specific to your server environment, based on the information provided in other role migration guides. Migration guides for additional roles are available on the Windows Server Migration Portal.

Supported operating systems

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2008, both full and Server Core installation options	Windows Server® 2012 R2 (Server Core and full installation options)	x64-based
x64-based	Windows Server 2008 R2		
x64-based	Server Core installation option of Windows Server 2008 R2		
x64-based	Server Core and full installation options of Windows Server 2012		

Notes

- The versions of operating systems that are listed in the preceding table are the oldest combinations of operating systems and service packs that are supported.
- The Foundation, Standard, Enterprise, and Datacenter editions of the Windows Server operating system are supported as the source or the destination server.
- Migrations between physical operating systems and virtual operating systems are supported.

Supported AD FS role services and features

The following table describes the migration scenarios of the AD FS role services and their respective settings that are described in this guide.

From	To AD FS installed with Windows Server® 2012 R2
AD FS 2.0 federation server installed on Windows Server 2008 or Windows Server 2008 R2	Migration on the same server is supported. For more information, see: • Preparing to Migrate the AD FS Federation Server • Migrating the AD FS Federation Server
AD FS federation server installed on Windows Server 2012	Migration on the same server is supported. For more information see: • Preparing to Migrate the AD FS Federation Server • Migrating the AD FS Federation Server

See Also

Preparing to Migrate the AD FS Federation Server

Migrating the AD FS Federation Server

Migrating the AD FS Federation Server Proxy

Verifying the AD FS Migration to Windows Server 2012 R2

Preparing to Migrate the AD FS Federation Server

To perform the same server migration of your AD FS federation server farm for Windows Server 2012 R2, you must review the following information:

Note
The information below applies to migrating a one-node federation server, as well as a WID or a SQL Server federation server farm. It applies to the migration of a federation server running AD FS 2.0 running on Windows Server 2008 or Windows Server 2008 R2 or AD FS installed with Windows Server 2012.

- Migration Process Outline
- New AD FS functionality in Windows Server 2012 R2
- AD FS Requirements in Windows Server 2012 R2
- Increasing your Windows PowerShell limits
- Other migration tasks and considerations

Migration Process Outline

To complete the migration of your AD FS federation server farm to Windows Server 2012 R2, you must complete the following tasks:

1. Export, record, and backup the following configuration data in your existing AD FS farm. For detailed instructions on how to complete these tasks, see Migrating the AD FS Federation Server.

   The following settings are migrated with the scripts located in the \support\adfs folder on the Windows Server 2012 R2 installation CD:

   - Claims provider trusts, with the exception of custom claim rules on the Active Directory Claims provider trust. For more information, see Migrating the AD FS Federation Server.
   - Relying party trusts.
   - AD FS internally generated, self-signed token signing and token decryption certificates.

   Any of the following custom settings must be migrated manually:

   - Service settings:
     - Non-default token signing and token decryption certificates that were issued by an enterprise or public certification authority.
     - The SSL server authentication certificate used by AD FS.
     - The service communications certificate used by AD FS (by default, this is the same certificate as the SSL certificate.
     - Non-default values for any federation service properties, such as AutoCertificateRollover or SSO lifetime.
     - Non-default AD FS endpoint settings and claim descriptions.
   - Custom claim rules on the Active Directory claims provider trust.
   - AD FS sign-in page customizations

   For more information, see Migrating the AD FS Federation Server.
2. Create a Windows Server 2012 R2 federation server farm.
3. Import the original configuration data into this new Windows Server 2012 R2 AD FS farm.
4. Configure and customize the AD FS sign-in pages.

New AD FS functionality in Windows Server 2012 R2

The following AD FS functionality changes in Windows Server 2012 R2 impact a migration from AD FS 2.0 or AD FS in Windows Server 2012:

- **IIS dependency**

  AD FS in Windows Server 2012 R2 is self-hosted and does not require IIS installation. Make sure you note the following as a result of this change:

  - SSL certificate management for both federation servers and proxy computers in your AD FS farm must now be performed via Windows PowerShell.

- **Changes to AD FS sign-in pages' settings and customizations**

  In AD FS in Windows Server 2012 R2, there are several changes intended to improve the sign-in experience for both administrators and users. The IIS-hosted web pages that existed in the previous version of AD FS are now removed. The look and feel of the AD FS sign-in web pages are self-hosted in AD FS and can now be customized to tailor the user experience. The changes include:

  - Customizing the AD FS sign-in experience, including the customization of the company name, logo, illustration, and sign-in description.
  - Customizing the error messages.
  - Customizing the ADFS Home Realm Discovery experience, which includes the following:
    - Configuring your identity provider to use certain email suffixes.
    - Configuring an identity provider list per relying party.
    - Bypassing Home Realm Discovery for intranet.
    - Creating custom web themes.

  For detailed instructions on configuring the look and feel of the AD FS sign-in pages, see Customizing the AD FS Sign-in Pages.

  If you have web page customization in your existing AD FS farm that you want to migrate to Windows Server 2012 R2, you can recreate them as part of the migration process using the new customization features in Windows Server 2012 R2.

- **Other changes**
  - AD FS in Windows Server 2012 R2 is based on Windows Identity Foundation (WIF) 3.5, not WIF 4.5. Therefore, some specific features of WIF 4.5 (for example, Kerberos claims and dynamic access control) are not supported in AD FS in Windows Server 2012 R2.
  - Device Registration Service (DRS) in Windows Server 2012 R2 operates on port 443; ClientTLS for user certificate authentication operates on port 49443
    - For active, non-browser clients using certificate transport mode authentication that are specifically hard-coded to point to port 443, a code change is required to continue to use user certificate authentication on port 49443.
    - For passive applications no change is required because AD FS redirects to the correct port for user certificate authentication.
    - Firewall ports between the client and the proxy must enable port 49443 traffic to pass through for user certificate authentication.

AD FS Requirements in Windows Server 2012 R2

In order to successfully migrate your AD FS farm to Windows Server 2012 R2, you must meet the following requirements:

For AD FS to function, each computer that you want to be a federation server must be joined to a domain.

For AD FS running on Windows Server 2012 R2 to function, your Active Directory domain must run either of the following:

- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2
- Windows Server 2008

If you plan to use a group Managed Service Account (gMSA) as the service account for AD FS, you must have at least one domain controller in your environment that is running on Windows Server 2012 or Windows Server 2012 R2 operating system.

If you plan to deploy Device Registration Service (DRS) for AD Workplace Join as a part of your AD FS deployment, the AD DS schema needs to be updated to the Windows Server 2012 R2level. There are three ways to update the schema:

1. In an existing Active Directory forest, run adprep /forestprep from the \support\adprep folder of the Windows Server 2012 R2 operating system DVD on any 64-bit server that runs Windows Server 2008 or later. In this case, no additional domain controller needs to be installed, and no existing domain controllers need to be upgraded.

   To run adprep/forestprep, you must be a member of the Schema Admins group, the Enterprise Admins group, and the Domain Admins group of the domain that hosts the schema master.
2. In an existing Active Directory forest, install a domain controller that runs Windows Server 2012 R2. In this case, adprep /forestprep runs automatically as part of the domain controller installation.

   During the domain controller installation, you may need to specify additional credentials in order to run adprep /forestprep.
3. Create a new Active Directory forest by installing AD DS on a server that runs Windows Server 2012 R2. In this case, adprep /forestprep does not need to be run because the schema will be initially created with all the necessary containers and objects to support DRS.

SQL Server support for AD FS in Windows Server 2012 R2

If you want to create an AD FS farm and use SQL Server to store your configuration data, you can use SQL Server 2008 and newer versions, including SQL Server 2012.

Increasing your Windows PowerShell limits

If you have more than 1000 claims provider trusts and relying party trusts in your AD FS farm, or if you see the following error while trying to run the AD FS migration export/import tool, you must increase your Windows PowerShell limits:

```
'Exception of type 'System.OutOfMemoryException' was thrown. At
E:\dev\ds\security\ADFSv2\Product\Migration\Export-FederationConfiguration.ps1:176
char:21 + $configData = Invoke-Command -ScriptBlock $GetConfig -Argume ...
```

This error is thrown because the Windows PowerShell session default memory limit is too low. In Windows PowerShell 2.0, the session default memory is 150MB. In Windows PowerShell 3.0, the session default memory is 1024MB. You can verify Windows PowerShell remote session memory

limit using the following command: `Get-Item wsman:localhost\Shell\MaxMemoryPerShellMB`. You can increase the limit by running the following command: `Set-Item wsman:localhost\Shell\MaxMemoryPerShellMB 512`.

Other migration tasks and considerations

In order to successfully migrate your AD FS farm to Windows Server 2012 R2, make sure you are aware of the following:

- The migration scripts located in the \support\adfs folder on the Windows Server 2012 R2 installation CD require that you retain the same federation server farm name and service account identity name that you used in your legacy AD FS farm when you migrate it to Windows Server 2012 R2.
- If you want to migrate a SQL Server AD FS farm, note that the migration process involves creating a new SQL database instance into which you must import the original configuration data.

See Also

Migrating Active Directory Federation Services Role Service to Windows Server 2012 R2

Migrating the AD FS Federation Server

To migrate an AD FS federation server that belongs to a single-node AD FS farm, a WIF farm, or a SQL Server farm to Windows Server 2012 R2, you must perform the following tasks:

1. Export and backup the AD FS configuration data
2. Create a Windows Server 2012 R2 federation server farm
3. Import the original configuration data into the Windows Server 2012 R2 AD FS farm
4. **Configure and customize the AD FS sign-in pages in the migrated AD FS farm**
5. **Other migration tasks**

Export and backup the AD FS configuration data

To export the AD FS configuration settings, perform the following procedures:

- To export service settings
- To export claims provider trusts and relying party trusts
- **To export relying party trusts**
- To back up custom attribute stores
- **To back up AD FS sign-in pages' settings and customizations**

To export service settings

1. Make sure that you have access to the following certificates and their private keys in a .pfx file:
   - The SSL certificate that is used by the federation server farm that you want to migrate
   - The service communication certificate (if it is different from the SSL certificate) that is used by the federation server farm that you want to migrate
   - All third-party party token-signing or token-encryption/decryption certificates that are used by the federation server farm that you want to migrate

   To find the SSL certificate, open the Internet Information Services (IIS) management console, Select **Default Web Site** in the left pane, click **Bindings…** in the **Action** pane, find and select the https binding, click **Edit**, and then click **View**.

   You must export the SSL certificate used by the federation service and its private key to a .pfx file. For more information, see Export the Private Key Portion of a Server Authentication Certificate.

   Note
   If you plan to deploy the Device Registration Service as part of running your AD FS in Windows Server 2012 R2, you must obtain a new SSL cert. For more information, see Enroll an SSL Certificate for AD FS and Configure a federation server with Device Registration Service.

   To view the token signing, token decryption and service communication certificates that are used, run the following Windows PowerShell command to create a list of all certificates in use in a file:

   ```
   Get-ADFSCertificate | Out-File ".\certificates.txt"
   ```

2. Export AD FS federation service properties, such as the federation service name, federation service display name, and federation server identifier to a file.

   To export federation service properties, open Windows PowerShell and run the following command: `PSH:> Get-ADFSProperties | Out-File ".\properties.txt"`.

   The output file will contain the following important configuration values:

Federation Service Property name as reported by Get-ADFSProperties	Federation Service Property name in AD FS management console
HostName	Federation Service name
Identifier	Federation Service identifier
DisplayName	Federation Service display name

3. Back up the application configuration file. Among other settings, this file contains the policy database connection string.

   To back up the application configuration file, you must manually copy the `%programfiles%\Active Directory Federation Services`

2.0\Microsoft.IdentityServer.Servicehost.exe.config file to a secure location on a backup server.

Notes

Make note of the database connection string in this file, located immediately after "policystore connectionstring="). If the connection string specifies a SQL Server database, the value is needed when restoring the original AD FS configuration on the federation server.

The following is an example of a WID connection string: "Data Source=\\.\pipe\mssql$microsoft##ssee\sql\query;Initial Catalog=AdfsConfiguration;Integrated Security=True". The following is an example of a SQL Server connection string: "Data Source=databasehostname;Integrated Security=True".

4. Record the identity of the AD FS federation service account and the password of this account.

   To find the identity value, examine the **Log On As** column of **AD FS 2.0 Windows Service** in the **Services** console and manually record this value.

   Note

   For a stand-alone federation service, the built-in NETWORK SERVICE account is used. In this case, you do not need to have a password.

5. Export the list of enabled AD FS endpoints to a file.

   To do this, open Windows PowerShell and run the following command: PSH:> Get-ADFSEndpoint | Out-File ".\endpoints.txt".

6. Export any custom claim descriptions to a file.

   To do this, open Windows PowerShell and run the following command: Get-ADFSClaimDescription | Out-File ".\claimtypes.txt".

7. If you have custom settings such as useRelayStateForIdpInitiatedSignOn configured in the web.config file, ensure you back up the web.config file for reference. You can copy the file from the directory that is mapped to the virtual path **"/adfs/ls"** in IIS. By default, it is in the **%systemdrive%\inetpub\adfs\ls** directory.

To export claims provider trusts and relying party trusts

1. To export AD FS claims provider trusts and relying party trusts, you must log in as Administrator (however, not as the Domain Administrator) onto your federation server and run the following Windows PowerShell script that is located in the **media/server_en-us/support/adfs** folder of the Windows Server 2012 R2 installation CD: export-federationconfiguration.ps1.

   Important

   The export script takes the following parameters:

   - Export-FederationConfiguration.ps1 -Path <string> [-ComputerName <string>] [-Credential <pscredential>] [-Force] [-CertificatePassword <securestring>]

- Export-FederationConfiguration.ps1 -Path <string> [-ComputerName <string>] [-Credential <pscredential>] [-Force] [-CertificatePassword <securestring>] [-RelyingPartyTrustIdentifier <string[]>] [-ClaimsProviderTrustIdentifier <string[]>]
- Export-FederationConfiguration.ps1 -Path <string> [-ComputerName <string>] [-Credential <pscredential>] [-Force] [-CertificatePassword <securestring>] [-RelyingPartyTrustName <string[]>] [-ClaimsProviderTrustName <string[]>]

  -RelyingPartyTrustIdentifier <string[]> - the cmdlet only exports relying party trusts whose identifiers are specified in the string array. The default is to export NONE of the relying party trusts. If none of RelyingPartyTrustIdentifier, ClaimsProviderTrustIdentifier, RelyingPartyTrustName, and ClaimsProviderTrustName is specified, the script will export all relying party trusts and claims provider trusts.

  -ClaimsProviderTrustIdentifier <string[]> - the cmdlet only exports claims provider trusts whose identifiers are specified in the string array. The default is to export NONE of the claims provider trusts.

  -RelyingPartyTrustName <string[]> - the cmdlet only exports relying party trusts whose names are specified in the string array. The default is to export NONE of the relying party trusts.

  -ClaimsProviderTrustName <string[]> - the cmdlet only exports claims provider trusts whose names are specified in the string array. The default is to export NONE of the claims provider trusts.

  -Path <string> - the path to a folder that will contain the exported files.

  -ComputerName <string> - specifies the STS server host name. The default is the local computer. If you are migrating AD FS 2.0 or AD FS in Windows Server 2012 to AD FS in Windows Server 2012 R2, this is the host name of the legacy AD FS server.

  -Credential <PSCredential> - specifies a user account that has permission to perform this action. The default is the current user.

  -Force – specifies to not prompt for user confirmation.

  -CertificatePassword <SecureString> - specifies a password for exporting AD FS certificates' private keys. If not specified, the script will prompt for a password if an AD FS certificate with private key needs to be exported.

  Inputs: None

  Outputs: string - this cmdlet returns the export folder path. You can pipe the returned object to Import-FederationConfiguration.

To back up custom attribute stores

1. You must manually export all custom attribute stores that you want to keep in your new AD FS farm in Windows Server 2012 R2.

   Note

In Windows Server 2012 R2, AD FS requires custom attribute stores that are based on .NET Framework 4.0 or above. Follow the instructions in Microsoft .NET Framework 4.5 to install and setup .Net Framework 4.5.

You can find information about custom attribute stores in use by AD FS by running the following Windows PowerShell command: `PSH:>Get-ADFSAttributeStore`. The steps to upgrade or migrate custom attribute stores vary.

2. You must also manually export all .dll files of the custom attribute stores that you want to keep in your new AD FS farm in Windows Server 2012 R2. The steps to upgrade or migrate .dll files of custom attribute stores vary.

Create a Windows Server 2012 R2 federation server farm

1. Install the Windows Server 2012 R2 operating system on a computer that you want to function as a federation server and then add the AD FS server role. For more information, see Install the AD FS Role Service. Then configure your new federation service either through the Active Directory Federation Service Configuration Wizard or via Windows PowerShell. For more information, see “Configure the first federation server in a new federation server farm” in Configure a Federation Server.

   While completing this step, you must follow these instructions:

   - You must have Domain Administrator privileges in order to configure your federation service.
   - You must use the same federation service name (farm name) as was used in the AD FS 2.0 or AD FS in Windows Server 2012. If you do not use the same federation service name, the certificates that you backed up will not function in the Windows Server 2012 R2 federation service that you are trying to configure.
   - Specify whether this is a WID or SQL Server federation server farm. If it is a SQL farm, specify the SQL Server database location and instance name.
   - You must provide a pfx file containing the SSL server authentication certificate that you backed up as part of preparing for the AD FS migration process.
   - You must specify the same service account identity that was used in the AD FS 2.0 or AD FS in Windows Server 2012 farm.

2. Once the initial node is configured, you can add additional nodes to your new farm. For more information, see “Add a federation server to an existing federation server farm” in Configure a Federation Server.

Import the original configuration data into the Windows Server 2012 R2 AD FS farm

Now that you have an AD FS federation server farm running in Windows Server 2012 R2, you can import the original AD FS configuration data into it.

1. Import and configure other custom AD FS certificates, including externally enrolled token-signing and token- decryption/encryption certificates, and the service communication certificate if it is different from the SSL certificate.

   In the AD FS management console, select **Certificates**. Verify the service communications, token-encryption/decryption, and token-signing certificates by checking each against the values you exported into the certificates.txt file while preparing for the migration.

   To change the token-decrypting or token-signing certificates from the default self-signed certificates to external certificates, you must first disable the automatic certificate rollover feature that is enabled by default. To do this, you can use the following Windows PowerShell command:

   ```
   Set-ADFSProperties –AutoCertificateRollover $false.
   ```

2. Configure any custom AD FS service settings such as AutoCertificateRollover or SSO lifetime using the Set-AdfsProperties cmdlet.
3. To import AD FS relying party trusts and claims provider trusts, you must be logged in as Administrator (however, not as the Domain Administrator) onto your federation server and run the following Windows PowerShell script that is located in the \support\adfs folder of the Windows Server 2012 R2 installation CD:

   ```
   import-federationconfiguration.ps1
   ```

   Important

   The import script takes the following parameters:

   - Import-FederationConfiguration.ps1 -Path <string> [-ComputerName <string>] [-Credential <pscredential>] [-Force] [-LogPath <string>] [-CertificatePassword <securestring>]
   - Import-FederationConfiguration.ps1 -Path <string> [-ComputerName <string>] [-Credential <pscredential>] [-Force] [-LogPath <string>] [-CertificatePassword <securestring>] [-RelyingPartyTrustIdentifier <string[]>] [-ClaimsProviderTrustIdentifier <string[]>
   - Import-FederationConfiguration.ps1 -Path <string> [-ComputerName <string>] [-Credential <pscredential>] [-Force] [-LogPath <string>] [-CertificatePassword <securestring>] [-RelyingPartyTrustName <string[]>] [-ClaimsProviderTrustName <string[]>]

     -RelyingPartyTrustIdentifier <string[]> - the cmdlet only imports relying party trusts whose identifiers are specified in the string array. The default is to import

NONE of the relying party trusts. If none of RelyingPartyTrustIdentifier, ClaimsProviderTrustIdentifier, RelyingPartyTrustName, and ClaimsProviderTrustName is specified, the script will import all relying party trusts and claims provider trusts.

-ClaimsProviderTrustIdentifier <string[]> - the cmdlet only imports claims provider trusts whose identifiers are specified in the string array. The default is to import NONE of the claims provider trusts.

-RelyingPartyTrustName <string[]> - the cmdlet only imports relying party trusts whose names are specified in the string array. The default is to import NONE of the relying party trusts.

-ClaimsProviderTrustName <string[]> - the cmdlet only imports claims provider trusts whose names are specified in the string array. The default is to import NONE of the claims provider trusts.

-Path <string> - the path to a folder that contains the configuration files to be imported.

-LogPath <string> - the path to a folder that will contain the import log file. A log file named “import.log” will be created in this folder.

-ComputerName <string> - specifies host name of the STS server. The default is the local computer. If you are migrating AD FS 2.0 or AD FS in Windows Server 2012 to AD FS in Windows Server 2012 R2, this parameter should be set to the hostname of the legacy AD FS server.

-Credential <PSCredential>- specifies a user account that has permission to perform this action. The default is the current user.

-Force – specifies to not prompt for user confirmation.

-CertificatePassword <SecureString> - specifies a password for importing AD FS certificates’ private keys. If not specified, the script will prompt for a password if an AD FS certificate with private key needs to be imported.

Inputs: string - this command takes the import folder path as input. You can pipe Export-FederationConfiguration to this command.

Outputs: None.

Any trailing spaces in the WSFedEndpoint property of a relying party trust may cause the import script to error. In this case, manually remove the spaces from the file prior to import. For example, these entries cause errors:

```
<URI N="WSFedEndpoint">https://127.0.0.1:444 /</URI>
<URI N="WSFedEndpoint">https://myapp.cloudapp.net:83 /</URI>
```

They must be edited to:

```
<URI N="WSFedEndpoint">https://127.0.0.1:444/</URI>
<URI N="WSFedEndpoint">https://myapp.cloudapp.net:83/</URI>
```

Important

If you have any custom claim rules (rules other than the AD FS default rules) on the Active Directory claims provider trust in the source system, these will not be migrated by the scripts. This is because Windows Server 2012 R2 has new defaults. Any custom rules must be merged by adding them manually to the Active Directory claims provider trust in the new Windows Server 2012 R2 farm.

4. Configure all custom AD FS endpoint settings. In the AD FS Management console, select **Endpoints**. Check the enabled AD FS endpoints against the list of enabled AD FS endpoints that you exported to a file while preparing for the AD FS migration.

   - And -

   Configure any custom claim descriptions. In the AD FS Management console, select **Claim Descriptions**. Check the list of AD FS claim descriptions against the list of claim descriptions that you exported to a file while preparing for the AD FS migration. Add any custom claim descriptions included in your file but not included in the default list in AD FS. Note that Claim identifier in the management console maps to the ClaimType in the file.

5. Install and configure all backed up custom attribute stores. As an administrator, ensure any custom attribute store binaries are upgrade to .NET Framework 4.0 or higher before updating the AD FS configuration to point to them.
6. Configure service properties that map to the legacy web.config file parameters.
   - If **useRelayStateForIdpInitiatedSignOn** was added to the **web.config** file in your AD FS 2.0 or AD FS in Windows Sever 2012 farm, then you must configure the following service properties in your AD FS in Windows Server 2012 R2 farm:
     - AD FS in Windows Server 2012 R2 includes a **%systemroot%\ADFS\Microsoft.IdentityServer.Servicehost.exe.config** file. Create an element with the same syntax as the **web.config** file element: `<useRelayStateForIdpInitiatedSignOn enabled="true" />`. Include this element as part of **<microsoft.identityserver.web>** section of the **Microsoft.IdentityServer.Servicehost.exe.config** file.
   - If **<persistIdentityProviderInformation enabled="true|false" lifetimeInDays="90" enablewhrPersistence="true|false" />** was added to the **web.config** file in your AD FS 2.0 or AD FS in Windows Sever 2012 farm, then you must configure the following service properties in your AD FS in Windows Server 2012 R2 farm:

     i. In AD FS in Windows Server 2012 R2, run the following Windows PowerShell command: `Set-AdfsWebConfig –HRDCookieEnabled –HRDCookieLifetime`.
   - If **<singleSignOn enabled="true|false" />** was added to the **web.config** file in your AD FS 2.0 or AD FS in Windows Sever 2012 farm, you do not need to set any additional service properties in your AD FS in Windows Server 2012 R2 farm. Single sign-on is enabled by default in AD FS in Windows Server 2012 R2 farm.
   - If localAuthenticationTypes settings were added to the **web.config** file in your AD FS 2.0 or AD FS in Windows Sever 2012 farm, then you must configure the following service properties in your AD FS in Windows Server 2012 R2 farm:
     - Integrated, Forms, TlsClient, Basic Transform list into equivalent AD FS in

Windows Server 2012 R2 has global authentication policy settings to support both federation service and proxy authentication types. These settings can be configured in the AD FS in Management snap-in under the **Authentication Policies**.

After you import the original configuration data, you can customize the AD FS sign in pages as needed. For more information, see Customizing the AD FS Sign-in Pages.

See Also

Migrating Active Directory Federation Services Role Service to Windows Server 2012 R2

Migrating the AD FS Federation Server Proxy

In Active Directory Federation Services (AD FS) in Windows Server 2012 R2, the role of a federation server proxy is handled by a new Remote Access role service called Web Application Proxy. In Windows Server 2012 R2, to enable your AD FS for accessibility from outside of the corporate network, you can deploy one or more Web Application Proxies. However, you cannot migrate a federation server proxy running on Windows Server 2008 R2 or Windows Server 2012 to a Web Application Proxy running on Windows Server 2012 R2.

Important
The migration of a federation server proxy running on Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012 to a Web Application Proxy running on Windows Server 2012 R2 is NOT supported.

If you want to configure AD FS in a Windows Server 2012 R2 migrated farm for extranet access, you must perform a fresh deployment of one or more Web Application Proxy computers as part of your AD FS infrastructure.

To plan Web Application Proxy deployment, you can review the information in the following topics:

- Step 1: Plan the Web Application Proxy Infrastructure
- Step 2: Plan the Web Application Proxy Server

To deploy Web Application proxy, you can follow the procedures in the following topics:

- Step 1: Configure the Web Application Proxy Infrastructure
- Step 2: Install and Configure the Web Application Proxy Server

See Also

Migrating Active Directory Federation Services Role Service to Windows Server 2012 R2

Verifying the AD FS Migration to Windows Server 2012 R2

Once you complete the same server migration of your Active Directory Federation Service (AD FS) farm to Windows Server 2012 R2, you can use the following procedure to verify that federation servers in your farm are operational; that is, that any client on the same network can reach your federation servers.

Membership in **Users**, **Backup Operators**, **Power Users**, **Administrators** or equivalent, on the local computer is the minimum required to complete this procedure. Review details about using the appropriate accounts and group memberships at Local and Domain Default Groups (http://go.microsoft.com/fwlink/?LinkId=83477).

To verify that a federation server is operational

1. Open a browser window and in the address bar, type the federation server name, and then append it with **federationmetadata/2007-06/federationmetadata.xml** to browse to the federation service metadata endpoint. For example, **https://fs.contoso.com/federationmetadata/2007-06/federationmetadata.xml** .

   If in your browser window you can see the federation server metadata without any SSL errors or warnings, your federation server is operational.
2. You can also browse to the AD FS sign-in page (your federation service name appended with **adfs/ls/idpinitiatedsignon.htm**, for example, **https://fs.contoso.com/adfs/ls/idpinitiatedsignon.htm**). This displays the AD FS sign-in page where you can sign in with domain administrator credentials.

   Important
   Make sure to configure your browser settings to trust the federation server role by adding your federation service name (for example, **https://fs.contoso.com**) to the browser's local intranet zone.

See Also

Migrating Active Directory Federation Services Role Service to Windows Server 2012 R2

Migrate DHCP Server to Windows Server 2012 R2

DHCP server role migration involves moving the settings for your existing DHCP server to a new DHCP server on the network. The goal of this server migration is to install the DHCP server role on the Windows Server® 2012 R2 operating system so that it provides DHCP leases on a network without any perceptible change to DHCP client computers.

About this guide

This guide describes the steps for migrating existing DHCP server settings to a server that is running Windows Server 2012 R2. Migration documentation and tools ease the migration of server role settings and data from an existing server to a destination server that is running Windows Server 2012 R2. By using the tools that are described in this guide to migrate a DHCP server, you can simplify migration, reduce migration time, increase the accuracy of the migration process, and help eliminate possible conflicts that might otherwise occur during DHCP migration. For more information about the migration tools, see DHCP Server Migration: Appendix A.

Target audience

This guide is intended for information technology (IT) administrators, IT professionals, and other knowledge workers who are responsible for the operation and deployment of DHCP servers in a managed environment.

What this guide does not provide

The following scenarios are not supported or are beyond the scope of this guide.

- Clustering scenarios are not supported by this migration process. For more information about migrating DHCP Server in a cluster environment, see Migrating DHCP to a Cluster Running Windows Server 2008 R2 Step-by-Step Guide (http://go.microsoft.com/fwlink/?LinkId=140512) on the Windows Server TechCenter.

  Also see Migrate to DHCP Failover. DHCP failover is a new option for DHCP high availability, introduced in Windows Server 2012.
- Upgrading roles on the same computer is out of scope for this guide.
- Scenarios in which the new operating system is installed on existing server hardware by using the **Upgrade** option during setup (in-place upgrades) are not covered in this guide.
- Migrating more than one server role is not covered in this guide.

Supported migration scenarios

This guide gives you the instructions to migrate an existing DHCP server to a server that is running Windows Server 2012 R2. This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, we recommend that you design a custom migration procedure specific to your server environment based on the information provided in other role migration guides. Migration guides for additional roles are available on the Windows Server 2012 TechCenter (http://technet.microsoft.com/library/jj134039.aspx).

Caution

If the source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other roles that are running on the source server to fail.

This guide provides instructions only for migrating DHCP data and settings from a server that is being replaced by an x64-based server running Windows Server 2012 R2.

Supported operating systems

This guide provides instructions for migration of a DHCP server from a server that is running Windows Server 2003 or a later operating system to a server running Windows Server 2012 R2. Supported operating systems are listed in the following table.

Supported operating systems for migration

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 with Service Pack 2	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2003 R2	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2008	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2008 R2	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x64-based	Windows Server 2012	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2012	Windows Server 2012 R2, both full and Server Core installation options	x64-based

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x64-based	Windows Server 2012 R2	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2012 R2	Windows Server 2012 R2, both full and Server Core installation options	x64-based

The versions of operating systems shown in the previous table are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are supported for the migration of DHCP server settings.

Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported as either source or destination servers.

Migrations between physical operating systems and virtual operating systems are supported.

Migration from a source server to a destination server that is running an operating system in a different system user interface (UI) language than the source server is not supported. The system UI language is the language of the localized installation package that was used to set up the Windows operating system. For example, you cannot use Windows Server migration tools to migrate roles, operating system settings, data, or shares from a computer that is running Windows Server 2008 R2 in the French system UI language to a computer that is running Windows Server 2012 R2 in the German system UI language.

Both x86-based and x64-based migrations are supported for Windows Server 2003 and Windows Server 2008. All editions of Windows Server 2012 R2 are x64-based.

Roles that are running on Server Core installations of Windows Server 2008 cannot be migrated, because there is no .NET Framework available on Server Core installations of Windows Server 2008.

We recommend migration rather than an upgrade even when the hardware is native x64-based. For example, with a server role split, a scenario in which the source server has more than one server role, because of increased use of this server you might decide to separate the roles onto several additional x64-based servers. In this case, migrating (not upgrading) individual server roles to other servers may be the best solution.

The server administrator can choose which components of an existing installation to migrate; together with the server role, these components usually include configuration, data, system identity, and operating system settings.

Supported role configurations

You can migrate all DHCP Server settings by using this guide, including registry and database settings.

Notes

If you are migrating a DHCP server in a cluster configuration, see Migrating DHCP to a Cluster Running Windows Server 2008 R2 Step-by-Step Guide (http://go.microsoft.com/fwlink/?LinkId=140512) on the Windows Server TechCenter.

Also see Migrate to DHCP Failover. DHCP failover is a new option for DHCP high availability, introduced in Windows Server 2012.

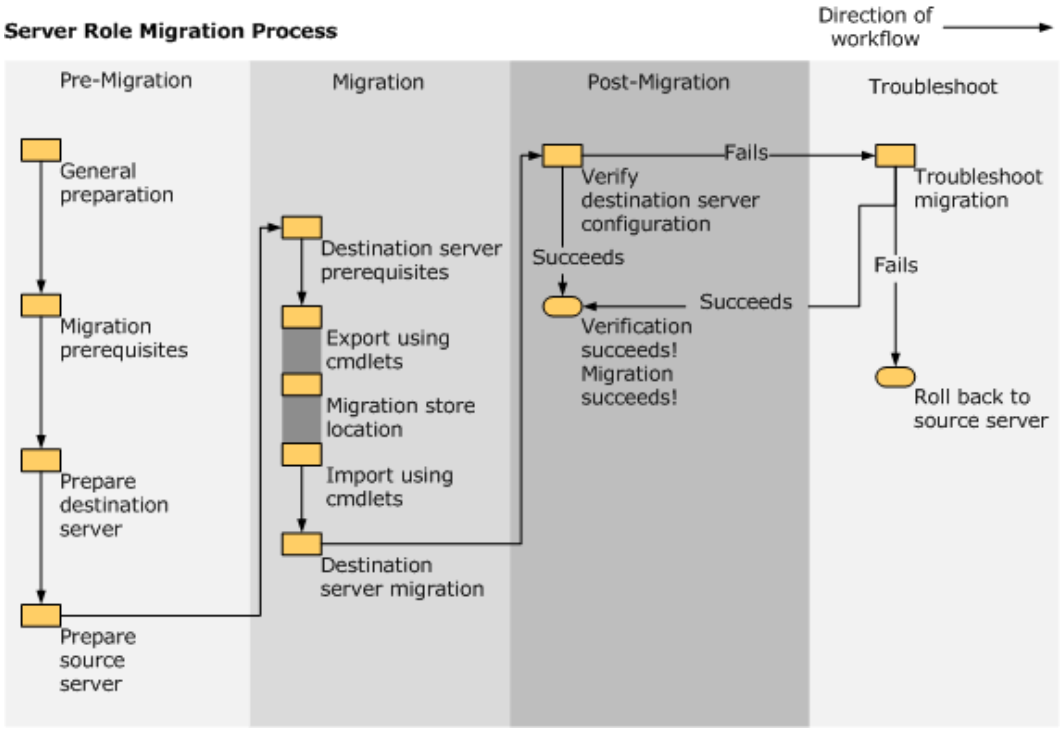
DHCP Server migration overview

DHCP Server migration is divided into the following major sections:

- DHCP Server Migration: Preparing to Migrate
- DHCP Server Migration: Migrating the DHCP Server Role
- DHCP Server Migration: Verifying the Migration
- DHCP Server Migration: Post-Migration Tasks

DHCP Server migration process

As shown in the following illustration, the pre-migration process involves the manual collection of data, followed by procedures on the destination and source servers. The migration process includes source and destination server procedures that use the **Export** and **Import** cmdlets to automatically collect, store, and then migrate server role settings. Post-migration procedures include verifying that the destination server successfully replaced the source server and then retiring or repurposing the source server. If the verification procedure indicates that the migration failed, troubleshooting begins. If troubleshooting fails, rollback instructions are provided to return to the use of the original source server.



Impact of migration on other computers in the enterprise

During migration, the source DHCP server might not be available. Therefore, client computers will not be able to obtain IP addresses from this DHCP server. We recommend that you maintain or create an auxiliary DHCP server so that client computers can obtain IP addresses while you migrate the primary DHCP server.

Be aware that if you choose to perform the migration without any auxiliary DHCP servers, all clients with valid leases must keep using those leases. If a lease for an existing client expires, that client will not be able to obtain an IP address. In addition, any new client that connects to the network will not be able to obtain an IP address when the single-source DHCP server is not available.

Permissions required to complete migration

The following permissions are required on the source server and the destination server:

- Domain administrative rights that are required to authorize DHCP Server.
- Local administrative rights are required to install or manage DHCP Server.
- Write permissions are required to the migration store location. For more information, see DHCP Server Migration: Preparing to Migrate.

Estimated duration

The migration can take two to three hours, including testing.

See also

DHCP Server Migration: Preparing to Migrate

DHCP Server Migration: Migrating the DHCP Server Role

DHCP Server Migration: Verifying the Migration

DHCP Server Migration: Post-Migration Tasks

DHCP Server Migration: Appendix A

DHCP Server Migration: Preparing to Migrate

Complete the following procedures before you migrate a DHCP Server from an x86-based or x64-based server to an x64-based server running Windows Server 2012 R2.

Migration planning

Install migration tools

Prepare the destination server

Prepare the source server

Migration planning

Membership in **Domain Administrators**, or equivalent, is the minimum required to complete these procedures. Review details about how to use the appropriate accounts and group memberships at Run a program with administrative credentials (http://go.microsoft.com/fwlink/?LinkId=131210).

To prepare for migration

- Identify your DHCP Server source and destination servers.
- Determine the domain, server name, and passwords on the source server. To identify the domain of the original server, click **Start**, right-click **Computer**, and then click **Properties**.
- If you have not already done so, install Windows Server Migration Tools on the destination and source servers as instructed in Install migration tools.
- Before migration, install all critical updates and service packs on the source server that were released before Windows Server 2012 R2. It is a recommended best practice that all current critical updates and service packs are installed on both the source and the destination servers.
- Count the number of network adapters in the source and destination servers and make

sure that they are equal in number. If the source server that is running DHCP Server has multiple network adapters and the DHCP Server service is bound to all and serving IP addresses on different subnets, the destination server that is running DHCP Server must also have multiple network adapters so that it can serve the same subnets as on the source server.

- Prepare a migration store file location. The store location must be accessible from the source server during the export and from the destination server during the import. Use a common drive that can contain all DHCP Server–related information from the source server. The storage location should be similar to the following: \\fileserver\users\username\.

Important

Before you run the **Import-SmigServerSetting**, **Export-SmigServerSetting**, or **Get-SmigServerFeature** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with domain users or groups who are members of local groups on the source server.

Before you run the **Send-SmigServerData** or **Receive-SmigServerData** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with those domain users who own files or shares that are being migrated.

Install migration tools

Install Windows Server Migration Tools on the destination and source servers. For more information, see Install, Use, and Remove Windows Server Migration Tools.

Working with Windows PowerShell cmdlets

Cmdlets (pronounced *command-lets*) are built-in commands, installed by default when you install role services and features in Windows Server 2012 R2. Throughout this guide, there are several PowerShell cmdlets that you will have to run to carry out some of the migration steps. For more information about Windows PowerShell, see Windows PowerShell Support for Windows Server on the Microsoft Web site (http://technet.microsoft.com/en-us/library/hh801904.aspx).

Except where specifically noted, cmdlets are not case-sensitive.

You can obtain detailed Help about specific syntax, parameters, and usage guidelines for any installed cmdlet by typing **Get-Help** <cmdlet name> **-full** in a wps session, in which *cmdlet name* represents the name of the cmdlet for which you want help. Add the **-Verbose** parameter to a cmdlet to display detailed information about the operation in the Windows PowerShell session.

Although most commands for DHCP Server migration are cmdlets, you can run executable files in a session by adding an ampersand (&) before the executable file name. The ampersand is the call operator.

If the executable file is not in the current directory, add the fully qualified path, as shown in the following examples. If an executable file name contains spaces enclose the file name in quotation

marks. If you are running the executable file from the current directory, precede the file name with **.**.

- Executable file that is not in the current directory: `PS C:\> & C:\Windows\System32\notepad.exe`
- Executable file that is in the current directory: `PS C:\Windows\System32> & .\notepad.exe`
- Executable file name that contains a space and is in the current directory: `PS C:\Windows\System32> & ".\executable test.exe"`

The commands in this document are provided in Windows PowerShell format. For more information, see DHCP Server Migration: Appendix A. You can run Command Prompt commands in a Windows PowerShell session by adding **cmd /C** before the command, as shown in the following example. The example shows the use of the **dir** command in wps.

```
cmd /C dir c:\*
```

Prepare the destination server

To install DHCP Server on the destination server, complete the menu-driven installation process. Complete the following procedure to prepare the destination server.

To prepare the destination server

1. Install Windows Server 2008 R2 and configure the destination server.
2. Make sure that there is sufficient disk space to store the DHCP Server database. The disk space needed varies with each installation and should be equal to or greater than the space for the DHCP Server database.
3. Add the destination server as a member server in the domain of the source server that is being replaced.
4. Verify that the destination server can resolve the names of domain users who are members of the local group during the import operation. If source and destination servers are in different domains, the destination server must be able to contact a global catalog server for the forest in which the source domain user accounts are located.
5. On a computer that is running Windows Server 2008 R2, open a wps session with elevated user rights. To do this, click **Start**, click **All Programs**, click **Accessories**, open the **Windows PowerShell** folder, right-click **Windows PowerShell**, and then click **Run as administrator**.
6. Load the Server Manager module into your wps session. To load the Server Manager module, type the following, and then press **Enter**.

   ```
   Import-Module ServerManager
   ```

   Note
   It is not mandatory that DHCP Server is installed on the destination server before you import the settings. If the role is not installed on the destination server, it will be installed automatically during the import process. However, because installation of the role during import might extend downtime, we recommend that

you install DHCP Server by using the Server Manager console on the destination server as part of your preparation for the migration.

7. On the destination server, run the following command to install DHCP Server:

   ```
   Add-WindowsFeature DHCP
   ```

   You can also install DHCP Server manually by using Server Manager. For more information, see Install Dynamic Host Configuration Protocol (DHCP) (http://go.microsoft.com/fwlink/?LinkId=128465).

   Note
   If you use the Add Roles Wizard in Server Manager to install DHCP Server on the destination server, you do not have to answer every question in the wizard. You can leave settings empty (the default) and then click **Next** through each wizard page. If you do not want to use the wizard, you can install DHCP Server by using the **Add-WindowsFeature** cmdlet, as described in this step.

8. By the end of the migration process, the destination server should have a static IP address. Although you will not change the destination server IP address now, consider the following scenarios in preparation for changing it when migration is complete.
   - If your migration scenario requires that you decommission and disconnect the source server from the network, only then can you make the IP address on the destination server the same as the IP address on the source server. The source server must be disconnected from the network or shut down so that there is no IP address conflict between the source server and destination server. However, the destination server can still serve clients that are searching for the legacy (source) server that was running DHCP Server.
   - If your migration scenario calls for continuing to run the source server on the network for other, non-DHCP purposes, you have to assign the destination server an unallocated IP address in the same subnet as the source server to avoid IP conflicts.
   - DHCP Server clients that attempt to renew an IP address lease send the renew request to the previous IP address of the DHCP server. If the source server has been decommissioned and then disconnected from the network and the new DHCP destination server is operating with a different IP address, this request initially fails because of the changed IP address. However, clients try to rediscover the IP address of the DHCP server on the network and therefore recover from this transient failure.

     Warning
     If the source server is running multiple roles, renaming the source server or changing its IP address can cause other roles that are running on the source server to fail.

9. If the DHCP Server database path does not match the default path, you must ensure that the destination server has a disk with the same drive letter as seen in source server's DHCP Server database path. For more information, see the "Known issues" section of DHCP Server Migration: Appendix A.

The destination server is now prepared for migration.

Prepare the source server

Follow these steps to prepare the source server for migration.

To prepare the source server

1. Back up the source server. The backup should be a DHCP Server-specific backup, not a Windows backup. (A Windows backup backs up the complete operating system.) You can create the DHCP Server-specific backup by using the **Netsh** command-line tool or Microsoft Management Console (MMC).
   - In the DHCP MMC tree, right-click the server node to open DHCP backup options.
   - Create the backup by using the **Netsh** command-line tool. For more information, see Netsh Commands for Dynamic Host Configuration Protocol server (http://go.microsoft.com/fwlink/?LinkId=128496).

   Note
   The Windows Server 2003 operating system does not support **Netsh**-based backup.
2. If it is running, stop the DHCP Server service. In a session that was opened as described in step 5 of To prepare the destination server, type the following, and then press **Enter**.

   ```
   Stop-Service DHCPserver
   ```
3. If the DHCP Server database path does not match the default path, make sure that the destination server has a disk with the same drive letter as in source server's DHCP Server database path. For more information, see the "Known issues" section of DHCP Server Migration: Appendix A.

You are now ready to begin DHCP Server migration, as described in DHCP Server Migration: Migrating the DHCP Server Role.

See also

Migrate DHCP Server to Windows Server 2012 R2

DHCP Server Migration: Migrating the DHCP Server Role

DHCP Server Migration: Verifying the Migration

DHCP Server Migration: Post-Migration Tasks

DHCP Server Migration: Appendix A

DHCP Server Migration: Migrating the DHCP Server Role

Complete the following procedures to migrate a DHCP Server.

Migrating DHCP Server to the destination server

Migrating DHCP Server to the destination server

Membership in **Domain Administrators** or equivalent is the minimum required to complete these procedures. Review details about how to use the appropriate accounts and group memberships at Run a program with administrative credentials (http://go.microsoft.com/fwlink/?LinkId=131210).

To migrate DHCP Server to the destination server

1. If it is not already installed, install DHCP Server on the destination server, as previously described in the “Prepare the destination server” section in DHCP Server Migration: Preparing to Migrate.
2. If it is running, stop the DHCP Server service by running the following command:

   ```
   Stop-Service DHCPserver
   ```

   If you are unsure whether the service is running, you can check its state by running the following command:

   ```
   Get-Service DHCPServer
   ```

Migrating DHCP Server from the source server

Follow these steps to migrate DHCP Server from the source server.

To migrate DHCP Server from the source server

1. Open a Windows PowerShell session with elevated user rights. To do this, click **Start**, click **All Programs**, click **Accessories**, open the **Windows PowerShell** folder, right-click **Windows PowerShell,** and then click **Run as administrator**.
2. Load Windows Server Migration Tools into your session.

   If you opened the current session by using the Windows Server Migration Tools shortcut on the **Start** menu, skip this step, and go to step 3. Only load the Windows Server Migration Tools snap-in in a session that was opened by using some other method, and into which the snap-in has not already been loaded. To load Windows Server Migration Tools, type the following, and then press **Enter**.

   ```
   Add-PSSnapin Microsoft.Windows.ServerManager.Migration
   ```
3. Collect data from the source server by running the **Export-SmigServerSetting** cmdlet as an administrator. The **Export-SmigServerSetting** cmdlet parameters can collect all source DHCP server data in a single file (Svrmig.mig). Or, the **Export-SmigServerSetting** cmdlet can be run multiple times, with each iteration using one or more parameters to collect and store data in multiple Svrmig.mig files. For more information, see DHCP Server Migration: Preparing to Migrate. Before you run this command, review the following:

- When you run the command in step 4, you are prompted to provide a password to encrypt the migration store data. You must provide this same password to import from the migration store.
- The **path** parameter can be an empty or nonempty directory. The actual data file in the directory (Svrmig.mig) is created by the **Export-SmigServerSetting** cmdlet. Therefore, the user does not have to specify a file name.
- If the path is not a shared location that the destination server can read from, you must manually copy the migration store to the destination server or a location that the destination server can access.
- If a migration store location already exists and you want to rerun the **Export-SmigServerSetting** cmdlet, you must move the Svrmig.mig file from that location and store it elsewhere, rename or first delete the migration store.
- You can perform both IP and DHCP Server migration at the same time from a Windows PowerShell prompt by using the **Export-SmigServerSetting** cmdlet combined with the **IPConfig** switch, on a single command line.
- Additional command line parameter information:
  - **-Users** and **-Group** parameters

  The **-Users** parameter must be specified only if the DHCP Administrators group includes local users. Otherwise, you can use the **-Group** parameter and all members of DHCP administrators will be migrated. Administrator group members can include domain users.

  Important
  If the source server is a domain controller, but the destination server is not, Domain Local groups are migrated as local groups, and domain users are migrated as local users.

  - The **-IPConfig** parameter collects IP information when it is used with the **Export-SmigServerSetting** cmdlet on the source server; the **-IPConfig** parameter applies settings when the **Import-SmigServerSetting** cmdlet is used on the destination server.

  If the source DHCP Server has multiple network adapters and the DHCP server service is bound to more than one network adapter and serving IP addresses on different subnets, the destination DHCP Server must also have multiple network adapters so that it can serve the same subnets as the source DHCP Server. For more information, see Migrate IP Configuration to Windows Server 2012. Because IP configuration details will be used later when importing IP configuration settings to the destination server, it is a best practice to save the IP configuration settings by using the following command:

```
IPConfig /all > IPSettings.txt
```

The **Import-SmigServerSetting** cmdlet requires you to map the source physical

address to the destination physical address.

Note
The destination server can be assigned the same static IP address as the source server, unless other roles on the source server must continue to run on it. In that case, the static IP address of the destination server can be any unallocated static IP address in the same subnet as the source server.

4. On the source server, run the **Export-SmigServerSetting** cmdlet, where <*storepath*> is the path that will contain the Svrmig.mig file after this step is completed. An example of the path is \\fileserver\users\username\dhcpstore.

```
Export-SmigServerSetting -featureID DHCP -User All -Group -
IPConfig -path <storepath> -Verbose
```

For more information about how to export IP configuration settings, see Migrate IP Configuration to Windows Server 2012.

5. On the source server, delete the DHCP authorization for the source DHCP server by running the following command, where *Server FQDN* is the fully qualified domain name (FQDN) of the DHCP server and *Server IPAddress* is the IP address of the server. The command parameters are case-sensitive and must appear exactly as shown.

```
Netsh DHCP delete server <Server FQDN> <Server IPAddress>
```

Destination server final migration steps

Return to the destination server and follow these steps to complete the migration.

1. Before you use the **Import-SmigServerSetting** cmdlet to import the DHCP server settings, be aware of the following conditions:
   - You can either use a single command line with all the parameters to import DHCP settings (as when you export data from the source server) or you can use the **Import** cmdlet multiple times to import data one parameter at a time.
   - If you decide to run the **Import-SmigServerSetting** cmdlet separately to import the IP settings, see Migrate IP Configuration to Windows Server 2012. Use the source IPSettings.txt file, referred to in step 3 of the previous procedure. You will map the source physical addresses to the destination physical addresses in step 3 of this procedure.

     Important
     If you will be importing role and IP settings separately, you should import IP settings first to avoid any IP conflicts. You can then import the DHCP role.
   - If the DHCP Administrators group includes local users, then use the **-Users** parameter combined with the **-Group** parameter to import local users into the DHCP Administrators group. If it only contains domain users, then use only the **-Group** parameter.

     Security

If the source server is a domain member server, but the destination server is a domain controller, imported local users are elevated to domain users, and imported local groups become Domain Local groups on the destination server.

- If the DHCP Server role that you are migrating has not yet been installed on the destination server, the **Import-SmigServerSetting** cmdlet will install that DHCP Server role and its dependencies, described in the next step. you might have to restart the destination computer to complete the installation after the DHCP Server role is installed by the cmdlet. Then, to complete the import operation after you restart the computer you must run the **Import-SmigServerSetting** cmdlet again along with the **-Force** parameter..

2. On the destination server, run the following command, where *<storepath>* is the available path that contains the Svrmig.mig file, *<SourcePhysicalAddress-1>* and *<SourcePhysicalAddress-2>* are comma-separated lists of the physical addresses of the source network adapter, and *<TargetPhysicalAddress-1>* and *<TargetPhysicalAddress-2>* are comma-separated lists of the physical addresses of the destination network adapter:

```
Import-SmigServerSetting -featureid DHCP -User All -Group -
IPConfig <All | Global | NIC>
-SourcePhysicalAddress <SourcePhysicalAddress-
1>,<SourcePhysicalAddress-2>
-TargetPhysicalAddress <TargetPhysicalAddress-
1>,<TargetPhysicalAddress-2>
-Force -path <storepath> -Verbose
```

The **-IPConfig** switch should be used with the value **All** in case the user wants to import all source settings. For more information, see Migrate IP Configuration to Windows Server 2012.

Important
If you import the source server IP address to the target server together with the DHCP role without disconnecting or changing the IP address of the source server, an IP address conflict will occur.

3. Run the following command to start the DHCP service:

```
Start-Service DHCPServer
```

4. Authorize the destination server. Command parameters are case-sensitive and must appear exactly as shown. On the destination server, run the following command where *Server FQDN* is the FQDN of the DHCP Server and *Server IPAddress* is the IP address of the server:

```
netsh DHCP add server <Server FQDN> <Server IPAddress>
```

Note
After authorization, the Server Manager event log might show event ID 1046. This is a known issue and is expected to occur only once. The event can be safely ignored.

When this migration is finished, client computers on the network server are served by the new x64-based destination server running Windows Server 2012 R2. The migration is complete when the destination server is ready to serve IP addresses to the network.

See also

Migrate DHCP Server to Windows Server 2012 R2

DHCP Server Migration: Preparing to Migrate

DHCP Server Migration: Verifying the Migration

DHCP Server Migration: Post-Migration Tasks

DHCP Server Migration: Appendix A

DHCP Server Migration: Verifying the Migration

When all the migration steps are complete, you can use the following procedure to verify that the DHCP server role migration was successful. If the migration failed, you can return to the previous valid configuration by following the steps in DHCP Server Migration: Post-Migration Tasks.

Verifying destination server configuration

Follow these steps to confirm that the DHCP destination server is now serving the domain.

Membership in **Domain Administrators**, or equivalent, is the minimum required to complete this procedure. Review details about how to use the appropriate accounts and group memberships at Understanding Groups: Default groups (http://go.microsoft.com/fwlink/?LinkId=83477).

To verify the configuration of the destination server

1. Make sure that the destination server is authorized by running the following command in a Windows PowerShell window:

   ```
   netsh DHCP show server
   ```

   The output of this command must contain the name of the DHCP destination server.
2. Check whether DHCP server is running on the destination server. In Task Manager, on the **Services** tab, the DHCP server status should be **Started**. You also use Task Manager to confirm that the status of the source server is **Stopped**.
3. Verify that the client computers are correctly receiving IP addresses on request by running the following commands at a command prompt on a client computer:

   ```
   ipconfig /release
   ipconfig /renew
   ```

   If the IP address of the DHCP server has not changed, you do not have to run the **ipconfig /release** command. Running **ipconfig /renew** should be sufficient.

   The output of these commands should show that the client computer was issued an IP address.
4. Use the DHCP console to verify that the scopes and other settings were migrated. To

connect to the destination server, click **Action**, click **Add Server**, and then type the IP address or host name of the DHCP server. In the console tree, expand the server node, and then expand the IPv4 and IPv6 nodes to confirm that the scopes have been migrated. Then locate the folders for the scopes and view the address range, reservations, scope options, and active leases to verify the same. You can also go to the Server Options folder and verify the migrated server options.

See also

Migrate DHCP Server to Windows Server 2012 R2

DHCP Server Migration: Preparing to Migrate

DHCP Server Migration: Migrating the DHCP Server Role

DHCP Server Migration: Post-Migration Tasks

DHCP Server Migration: Appendix A

DHCP Server Migration: Post-Migration Tasks

The post-migration tasks for the source server are optional, depending on your migration scenario.

Completing migration

Restoring DHCP in the event of migration failure

Troubleshooting cmdlet-based migration

Completing migration

Migration is complete after you have verified that the destination server, not the source server, is now serving the network. If your verification efforts demonstrate that the migration failed, see “Restoring DHCP in the event of migration failure” later in this topic.

Retiring DHCP on your source server

After you have verified the migration, you can disconnect, repurpose, or retire the source server. If the source server is running other server roles, it should be left on the network. If you do not have to use this computer, you can store it as a backup in case you ever have to revert to your previous DHCP configuration.

- If your migration scenario includes a standalone DHCP Server, then this source server was taken offline after the export file was created, as described in DHCP Server Migration: Preparing to Migrate. In this scenario, the DHCP service was interrupted from the time that it was stopped until the migration was complete on the new server, as described in DHCP Server Migration: Migrating the DHCP Server Role.

- If your migration scenario includes more than one DHCP Server in a domain, a backup or other DHCP server continues to serve IP addresses during the migration so that services to clients are never interrupted. The migration is complete on the new server when the IP address of the source server is migrated to the destination server.

Retiring your source server

After you have confirmed that the destination server is performing the functions previously handled by the source server, you can retire or repurpose the source server, depending on your needs. Follow your organization's policy regarding server decommissioning. For information about decommissioning a domain controller, see Decommissioning a Domain Controller (http://go.microsoft.com/fwlink/?LinkID=128290).

Warning
After the source server is repurposed as a member server, otherwise repurposed or retired from service, you cannot roll that server back to its previous working state.

Restoring DHCP in the event of migration failure

If the migration of DHCP Server fails, you have these options:

- If the source server has not been repurposed, an administrator can reassign the IP configuration settings, reauthorize the server, and restart the DHCP service on the original server.
- Use the backup files that were created on the source server, as described in DHCP Server Migration: Preparing to Migrate, to restore DHCP server on the original DHCP server.

Estimated time to complete a rollback

You should be able to complete a rollback in one to two hours.

Troubleshooting cmdlet-based migration

The Windows Server Migration Tools deployment log file is located at %*windir*%\Logs\SmigDeploy.log. Additional Windows Server Migration Tools log files are created at the following locations.

- %*windir*%\Logs\ServerMigration.log
- On Windows Server 2008 and Windows Server 2008 R2: %*localappdata*%\SvrMig\Log
- On Windows Server 2003: %*userprofile*%\Local Settings\Application Data\SvrMig\Log

If migration log files cannot be created in the previous locations, **ServerMigration.log** and **SmigDeploy.log** are created in %*temp*%, and other logs are created in %*windir*%\System32.

For DHCP-specific troubleshooting tips, see Troubleshooting DHCP servers on the Windows Server TechCenter (http://go.microsoft.com/fwlink/?LinkId=128533). Although these tips are written for Windows Server 2003, they also address common issues that apply to later versions of the operating system.

If a migration cmdlet fails, and the wps session closes unexpectedly with an access violation error message, look for a message similar to the following example in the %*localappdata*%\SvrMig\Logs\setuperr.log file.FatalError [0x090001] PANTHR Exception (code 0xC0000005: ACCESS_VIOLATION) occurred at 0x000007FEEDE9E050 in C:\Windows\system32\migwiz\unbcl.dll (+000000000008E050). Minidump attached (317793 bytes).

This failure occurs when the server cannot contact domain controllers that are associated with domain users or groups who are members of local groups, or who have rights to files or shares that are being migrated. When this happens, each domain user or group is displayed in the GUI as an unresolved security identifier (SID). An example of a SID is **S-1-5-21-1579938362-1064596589-3161144252-1006**.

To prevent this problem, verify that required domain controllers or global catalog servers are running, and that network connectivity allows communication between both source and destination servers and required domain controllers or global catalog servers. Then, run the cmdlets again.

If connections between either the source or destination servers and the domain controllers or global catalog servers cannot be restored, do the following.

1. Before you run **Export-SmigServerSetting**, **Import-SmigServerSetting** or **Get-SmigServerFeature** again, remove all unresolved domain users or groups who are members of local groups from the server on which you are running the cmdlet.
2. Before you run **Send-SmigServerData** or **Receive-SmigServerData** again, remove all unresolved domain users or groups who have user rights to files, folders, or shares on the migration source server.

Viewing the content of Windows Server Migration Tools result objects

All Windows Server Migration Tools cmdlets return results as objects. You can save result objects, and query them for more information about settings and data that were migrated. You can also use result objects as input for other wps commands and scripts.

Result object descriptions

The Windows Server Migration Tools **Import-SmigServerSetting** and **Export-SmigServerSetting** cmdlets return results in a list of **MigrationResult** objects. Each **MigrationResult** object contains information about the data or setting that the cmdlet processes, the result of the operation, and any related error or warning messages. The following table describes the properties of a **MigrationResult** object.

Property name	Type	Definition
ItemType	Enum	The type of item being migrated.

Property name	Type	Definition
		Values include **General**, **WindowsFeatureInstallation**, **WindowsFeature**, and **OSSetting**.
ID	String	The ID of the migrated item. Examples of values include **Local User**, **Local Group**, and **DHCP**.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise, **False** is displayed.
DetailsList	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.

Send-SmigServerData and **Receive-SmigServerData** cmdlets return results in a list of **MigrationDataResult** objects. Each **MigrationDataResult** object contains information about the data or share that the cmdlet processes, the result of the operation, any error or warning messages, and other related information. The following table describes the properties of a **MigrationDataResult** object.

Property name	Type	Definition
ItemType	Enum	The type of migrated item. Values include **File**, **Folder**, **Share**, and **Encrypted File**.
SourceLocation	String	The source location of the item, shown as a path.
DestinationLocation	String	The destination location of the item, shown as a path.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise, **False** is displayed.
Size	Integer	The item size, in bytes.
ErrorDetails	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.
Error	Enum	Errors enumeration for errors that occurred.
WarningMessageList	List <String>	A list of warning messages.

The following table describes the properties of objects within the **MigrationResultDetails** object that are common to both **MigrationResult** and **MigrationDataResult** objects.

Property name	Type	Definition
FeatureId	String	The name of the migration setting that is related to the item. Examples of values include **IPConfig** and **DNS**. This property is empty for data migration.
Messages	List <String>	A list of detailed event messages.
DetailCode	Integer	The error or warning code associated with each event message.
Severity	Enum	The severity of an event, if events occurred. Examples of values include **Information**, **Error**, and **Warning**.
Title	String	Title of the result object. Examples of values include the network adapter physical address for IP configuration, or user name for local user migration.

Examples

The following examples show how to store the list of the result objects in a variable, and then use the variable in a query to return the content of result objects after migration is complete.

To store a list of result objects as a variable for queries

1. To run a cmdlet and save the result in variable, type a command in the following format, and then press **Enter**.

   ```
   $VariableName = $(Cmdlet)
   ```

   The following is an example.

   ```
   $ImportResult = $(Import-SmigServerSetting -FeatureId DHCP -
   User all -Group -Path D:\rmt\DemoStore -force -Verbose)
   ```

This command runs the **Import-SmigServerSetting** cmdlet with several parameters specified, and then saves result objects in the variable **ImportResult**.

2. After the **Import-SmigServerSetting** cmdlet has completed its operations, return the information that is contained in the result object by typing a command in the following format, and then pressing **Enter**.

   ```
   $VariableName
   ```

   In the following example, the variable is named **ImportResult**.

   ```
   $ImportResult
   ```

   This command returns information that was contained in the result objects that were returned by **Import-SmigServerSetting** in the example shown in step 1. The following is an example of the output that is displayed by calling the **ImportResult** variable.

   ```
               ItemType  ID                                  Success
   DetailsList
               --------  --                                  -------
   -----------
             OSSetting  Local User                            True
   {Local User, Loc...
             OSSetting  Local Group                           True
   {Local Group, Lo...
        WindowsFeature  DHCP                                  True
   {}
   ```

   Each line of the previous sample is a migration result for an item that was migrated by using the **Import-SmigServerSetting** cmdlet. The column heading names are properties of **MigrationResult** objects. You can incorporate these properties into another command to return more detail about result objects, as shown by examples in step 3 and forward.

3. To display a specific property for all result objects in the list, type a command in the following format, and then press **Enter**.

   ```
   $<VariableName>| Select-Object -ExpandProperty <PropertyName>
   ```

   The following is an example.

   ```
   $importResult | Select-Object -ExpandProperty DetailsList
   ```

4. You can run more advanced queries to analyze result objects by using wps cmdlets. The following are examples.

   - The following command returns only those details of result objects that use the ID **Local User**.

     ```
     $ImportResult | Where-Object { $_.ID -eq "Local User" } |
     Select-Object -ExpandProperty DetailsList
     ```

   - The following command returns only those details of result objects that use an ID of

Local User that have a message severity equal to **Warning**.

```
$ImportResult | Where-Object { $_.ID -eq "Local User" } |
Select-Object -ExpandProperty DetailsList | ForEach-Object {
if ($_.Severity -eq "Warning") {$_} }
```

- The following command returns only the details of result objects that use an ID of **Local Group** that also have the title **Remote Desktop Users**.

```
$ImportResult | Where-Object { $_.ID -eq "Local Group" } |
Select-Object -ExpandProperty DetailsList | ForEach-Object {
if ($_.Title -eq "Remote DesktopUsers") {$_} }
```

More information about querying results

For more information about the cmdlets that are used in the previous examples, see the following additional resources.

- Where-Object on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134853).
- Select-Object on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134858).
- ForEach-Object on the Microsoft Script Center Web site (http://www.microsoft.com/technet/scriptcenter/topics/msh/cmdlets/foreach-object.mspx)

See also

Migrate DHCP Server to Windows Server 2012 R2

DHCP Server Migration: Preparing to Migrate

DHCP Server Migration: Migrating the DHCP Server Role

DHCP Server Migration: Verifying the Migration

DHCP Server Migration: Appendix A

DHCP Server Migration: Appendix A

Migration tools

Migration tools are provided in Windows Server® 2012 R2. The tools for earlier versions of the Windows operating system are also available in Windows Server 2012 R2.

Follow these steps to access the tools on the destination server:

1. Open Server Manager.
2. Click **Action**, and then select **Add Features**. The Add Features Wizard opens.

3. On the **Select Features** page, from the **Features** list, select **Windows Server Migration Tools**, and then click **Next**.
4. Complete the steps in the wizard, and then click **Close**.

The previous steps do not work for Server Core installations. To install the migration tools on a Server Core installation, see the Install Windows Server Migration Tools topic in Install, Use, and Remove Windows Server Migration Tools.

Installing and using Windows PowerShell with migration cmdlets

To access, download, and install migration tools (the migration toolkit), any role-specific tools, and Windows PowerShell, see Install, Use, and Remove Windows Server Migration Tools.

Known issues

If the DHCP installation on the source server has a database path that varies from the default, before you perform the import, provide the destination server with a volume with the same drive letter on which the DHCP server database exists on the source server. For example, if the DHCP server database on the source server is located on D:\, then the destination server should have a volume with the driver letter D.

If you cannot match the volume on the destination server that has the same driver letter as that shown for the source DHCP server database, then the DHCP database path on the source server must be changed back to the default path (%systemroot%\system32\dhcp) before you start the migration.

See also

Migrate DHCP Server to Windows Server 2012 R2

DHCP Server Migration: Preparing to Migrate

DHCP Server Migration: Migrating the DHCP Server Role

DHCP Server Migration: Verifying the Migration

DHCP Server Migration: Post-Migration Tasks

Migrate Hyper-V to Windows Server 2012 R2 from Windows Server 2012

With Hyper-V, you can create a virtualized server computing environment by using a technology that is part of Windows. This guide provides information and instructions about migrating the Hyper-V role that include virtual machines, data, and operating system settings from the source server running Hyper-V in Windows Server 2012 to the destination server that is running the Windows Server 2012 R2 operating system.

About this guide

This guide describes how to migrate the Hyper-V role by providing preparation, migration, and verification steps.

Migration documentation and tools facilitate the migration of server role settings and data from an existing source server to a destination server that is running Windows Server 2012 R2. By using the tools that are described in this guide, you can simplify the migration process, reduce migration time, increase the accuracy of the migration process, and help eliminate possible conflicts that might otherwise occur during the migration process.

In addition to the migration options that are described in this topic, Virtual Machine Manager in Microsoft System Center 2012 R2 can facilitate and automate a considerable part of the migration process. For more information about Virtual Machine Manager, see Virtual Machine Manager.

Target audience

This document is intended for information technology (IT) professionals who are responsible for operating and deploying Hyper-V in a managed environment.

What this guide does not provide

- Migration of Hyper-V from one server that runs Windows Server 2008 R2 to another server that runs Windows Server 2012 R2.
- Instructions for migrating more than one server role at one time.
- Migration of Hyper-V from one server that runs Windows Server 2012 R2 to another server that runs Windows Server 2012 R2. Instead, this process is supported by Hyper-V management tools and features. The general process is as follows:
    a. Determine whether to use export and import or live migration to move the virtual machines:
        - Export and import can be used in either a workgroup or a domain environment. In Hyper-V running on Windows Server 2012 R2, you can now export a running virtual machine.
        - Live migration requires a domain environment and some additional configuration, but the virtual machine is running throughout the move process.
    b. Add the Hyper-V role to the destination server. You can configure the default storage locations and live migration when you add the role.
    c. Configure virtual switches, and optionally, other networking features on the destination server. Management tools include the Windows PowerShell cmdlets New-VMSwitch and Set-VMSwitch, and the Hyper-V Virtual Switch Manager in the Hyper-V Manager Console.
    d. Move the virtual machines by using export and import or live migration. Management tools include the Windows PowerShell cmdlets Export-VM, Import-VM, and Move-VM and the **Export**, **Import**, and **Move** menu commands in the Hyper-V Manager Console.

Supported migration scenarios

This guide provides you with instructions to migrate a server that is running Hyper-V in Windows Server 2012 to a server that is running Windows Server 2012 R2. This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, we recommend that you design a custom migration procedure that is specific to your server environment and is based on the information in other role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server 2012 R2.

Migration dependencies

The Hyper-V role does not depend on any other roles. As a best practice, we recommend that no other roles are installed on a server running Hyper-V.

Migration scenarios that are not supported

The following migration scenarios are not supported:

- Virtual machine configuration under one of the following conditions:
  - When the number of virtual processors that are configured for the virtual machine is greater than the number of logical processors on the destination server.
  - When the memory that is configured for a virtual machine is greater than the available memory on the destination server.

Overview of migration process for this role

Hyper-V role migration involves moving the virtual machines, virtual networks, and all the associated settings from one physical computer to another physical computer in the enterprise. The process supports moving from a server running Hyper-V in Windows Server 2012 to a server running Hyper-V in Windows Server 2012 R2. The Hyper-V role does not depend on any other roles.

The migration tools include Windows PowerShell cmdlets that you can use to perform some of the tasks that are required to migrate the Hyper-V role and script or to automate the migration process.

In previous versions of Hyper-V, you were required to shut down a virtual machine before you moved it to a new server. If the move was performed correctly, downtime was limited, but still, there was downtime. A new feature in Windows Server 2012 R2, cross-version live migration, supports moving a running virtual machine from Windows Server 2012 to Windows Server 2012 R2. The Windows PowerShell Export-VM cmdlet captures the majority of the Hyper-V settings that are required to perform a successful migration, which includes the virtual machine configurations, virtual networks, and virtual hard disks. Now you can decide how to move virtual machines to Windows Server 2012 R2, where in the past, your options where limited.

The following options are available to move a virtual machine:

- In-place upgrade
- Cross-version live migration
- Export and Import
- Copy Cluster Role Wizard

For additional information about each option, see Hyper-V: Migration Options.

This guide explains the migration process for the following three main scenarios:

- Hyper-V: Stand-alone Migration
- Hyper-V Cluster Using Separate Scale-Out File Server Migration
- Hyper-V Cluster Using Cluster Shared Volumes (CSV) Migration

Estimated duration

The length of time it takes to migrate the Hyper-V role depends on the size of the data to be transferred and on the tools that are used. Of the various types of files to be transferred, the virtual hard disk (VHD), .vhd and .vhdx files, have the largest file sizes from a few gigabytes to many gigabytes in size. The length of time that is required for migration is largely affected by the size of the VHD files and by the network bandwidth.

Additional references

Windows Server Migration forum

Hyper-V: Migration Options

When you migrate virtual machines from the Windows Server 2012 operating system to the Windows Server 2012 R2 operating system, you have various options on how to migrate your virtual machines. You now can select the migration options that meet the requirements of your environment.

Hyper-V migration options

Depending on your requirements and service level agreements that must be maintained, you can use one migration option or a combination of migration options. For example, if you have virtual machines that either must be running all the time or that only have a short maintenance period for you to shut them down, you might select to use cross-version live migration to move the virtual machines from Windows Server 2012 to Windows Server 2012 R2. For other virtual machines that are not as critical, have a long maintenance period, or might take too long to move by using live migration, you can use the Copy Cluster Roles Wizard or **Export / Import**, which depends on your environment.

The following table shows the benefits and disadvantages of the various migration options.

Migration option	Benefits	Disadvantages
In-place upgrade	• No new hardware required.	• Virtual machines must be shut off during the upgrade.
Cross-version live migration	• Virtual machines continue running during migration. • If the virtual hard disk is stored on a Scale-out File Server share that is accessible from both servers, the virtual hard disks do not have to be copied. • Moves virtual machines from one Hyper-V cluster to another cluster without any downtime. • Migrates individual virtual machines that are part of the Hyper-V cluster.	• Requires additional hardware or extra capacity in the existing cluster to create the destination cluster. • The amount of time it takes to migrate a virtual machine depends on various factors, for example, the size of memory that is configured for the virtual machine and the network configuration. If the virtual hard disks are not stored on a Scale-out File Server, additional time is required to move the virtual hard disk. • The virtual machine must be removed from the existing cluster before it is moved to the new cluster. After the virtual machine has successfully moved to new cluster node, high availability is added to the virtual machine. During the move process, the virtual machine is not protected by the cluster services.
Copy Cluster Roles Wizard	• Easily migrates a Hyper-V cluster from Windows Server 2012 to Windows Server 2012 R2. • Tests the Copy Cluster Roles process without affecting production. • Reverses the process if you encounter any issues. • Copies roles on test	• All virtual machines on the same Shared Clustered Volume are migrated at the same time. • Virtual machines must be shut down for a short period of time. • The Copy Cluster Wizard does not copy the Hyper-V replication settings when it

Migration option	Benefits	Disadvantages
	clusters to production clusters.	copies a virtual machine to a new failover cluster. Hyper-V replication must be re-enabled on the virtual machine after it is copied. For the **Initial Replication Method**, select **Use an existing virtual machine on the Replica server as the initial copy**. • Requires additional hardware.
Export / Import	• Migrates one virtual machine at a time. • Controls the method of copying the virtual machine to the new server.	• The virtual machine is shut down during the Export / Import process. • Requires additional hardware. • The Import of a virtual machine removes any Hyper-V Replica configuration settings for a virtual machine. Hyper-V replication must be re-enabled on the virtual machine after it is imported. For the **Initial Replication Method**, select **Use an existing virtual machine on the Replica server as the initial copy**.

The following table shows the available options to use in different deployments of Hyper-V.

Scenario / Migration option	In-place upgrade	Export / Import	Cross-version live migration	Copy Cluster Roles Wizard
Standalone host	Yes	Yes	Yes	No
Hyper-V Cluster with Cluster Shared Volumes (CSV)	No	Yes	Yes, the virtual machine must be removed from the cluster first, and the virtual hard	Yes

Scenario / Migration option	In-place upgrade	Export / Import	Cross-version live migration	Copy Cluster Roles Wizard
			disks must be copied as part of the live migration.	
Hyper-V Cluster with a separate Scale-out File Server for storage	No	Yes	Yes, the virtual machine must be removed from the cluster first.	Yes

Important
When Hyper-V Replica is enabled, we recommend that you migrate the virtual machines on the Replica site first, and then migrate the primary site.

Cross-version live migration

The upgrade to a new version of Windows Server no longer requires downtime of the virtual machines. In Windows Server 2012 R2, Hyper-V live migration has been updated to support the migration of virtual machines in Windows Server 2012 to Windows Server 2012 R2. If the virtual hard disk files are stored on an SMB 3.0 share that is accessible from both the source and destination server, you only must move the virtual machine configuration and memory files, but not the virtual hard disk files. If the virtual hard disk files are not stored on an SMB 3.0 share, or if the share is not accessible to the destination server, you can use the Shared Nothing Live Migration to migrate the virtual hard disk files, virtual machine configuration files, and the running virtual machine with no downtime.

Hyper-V Replica

Hyper-V Replica was introduced in Windows Server 2012 and provides asynchronous replication of Hyper-V virtual machines between two hosting servers. It is simple to configure and does not require either shared storage or any particular storage hardware. Any server workload that can be virtualized in Hyper-V can be replicated. Replication works over any ordinary IP-based network, and the replicated data can be encrypted during transmission. Hyper-V Replica works with standalone servers, failover clusters, or a mixture of both. The servers can be physically co-located or widely separated geographically. The physical servers do not have to be in the same domain or even be joined to any domain at all.

Consider the following factors for the move from Windows Server 2012 to Windows Server 2012 R2 when you use Hyper-V Replica:

- You must upgrade the Replica server first. A Windows Server 2012 R2 Replica server can accept replication from a primary server that runs Windows Server 2012. However, a Windows Server 2012 Replica server cannot accept replication from a primary server that runs Windows Server 2012 R2.

- When you upgrade the Replica server, consider the following factors:
  - When you perform an in-place upgrade on the Replica server, the post-upgrade of the Replica server to Windows Server 2012 R2 replication continues from the primary server that runs Windows Server 2012 at the default replication frequency of 5 minutes.
  - When you move the virtual machines to a new server that runs Windows Server 2012 R2, you must update the virtual machine replication settings on the primary server with the name of the new Replica server or Hyper-V Replica Broker. Until the Replica server name is updated, replication does not resume.
  - You can start to use new Hyper-V Replica features, such as extended replication from the Replica server.
  - You can add new virtual machines to the primary server that runs Windows Server 2012 and start replication to a Replica server that runs Windows Server 2012 R2.
  - In case of emergency, you can fail over your virtual machines from the primary server to the Replica server. You cannot start reverse replication because replication is not supported from Windows Server 2012 R2 to Windows Server 2012.

    Note
    At this point, the virtual machine is no longer protected by Hyper-V Replica. You can configure extended replication by using another server running Hyper-V in Windows Server 2012 R2. After the primary server has been upgraded to Windows Server 2012 R2, you can reverse replication back to the primary server. When you reverse replication, you can select to use an existing virtual machine to limit the amount of replication that must be transmitted over the network.
  - Migration cancels a test failover that currently runs for a Replica virtual machine and deletes the test virtual machine.
- When you upgrade the primary server, consider the following factors:
  - The Replica server has already been upgraded to Windows Server 2012 R2. If the Replica server has not been upgraded to Windows Server 2012 R2, replication fails until the Replica server is upgraded to Windows Server 2012 R2.
  - Replication continues at the default frequency of 5 minutes, which can be modified if it is required.
  - When you use certificate-based authentication for Hyper-V Replica, after you move the primary virtual machine to a new server, you must update the certificate thumbprint for the virtual machine.

    You can update the certificate thumbprint in the Hyper-V Manager Console by editing the Replication settings of the virtual machine, or you can use the following Windows PowerShell cmdlet, **Set-VMReplication**.

    ```
    Set-VMReplication –VMName <virtual machine name. –
    CertificateThumbprint <thumbprint>
    ```

See also

Hyper-V Replica Overview

Virtual Machine Live Migration Overview

Deploy Hyper-V over SMB

Failover Clustering Overview

Migrating Clustered Services and Applications to Windows Server 2012

Hyper-V: Stand-alone Migration

This scenario describes how to migrate a single server running the Hyper-V role in Windows Server 2012 to Windows Server 2012 R2.

Migration options

When you migrate a single server, you have the following migration options available:

- In-place upgrade
- **Cross-version live migration**
- Export and Import (not covered in this guide)

In-place upgrade

This scenario describes how to use the existing hardware that runs the Windows Server 2012 operating system and to perform an in-place upgrade of the operating system to Windows Server 2012 R2. This scenario does not require any additional hardware; however, during the upgrade process, all of the virtual machines must be turned off or be in a saved state.

Notes

- We recommend that you shut down or turn off all of the virtual machines before you upgrade. Virtual machines can be in a saved state during the upgrade, but we do not recommend it. You receive a warning during the upgrade process if any of the virtual machines are in a saved state.
- Before you run an in-place upgrade, we recommend that you back up the management operating system and the virtual machines.

Perform an in-place upgrade

Use the following steps to perform an in-place upgrade.

Note

If Hyper-V Replica has been enabled on any of the virtual machines, we recommend that you upgrade the Replica server first. During the upgrade of the Replica server, the primary server continues to send updates to the Replica server, and you might see warning messages about the health of the replication. After the Replica server has successfully upgraded, the replication should continue normally.

- Log on to the server by using a user account with local Administrator rights.
- Insert media for Windows Server 2012 R2 and run **Setup.exe** if the installation program did not start automatically.
- Review the upgrade report and fix any blocking warning messages.
- After the server running Hyper-V has restarted, confirm that the server running Hyper-V was successfully upgraded.
- Install the latest updates.
- Start each of the virtual machines that were running before the upgrade.
- Confirm that each virtual machine operates as expected.
- Upgrade the integration services for each virtual machine. A restart might be required to complete the integration services update.

Cross-version live migration

The upgrade to a new version of the Windows Server operating system no longer requires downtime of the virtual machines. In Windows Server 2012 R2, live migration has been updated to support the migration of Hyper-V virtual machines in Windows Server 2012 to Hyper-V in Windows Server 2012 R2. If the virtual hard disk (VHD) files are stored on an SMB 3.0 file share, you must only move the virtual machine, but not the storage.

This scenario requires additional hardware for a destination server. Ensure that the destination server has the capacity to run the virtual machines that you are currently running and has room for future expansion.

Use the following steps to move a virtual machine from Windows Server 2012 to Windows Server 2012 R2.

Prepare the new server hardware

1. Install Windows Server 2012 R2 on the new server hardware.
2. Install the Hyper-V role on the server.
3. Configure the following Hyper-V settings, for example:
    - The default location for virtual hard disks and virtual machine configuration files.
    - NUMA settings.
    - Live migration settings. Even if live migration was not previously configured, you must enable and configure live migration on both servers.
    - Replication settings if Hyper-V Replica is used. If certificate-based authentication is configured, an appropriate certificate must be installed on the new server.
    - Virtual switches.
    - **Hyper-V Administrators** local group membership.
4. Install the latest updates.

Move a virtual machine from Hyper-V in Windows Server 2012 to Windows Server 2012 R2

In this section, you move a virtual machine from Hyper-V in Windows Server 2012 to Hyper-V in Windows Server 2012 R2.

Perform this procedure on the source server running Hyper-V in Windows Server 2012.

To move the virtual machine to Hyper-V in Windows Server 2012 R2

1. On the **source server** running Hyper-V in Windows Server 2012, open the Hyper-V Manager Console, and then select the virtual machine that you want to move.
2. From the **Actions** pane, click **Move**. This action opens the **Move Wizard**.
   a. On the **Choose Move Type** page, select **Move the virtual machine**.
   b. On the **Specify Destination Computer**, specify the name or server that is running Windows Server 2012 R2.
   c. On the **Choose Move Options** page, select **Move only the virtual machine**.

   You can also use the Windows PowerShell cmdlet **Move-VM**. The following example shows a virtual machine *test VM* that is moved to a remote computer *NewServer* where the virtual machine is stored on an SMB share.

   ```
   PS C:\> Move-VM –Name "Test VM" -DestinationHost NewServer
   ```

Modify the Hyper-V Replica settings

Note
Perform the following procedure on the primary server after moving a virtual machine on the Replica server.

[Optional] To modify Hyper-V Replica settings

1. On the primary server, open the Hyper-V Manager Console, and then select the virtual machine whose Replica virtual machine was just moved.
2. Right-click the virtual machine to select **Settings**.
3. Select **Replication** and update the value for **Replica server** with the name of the destination Replica server.
4. Confirm that replication has successfully started.

   You can also use the **Set-VMReplication** cmdlet to update the name of the Replica server.

Verify that the virtual machine runs correctly

This procedure describes how to confirm that the virtual machine that was moved runs correctly on the destination server running Hyper-V in Windows Server 2012 R2.

Note

Skip this step when you move a virtual machine on a Replica server. Replica virtual machines are in an off state until the virtual machine is failed over by the administrator.

To verify that virtual machine runs correctly

1. Open the Hyper-V Manager Console on the destination server.
2. Verify that the virtual machine is running. If the virtual machine is not running, attempt to start it. If the virtual machine does not start, check the event log to see why it failed to start.
3. [Optional] - Run some basic operations that change the state of the virtual machine.
4. Run the necessary application-specific tests to ensure that the application on the virtual machine can provide the same service levels as it provided before the virtual machine was migrated. Although the virtual machine was moved while it was running the services that the virtual machine provides, the services should not have been interrupted.
5. Verify that you can connect to the virtual machine by using **Remote Desktop** or **Virtual Machine Connection**.
6. Upgrade the integration services on the virtual machine. Because the virtual machine was never shut down during the migration, you can update the integration services silently without a restart. The update occurs the next time that the virtual machine is restarted during its scheduled maintenance period.
    a. Modify the settings of the virtual machine and specify the following media to be used for the CD/DVD drive, *%Systemroot%\System32\Vmguest.iso*.
    b. Run the following command from an elevated command prompt in the virtual machine:
        i. For 64-bit Windows Server operating systems, *drive:\Support\Amd64\Setup.exe /quiet /norestart*
        ii. For 32-bit Windows Server operating systems, *drive:\Support\X86\Setup.exe /quiet /norestart*

See also

Virtual Machine Live Migration Overview

Configure Live Migration and Migrating Virtual Machines without Failover Clustering

Install and Deploy Windows Server 2012 R2 and Windows Server 2012

Hyper-V Replica Overview

Hyper-V: Hyper-V Cluster Migration

Hyper-V Cluster Migrations

The following sections describe how to migrate a Hyper-V cluster running in Windows Server 2012 to a Hyper-V cluster running in Windows Server 2012 R2. Depending on the configuration of the storage that the cluster uses, the following migration options are available:

- Hyper-V Cluster Using Separate Scale-Out File Server Migration
- Hyper-V Cluster Using Cluster Shared Volumes (CSV) Migration

Hyper-V Cluster Using Separate Scale-Out File Server Migration

This scenario describes how to migrate virtual machines from a Hyper-V cluster by using a separate Scale-out File Server that runs on the Windows Server 2012 operating system to the Windows Server 2012 R2 operating system. In this scenario, you move the virtual machines that run on a Hyper-V cluster from Windows Server 2012 to a Hyper-V cluster that runs on Windows Server 2012 R2.

Depending on your requirements, you have two main options to move your virtual machines from a Hyper-V cluster that runs on Windows Server 2012 to a Hyper-V cluster that runs on Windows Server 2012 R2. For information about the advantages or disadvantages for each option, see Hyper-V: Migration Options.

- Cross-version live migration
- Copy Cluster Roles Wizard

Note

Because the Hyper-V cluster and Scale-Out File Server run on separate clusters, you can upgrade each cluster independently of the other. This topic only describes how to move the virtual machines to a new Hyper-V cluster, while the storage remains on the same Scale-out File Server.

Cross-version live migration

With cross-version live migration, you can move a running virtual machine from a server running Hyper-V in Windows Server 2012 to a server running Hyper-V in Windows Server 2012 R2.

Cross-version live migration does not support moving a virtual machine to a down-level version of Hyper-V.

Important

To use cross-version live migration, the virtual machine must be removed from the cluster. The virtual machine is then moved to one of the servers in the new cluster. After the virtual machine has successfully moved to the new server, the virtual machine is configured for high availability on the new cluster. During this process, the virtual machine is not highly available.

In this option, you must create a new Windows Server 2012 R2 Hyper-V cluster. You have various options to create the new Hyper-V cluster:

- Evict two nodes from the existing cluster and create a new two-node cluster.
- Evict one node from the existing cluster and use new hardware to create a new two-node cluster.
- Use two new servers to create the new cluster.
- Evict one node from the existing cluster and create the new one-node cluster. Until a second node is added, the virtual machines that are moved to the new cluster are not highly available.
- Use one new server to create a new cluster with one node. Until a second node is added, the virtual machines that are moved to the new cluster are not highly available.

Now that the Windows Server 2012 R2 Hyper-V cluster is running, you can move the virtual machines that are currently running on one of the nodes to the new cluster.

Important

The folder that Hyper-V uses to store virtual machine data requires specific permissions to access the Server Message Block (SMB) file share. You must ensure that the Hyper-V computer accounts, the SYSTEM account, and all Hyper-V administrators have full control permissions. For more information about deploying Hyper-V over SMB, see Deploy Hyper-V over SMB.

Cross-version live migration scenario

The following migration scenario is based on the following factors:

The following table lists the servers at the start of the migration.

For the old cluster, there are three servers running Hyper-V with eight highly available virtual machines.

A two-node Windows Server 2012 R2 cluster has been prepared and is ready to receive the virtual machines from the server running Hyper-V in Windows Server 2012.

Note

You must enable and configure Hyper-V live migration on all of the servers running Hyper-V.

Name	Windows Server operating system	Cluster name
HVSRV1	Windows Server 2012	HVHA2012
HVSRV2	Windows Server 2012	HVHA2012
HVSRV3	Windows Server 2012	HVHA2012
HVR2_1	Windows Server 2012 R2	HVHAR2
HVR2_2	Windows Server 2012 R2	HVHAR2

The following table lists the virtual machines that are currently running on each of the nodes at the start of the migration.

Virtual machine name	Server running Hyper-V
Testvm_1	HVSRV1
Testvm_2	HVSRV1
Testvm_3	HVSRV2
Testvm_4	HVSRV2
Testvm_5	HVSRV3
Testvm_6	HVSRV3

The following are the general steps to move the virtual machines from the HVHA2012 cluster to the new HVHAR2 cluster.

1. Create a new Hyper-V cluster by using a separate Scale-Out File Server.
2. Move all of the virtual machines from HVSRV1 to HVR2_1.
    a. On the HVHA2012 cluster, remove one virtual machine that runs on HVSRV1. The virtual machine still runs on Hyper-V, but it is no longer highly available.
    b. From the Hyper-V Manager, move the virtual machine that was removed in step 2a to the HVR2_1 server. Because there is shared storage, you must move only the virtual machine, not the storage.
    c. On the HVHAR2 cluster, add the virtual machine that was moved to HVR2_1 in step 2b to the new cluster. The virtual machine is now highly available.
    d. Repeat steps 2a-2c until all of the virtual machines from HVSRV1 node have been moved to the new R2Cluster cluster.
3. Evict HVSRV1 from the HVHA2012 cluster.

4. Install Windows Server 2012 R2 on HVSRV1, and then join the server to the HVHAR2 cluster.
5. Repeat steps 2-4 for HVSRV2 and HVSRV3.

Note

When you get down to the last two servers in the Windows Server 2012 cluster, if you evict another node, you have a single-node cluster, and the remaining virtual machines are no longer highly available. If there is enough capacity on the new cluster to run the remaining virtual machines, move all of the remaining virtual machines to the new cluster, and then evict the last two servers at the same time.

To create a Windows Server 2012 R2 Hyper-V cluster

1. Create a new Hyper-V cluster that is connected to the same Scale-out File Server, to which Windows Server 2012 is connected.

   Configure live migration on the new servers running Hyper-V and the old servers running Hyper-V.

   If Hyper-V replication is enabled, configure the Hyper-V Replica Broker on the new cluster, HVHAR2.

   For more information about creating a Hyper-V cluster, see Deploy a Hyper-V Cluster.

   Important

   The folder that Hyper-V uses to store virtual machine data requires specific permissions to access the SMB file share. You must ensure that the Hyper-V computer accounts, the SYSTEM account, and all Hyper-V administrators have full control permissions. For more information about deploying Hyper-V over SMB, see Deploy Hyper-V over SMB.

To move the virtual machines to the new cluster

1. On HVSRV1, open the Failover Cluster Manager and select **Nodes**.
2. Right-click the HVSRV1 node, and then select **Pause** and **Do Not Drain Roles**.
3. In the information pane for the HVSRV1, select **Roles** to see the virtual machines on the node.
4. Right-click the **Testvm_1** virtual machine, and then select **Remove**. The virtual machine is still running but is no longer highly available.
5. On HVSRV1, open the Hyper-V Manager.
6. Right-click the **Testvm_1** virtual machine, and then select **Move**.
   a. On the **Choose Move Type** page, select **Move the virtual machine**.
   b. On the **Specify Destination Computer** tab, specify the name or server running Windows Server 2012 R2, HVR2_1. Do not enter the name of the new cluster, HVHAR2.
   c. On the **Choose Move Options** page, select **Move only the virtual machine**.

You can also use the Windows PowerShell cmdlet **Move-VM**. In the following example, a virtual machine *test VM* was moved to a remote computer *NewServer* where the virtual machine is stored on an SMB share.

```
PS C:\> Move-VM –Name "Test VM" –DestinationServer NewServer
```

7. After the move finishes successfully, on HVR2_1, open Hyper-V Manager and confirm that the virtual machine runs correctly.
8. On HVR2_1, open the Failover Cluster Manager, and then select **Roles**.
9. In the **Actions** pane, select **Configure Roles** to open the **High Availability Wizard**.

   On the **Select Role** page, select **Virtual Machine**.

   On the **Select Virtual Machine** page, select **Testvm_1**.
10. The virtual machine is now highly available.
11. Update the integration services on **Testvm_1**. Because the virtual machine was never shut down during the migration, you can update the integration services silently without a restart. The update occurs the next time that the virtual machine is restarted during its scheduled maintenance period.
    a. Modify the settings of the virtual machine and specify the following media to be used for the CD/DVD drive, *%Systemroot%\System32\Vmguest.iso*.
    b. Run the following command from an elevated command prompt in the virtual machine:
       i. For 64-bit Windows Server operating systems, *drive:\\Support\Amd64\Setup.exe /quiet /norestart*.
       ii. For 32-bit Windows Server operating systems, *drive:\\Support\X86\Setup.exe /quiet /norestart*.
12. Repeat steps 1 – 11 for all of the virtual machines on HVSRV1.
13. **[Optional]** For virtual machines that are moved from a Hyper-V Replica server, you must update the virtual machine replication settings on the Hyper-V primary server to reestablish replication.
    a. Open Failover Cluster Manager on the cluster where the primary virtual machine is running and select **Roles**.
    b. Select the virtual machine, and then select **Settings** from the **Actions** pane.
    c. Select **Replication** and update the value for the **Replica server** with the name of the Hyper-V Replica Broker that runs on the new cluster, HVHAR2.

Migrate the old cluster node to the new cluster

After all of the virtual machines from HVSRV1 have been moved to the HVHAR2 cluster, you can evict HVSRV1 from the HVHA201 cluster, install Windows Server 2012 R2, and join the HVHAR2 cluster.

To evict the node from the old cluster

1. On HVSRV2, open the Failover Cluster Manager and select **Nodes**.

2. Select **HVSRV1** and confirm that there are no virtual machines that have moved to **HVSRV1**.
3. Right-click **HVSRV1** to select **More Actions**, and then select **Evict**.

To install Windows Server 2012 R2 and join the new Hyper-V cluster

1. Install Windows Server 2012 R2 on HVSRV1.
2. Install the Hyper-V role and Failover Clustering feature if a clean installation was performed.
3. On HVR2_1, open the Failover Cluster Manager, and then select **Nodes**.
4. In the **Actions** pane, select **Add Node** to open the **Add Node Wizard**.

   Enter the name of the server to be added to the cluster, **HVSRV1**.

   Review the report.

To move the remaining virtual machines

Repeat the following procedures for the HVSRV2 and HVSRV3 servers to complete the migration.

- To move the virtual machines to the new cluster
- Migrate the old cluster node to the new cluster

Copy Cluster Roles Wizard

The Copy Cluster Role Wizard helps you copy clustered roles from clusters running Windows Server 2012 R2, Windows Server 2012, and Windows Server 2008 R2 to a new cluster running Windows Server 2012 R2. After the virtual machine has been created on the new cluster, to complete the migration, you must shut down each virtual machine on the source cluster before you start the virtual machine on the destination cluster.

You can use the Copy Cluster Roles Wizard to do the following:

- Test the Copy Cluster Roles process without affecting production.
- Reverse the process if you encounter any issues.
- Copy roles on test clusters to production clusters.

The Copy Cluster Role Wizard assumes that storage is reused between the old cluster and the new cluster. The cluster role settings are the only data that is copied.

Before migration of the virtual machines from the old cluster, perform the following actions:

- Before running the wizard, you must ensure that the new Windows Server 2012 R2 cluster is configured and is connected to the same storage as the old Hyper-V cluster. For more information about installing a Hyper-V cluster, see Deploy a Hyper-V Cluster.
- Before you work with shadow copies, you should back up all volumes that are attached to the virtual machines.
- Merge or discard all shadow copies for the volumes that store the virtual machines.
- Install the latest updates on all cluster nodes on both clusters.

Important

When you run the Copy Cluster Roles Wizard, the virtual machines are created on the new cluster, but they are not turned on. The virtual machines on the old cluster are still running. After the wizard has finished, you must turn off the virtual machines on the old cluster, and then, on the new cluster, you must start the virtual machines. There is some downtime but its duration should be limited, and you control when the downtime occurs.

To run the Copy Cluster Roles Wizard

1. You must have local Administrator rights on the new and old cluster to run the Copy Cluster Roles Wizard.
2. On the new cluster, open Failover Cluster Manager.
3. Select the top node for the cluster, and click **Copy Cluster Roles** from the **Configure** window.
4. On the **Specify Old Cluster** page, enter the name of the old cluster.
5. On the **Select Roles** page, select the role that you must copy.
6. On the **Customize Virtual Machine Networks** page, specify which virtual network switch the virtual machines are to use on the new cluster.
7. Review the settings and complete the wizard.
8. Review the Failover Cluster Copy Roles Report for any issues.
9. The virtual machines are still running on the old cluster, and the virtual machines that are created on the new cluster are shut off.

To run the virtual machine on new cluster

1. On the old cluster, open Failover Cluster Manager.
2. Turn off the virtual machines that have been copied over to the new cluster.

   Warning

   At no time should a virtual machine be running on both the old cluster and the new cluster. A virtual machine that runs on both the old cluster and the new cluster at the same time might become corrupted. You can run a virtual machine on the old cluster while you migrate it to a new cluster with no problems; the virtual machine on the new cluster is created in a Stopped state. However, to avoid corruption, it is important that you do not turn on the virtual machine on the new cluster until after you stop the virtual machine on the old cluster.
3. On the new cluster, open Failover Cluster Manager.
4. Start the virtual machines and verify that the virtual machine runs correctly.

   Note

   If the migrated cluster is a Hyper-V Replica server, do not start the virtual machines and go to step 6.

   - Run the necessary application-specific tests to ensure that the application on the

virtual machine can provide the same service levels as it provided before the virtual machine was migrated.

- Verify that you can connect to the virtual machine by using **Remote Desktop** or **Virtual Machine Connection**.

5. Update integration services on each virtual machine.
6. **[Optional]** For virtual machines that were copied from a Hyper-V Replica server, you must remove replication and re-enable replication of the virtual machine on the Hyper-V primary server to reestablish replication.
   a. Open Failover Cluster Manager on the cluster where the primary virtual machine is running and select **Roles**.
   b. Select the virtual machine whose Replica virtual machine was copied, and in the **Actions** pane, select **Replication**, and then select **Remove Replication**.
   c. Select the virtual machine, and in the **Actions** pane, select **Replication**, and then select **Enable Replication**. This process opens the **Enable Replication Wizard**.
      - On the **Specify Replica Server** page, specify the name of the Hyper-V Replica Broker in the **Replica server** box.
      - On the **Choose Initial Replication Method** page, select **Use an existing virtual machine on the Replica server as the initial copy**.
7. **[Optional]** For virtual machines that are copied from a Hyper-V primary server, you must remove replication from the Replica virtual machine and enable replication on the virtual machine on the Hyper-V primary server to re-establish replication.

   Perform the following steps on the Replica virtual machine:

   a. Open Failover Cluster Manager on the cluster where the Replica virtual machine is running and select **Roles**.
   b. Select the virtual machine whose primary virtual machine was copied, and in the **Actions** pane, select **Replication**, and then select **Remove Replication**.

   Perform the following steps on the primary virtual machine:

   a. Open Failover Cluster Manager on the new cluster where the primary virtual machine is running and select **Roles**.
   b. Select the virtual machine that was just copied, and in the **Actions** pane, select **Replication**, and then select **Enable Replication**. This process opens the **Enable Replication Wizard**.
      - On the **Specify Replica Server** page, specify the name of the Hyper-V Replica Broker in the **Replica server** box.
      - On the **Choose Initial Replication Method** page, select **Use an existing virtual machine on the Replica server as the initial copy**.

See also

Migrating Clustered Services and Applications to Windows Server 2012

Hyper-V Replica Feature Overview

Hyper-V Cluster Using Cluster Shared Volumes (CSV) Migration

This scenario describes how to migrate virtual machines from a Hyper-V cluster by using Cluster Shared Volumes (CSV) that run on the Windows Server 2012 operating system to the Windows Server 2012 R2 operating system. This scenario reuses the existing CSVs. Migrating the storage to a Scale-out File Server is beyond the scope of the scenario.

The Copy Cluster Roles Wizard is used to move the virtual machine roles to the new cluster.

Copy Cluster Roles Wizard

The Copy Cluster Role Wizard helps you copy cluster roles from clusters that are running Windows Server 2012 R2, Windows Server 2012, and Windows Server 2008 R2 to a new cluster that is running Windows Server 2012 R2.

The Copy Cluster Role Wizard assumes that storage is to be reused between the old cluster and the new cluster. The only data that is copied is the cluster role settings.

You can use the Copy Cluster Roles Wizard to do the following:

- Test the Copy Cluster Roles process without affecting production.
- Reverse the process if you encounter any issues.
- Copy roles on test clusters to production clusters.

Before you migrate the virtual machines from the old cluster, perform the following actions:

- Before you run the wizard, you must ensure that the new Windows Server 2012 R2 cluster is configured and is connected to the same logical unit numbers (LUNs) storage as the old Hyper-V cluster. For more information about installing a Hyper-V cluster, see Deploy a Hyper-V Cluster.
- Before you work with shadow copies, you should back up all volumes that are attached to the virtual machines.
- Merge or discard all shadow copies for the volumes that store the virtual machines.
- Ensure that no virtual machines that you do not want to migrate share a CSV volume with virtual machines that you plan to migrate.
- Install the latest updates on all cluster nodes on both clusters.

Important

When you run the Copy Cluster Roles Wizard, the virtual machines are created on the new cluster, but they are not turned on. The virtual machines on the old cluster are still running. After the wizard has finished, you must turn off the virtual machines and take the disk offline on the old cluster. Then, on the new cluster, you must enable the disk and start the virtual machines. There is some downtime, but its duration should be limited, and you control when the downtime occurs.

Caution

The Copy Cluster Roles Wizard does not copy the replication settings for a virtual machine. After a virtual machine that has Hyper-V replication enabled is moved by using the Copy Cluster Roles Wizard, Hyper-V replication must be removed and be re-enabled.

To run the Copy Cluster Roles Wizard

1. You must be a local administrator on the new and old clusters to run the Copy Cluster Roles Wizard.
2. On the new cluster, open Failover Cluster Manager.
3. Select the top node for the cluster, and then click **Copy Cluster Roles** in the **Configure** window.
4. On the **Specify Old Cluster** page, enter the name of the old cluster.
5. On the **Select Roles** page, select the role that you want to copy.

   Note

   All the virtual machines that are running on a CSV must be migrated at the same time. When you select one virtual machine on a CSV, it automatically selects all of the virtual machines on that CSV.
6. On the **Customize Virtual Machine Networks** page, specify which virtual network switch is to be used by the virtual machines on the new cluster.

   Click **View Report**, to view the Pre-migration report.
7. Review the settings and complete the wizard.
8. Review the Failover Cluster Copy Roles Report to verify that the virtual machines were migrated.
9. The virtual machines are still running on the old cluster, and the virtual machines that are created on the new cluster are off. Additionally, the CSV disk on the new cluster is offline.

To run the virtual machine on new cluster

1. On the old cluster, open Failover Cluster Manager.
2. Turn off the virtual machines that have been migrated over to the new cluster.
3. Take the CSV disk offline.
4. [Optional] In the storage, unmask the CSV disk so that the old cluster can no longer use it.

   Note

   Depending on storage topology, LUN masking and LUN unmasking might be necessary at this stage to ensure that the old cluster does not have write permission to the disks or LUNS that are used by the new cluster.

   Warning

   At no time should a virtual machine run on both the old cluster and the new

cluster. A virtual machine that runs on both the old cluster and the new cluster at the same time might become corrupted. You can run a virtual machine on the old cluster while you migrate it to a new cluster with no problems; the virtual machine on the new cluster is created in a Stopped state. However, to avoid corruption, it is important that you do not turn on the virtual machine on the new cluster until after you stop the virtual machine on the old cluster.

5. On the new cluster, open Failover Cluster Manager.
6. Bring the CSV disk online.
7. Start the virtual machines and verify that the virtual machine runs correctly.

   Note
   If the cluster that is migrated is a Hyper-V Replica server, do not start the virtual machines and go to step 9.

   - Run the necessary application-specific tests to ensure that the application on the virtual machine can provide the same service levels as it provided before the virtual machine was migrated.
   - Verify that you can connect to the virtual machine by using **Remote Desktop** or **Virtual Machine Connection**.
8. Update integration services on each virtual machine.
9. **[Optional]** For virtual machines that are copied from a Hyper-V Replica server, you must remove replication and re-enable replication for the virtual machine on the Hyper-V primary server to re-establish replication.
   a. Open Failover Cluster Manager on the cluster where the primary virtual machine is running and select **Roles**.
   b. Select the virtual machine whose Replica virtual machine was copied, and in the **Actions** pane, select **Replication**, and then select **Remove Replication**.
   c. Select the virtual machine, and in the **Actions** pane, select **Replication**, and then select **Enable Replication**. This action opens the **Enable Replication Wizard**.
      - On the **Specify Replica Server** page, specify the name of the Hyper-V Replica Broker in the **Replica server** box.
      - On the **Choose Initial Replication Method** page, select **Use an existing virtual machine on the Replica server as the initial copy**.
10. **[Optional]** For virtual machines that are copied from a Hyper-V primary server, you must remove replication from the Replica virtual machine and enable replication on the virtual machine on the Hyper-V primary server to re-establish replication.

    Perform the following steps on the Replica virtual machine:

    a. Open Failover Cluster Manager on the cluster where the Replica virtual machine is running and select **Roles**.
    b. Select the virtual machine whose primary virtual machine was copied, and in the **Actions** pane, select **Replication**, and then select **Remove Replication**.

    Perform the following steps on the primary virtual machine:

    a. Open Failover Cluster Manager on the new cluster where the primary virtual machine

is running and select **Roles**.

b. Select the virtual machine that was just copied, and in the **Actions** pane, select **Replication**, and then select **Enable Replication**. This action opens the **Enable Replication Wizard**.
   - On the **Specify Replica Server** page, specify the name of the Hyper-V Replica Broker in the **Replica server** box.
   - On the **Choose Initial Replication Method** page, select **Use an existing virtual machine on the Replica server as the initial copy**.

See also

Migrating Clustered Services and Applications to Windows Server 2012

Cluster Migrations Involving New Storage: Mount Points

Deploy Hyper-V Replica

Migrate File and Storage Services to Windows Server 2012 R2

The File and Storage Services Migration Guide provides step-by-step instructions for how to migrate the File and Storage Services role, including data, shared folders, and operating system settings from a source server to a destination server that is running Windows Server 2012 R2.

About this guide

Note

Your detailed feedback is very important and helps us to make Windows Server Migration Guides as reliable, complete, and easy to use as possible. Click **Rate this topic** at the top of the page and describe what you liked, did not like, or want to see in future versions of the topic. To submit additional suggestions about how to improve Migration guides or utilities, post on the Windows Server 2012 forum.

Migration documentation and tools ease the migration of server role settings and data from an existing server to a destination server that is running Windows Server 2012 R2. By using the tools that are described in this guide, you can simplify the migration process, reduce migration time, increase the accuracy of the migration process, and help to eliminate possible conflicts that might otherwise occur during the migration process. For more information about installing and using the migration tools on both source and destination servers, see Install, Use, and Remove Windows Server Migration Tools.

Specifically, this guide includes information about migrating the following:

- Information about the server's identity
- Local users and groups

- Data and shared folders
- Shadow Copies of Shared Folders
- Data Deduplication
- DFS Namespaces
- DFS Replication
- File Server Resource Manager (FSRM)
- Group Policy settings that are specific to server message block (SMB)
- Group Policy settings for Offline Files (also known as client-side caching or CSC)
- iSCSI Software Target

  Note
  iSCSI Software Target was previously an optional Windows Server and Windows Storage Server component download. Because of the amount of content, all iSCSI-specific migration information is located in File and Storage Services: Appendix C: Migrate iSCSI Software Target.
- Network File System (NFS) file shares
- Remote Volume Shadow Copy Service (RVSS)

Target audience

This document is intended for information technology (IT) professionals and knowledge workers who are responsible for operating and deploying file servers in a managed environment.

What this guide does not provide

This guide does not provide information or support for the following migration scenarios:

- Migrating Roaming User Profiles (for additional information see **Upgrading or Migrating a Roaming User Profiles Environment to Windows 8.1 or Windows Server 2012 R2**).
- Upgrading roles on the same computer
- Migrating more than one server role
- Migrating data across subnets
- Migrating file servers by using File Server Resource Manager
- Migrating encrypted files from Encrypting File System (EFS)
- Migrating file allocation tables (FAT) and FAT32 file systems
- Migrating hardware and software installation for storage resources

In addition to these unsupported scenarios, you should understand the following migration limitations:

- If the hard disk drive that contains your data is physically moved from the source server to the destination server, file and folder permissions for local users are not preserved.
- Reparse points, hard links, and mounted volumes are not migrated when data is copied, and they need to be migrated manually.

- To facilitate migrating file and shared folder permissions, you must migrate local users and groups as part of the migration procedure. However, not all user and group attributes are migrated.

  For more information about the attributes of local users and groups that can be migrated, see the Local User and Group Migration Guide.

Supported migration scenarios

This guide provides instructions for migrating an existing server that is running File and Storage Services to a server that is running Windows Server 2012 R2 or Windows Server 2012. This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, it is recommended that you design a custom migration procedure for your server environment, based on the information that is provided in other server role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server 2012 R2.

Caution

If your source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other server roles that are running on the source server to fail.

Supported migration scenarios include the following configurations or features:

- File server is joined to a domain
- File server is in a workgroup
- File server data and file shares are located in a storage area network (SAN) or other external storage location that preserves data and file share permissions (except data for local users and groups)
- File server data and file shares are located on the server disk (direct-attached storage) that is preserving data and files shares permissions
- DFS Namespaces
- File Server Resource Manager
- iSCSI Software Target
- Network File System (NFS) file shares
- Shadow Copies of Shared Folders

Important

The file migration portion of the Windows Server Migration Tools is designed for smaller data sets (less than 100 GB of data). It copies files one at a time over HTTPS. For larger datasets, we recommend using the version of Robocopy.exe included with Windows Server 2012 R2 or Windows Server 2012.

Supported operating systems

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86-based or x64-based	Windows Server 2003 with Service Pack 2	Windows Server 2012 R2 or Windows Server 2008 R2, both full and Server Core installation options	x64-based
x86-based or x64-based	Windows Server 2003 R2	Windows Server 2012 R2 or Windows Server 2008 R2, both full and Server Core installation options	x64-based
x86-based or x64-based	Windows Server 2008, full installation option	Windows Server 2012 R2 or Windows Server 2008 R2, both full and Server Core installation options	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2012 R2 or Windows Server 2008 R2, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2008 R2	Windows Server 2012 R2 or Windows Server 2008 R2, both full and Server Core installation options	x64-based
x64-based	Server Core and full installation options of Windows Server 2012	Windows Server 2012 R2 or Windows Server 2008 R2, both full and Server Core installation options	x64-based

The versions of operating systems shown in the preceding table are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are supported. Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported as either source or destination servers.

Migrations between physical operating systems and virtual operating systems are supported. Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use to migrate roles, operating system settings, data, or shares from a computer that is running in the French system UI language to a computer that is running in the German system UI language.Windows Server 2012Windows Server 2008Windows Server Migration Tools

Note

The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Both x86-based and x64-based migrations are supported for Windows Server 2008 R2 and Windows Server 2003. All editions of Windows Server 2008 R2 are x64-based.

File services migration overview

The following topics contain step-by-step information about how to migrate File and Storage Services from a computer that is running Windows Server 2003 or later to a computer that is running Windows Server 2012 R2:

- File and Storage Services: Prepare to Migrate
- File and Storage Services: Migrate the File and Storage Services Role
- File and Storage Services: Verify the Migration
- File and Storage Services: Post-Migration Tasks

Impact of migration on other computers in the enterprise

The content in this section describes the impact to the computers in your enterprise during migration.

Impact of data migration by copying data and shared folders

- The performance of your source server can be affected during the data migration. This can result in slower access to files that are stored on the server.
- At the beginning of the second phase of the data migration, all open files are closed, which can lead to data loss.
- During the second phase of data migration, clients are unable to access the file server.

Impact of data migration by physically moving data drives

Clients cannot access the file server from the moment the storage device is disconnected from the source server until it is attached to the destination server.

Impact on DFS Namespaces

The DFS Namespaces are unavailable at several times during the migration process. You should plan your migration when you can take the namespace offline that is hosted on the source server.

Impact on DFS Replication

The impact of migration activity on other servers in the enterprise depends largely on the configuration of the replication topology. Typically, DFS Replication is configured in a hub and spoke replication topology with multiple branch office servers (spokes) replicating with a single hub server. Depending on which server in the replication topology is migrated and how the data is migrated, the remaining servers in the enterprise can be affected. Client workstations that are accessing data that is contained in the replicated folder on the server can be affected during the migration process.

Client computers may be accessing data in the folder that is being replicated by using DFS Replication. The replicated folder is often exposed to client computers as an SMB shared folder.

For more information about the impact of the migration process on client computers, see Impact of data migration by copying data and shared folders earlier in this document.

Permissions required to complete migration

This section describes permissions that are required to perform the migration.

Permissions required for data and shared folder migration

For data and shared folder migration, local Administrator permissions are required on the source server and destination server.

Permissions required to complete migration on the destination server

This section describes permissions that are required to perform the migration on the destination server.

Permissions required to migrate DFS Namespaces

For a stand-alone namespace, the user must be a member of the local Administrators group on the destination server.

There are three permissions options for a domain-based namespace:

- Option 1: Membership in the Domain Admins group
- Option 2 (if there are more than one namespace servers):
  - Permission to administer all namespaces that are hosted on the source server
  - Member of the local Administrators group on the destination server
- Option 3 (if there is a single namespace server):

- Permission to delete and create domain-based namespaces in the domain
- Member of the local Administrators group on the destination server

Permissions required to complete migration on the source server

This section describes permissions that are required to perform the migration on the source server.

Permissions required to migrate DFS Namespaces

For a stand-alone namespace, the user must have membership in the local Administrators group on the source server.

There are three permissions options for a domain-based namespace:

- Option 1: Membership in the Domain Admins group
- Option 2 (if there are more than one namespace servers):
  - Permission to administer the all namespaces that are hosted on the source server
  - Member of the local Administrators group on the source server
- Option 3 (if there is a single namespace server):
  - Permission to delete and create domain-based namespaces in the domain
  - Member of the local Administrators group on the destination server

Permissions required for DFS Replication

For DFS Replication, the user who starts the migration must be a member of the Domain Admins group or delegated permissions to the replication groups and replication members. This permission is required to remove the source server from replication groups to which it belongs. If the permissions to administer a replication group have been delegated to a particular user through the DFS Management snap-in, that user can use the DFS Management snap-in to perform tasks such as removing the source server from a replication group. The user must also be a member of the local Administrators group on the source server and the destination server.

See also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Migrate an iSCSI Software Target

File and Storage Services: Migrate Network File System

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Prepare to Migrate

This guide provides you with instructions for migrating the File and Storage Services role to a server that is running Windows Server 2012 R2.

Install migration tools

Windows Server Migration Tools in Windows Server 2012 R2 allows an administrator to migrate some server roles, features, operating system settings, file shares, and other data from computers that are running certain editions of Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003 to computers that are running Windows Server 2012 R2.

For complete installation, configuration, and removal instructions for Windows Server Migration Tools, see Install, Use, and Remove Windows Server Migration Tools.

Migration documentation and tools ease the process of migrating server role settings and data from an existing server that is running a Windows server operating system to another computer. For a complete list of supported operating systems, see Migrate File and Storage Services to Windows Server 2012 R2.

By using these tools to migrate roles, you can simplify migration, reduce migration time, increase accuracy of the migration process, and help eliminate conflicts that could otherwise occur during the migration process.

Prepare for migration

The following list outlines the major steps for preparing to migrate the File and Storage Services role.

- Prepare the destination server
- Back up the source server
- Prepare the source server
- Prepare other computers in the enterprise

Important
Before you run the **Import-SmigServerSetting**, **Export-SmigServerSetting**, or **Get-SmigServerFeature** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with domain users or groups who are members of local groups on the source server.

Before you run the **Send-SmigServerData** or **Receive-SmigServerData** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with those domain users who have rights to files or shares that are being migrated.

Prepare the destination server

Use the following information to prepare the destination server for migration.

Hardware requirements for the destination server

Verify that the data locations for the destination server have sufficient free space to migrate the data. Ensure that the destination server hard disk drives are the same size or larger than the source server hard disk drives.

Software requirements for the destination server

There are several software requirements that must be met to ensure a successful migration.

- Consult the migration matrix to determine if you can migrate the version of Windows Server that you are running on the source server to Windows Server 2012 R2. For a complete list of supported operating systems, see Migrate File and Storage Services to Windows Server 2012 R2.
- Before migration, install all critical updates and service packs on the source server that were released before Windows Server 2012 R2. It is a recommended best practice that you install all current critical updates and service packs on the source server and the destination server.

Prepare for local user and group migration on the destination server

Verify that the destination server can resolve the names of domain users who are members of the local group during the import operation. If the source server and destination server are in different domains, the destination server must be able to contact a global catalog server for the forest in which the source domain user accounts are located.

Prepare for File and Storage Services on destination server

1. Install Windows Server 2012 R2 on the destination server.
2. Ensure that the time and date are set correctly on the destination server, and that they are in sync with the source server.
3. Determine the File Services role services that have been installed on the source server and then install the same File and Storage Services role services on the destination server.
4. Install Windows Server Migration Tools on the destination server.

   For more information about how to install Windows Server Migration Tools, see Install, Use, and Remove Windows Server Migration Tools.

5. Open UDP port 7000 and make sure that it is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to establish a data transfer connection.

   Note
   If you have changed the default behavior of Windows Firewall to block outbound traffic on computers that are running Windows Server 2012 R2, you must explicitly allow outbound traffic on UDP port 7000.

6. Open TCP port 7000 and make sure that it is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to perform the data transfer.

   For more information about how to open UDP port 7000 and TCP port 7000, see File and Storage Services: Appendix A: Optional Procedures.

   For more information about how to determine if a port is in use, see How To Determine Which Program Uses or Blocks Specific Transmission Control Protocol Ports in Windows Server 2003.

7. Verify that the destination path has sufficient disk space to migrate the data. If NTFS or folder quota management (in File Server Resource Manager) is enabled on the destination server disk drive, verify that the quota limit allows for sufficient free disk space to migrate data. For more information about quota management in File Server Resource Manager, see one of the following:

   - Quota Management for Windows Server 2008 R2 and Windows Server 2008
   - Quota Management for Windows Server 2003 R2

   For more information about NTFS quota management, see one of the following.

   - Setting Disk Quotas for Windows Server 2008 R2 and Windows Server 2008
   - Enable disk quotas for Windows Server 2003 R2 and Windows Server 2003

Prepare File Server Resource Manager on destination server

If you are using File Classification Infrastructure plug-ins from a non-Microsoft vendor, you should register the non-Microsoft plug-ins on the destination server and refer to additional instructions for migration from the non-Microsoft plug-in vendor. You should register the plug-in after File Server Resource Manager (FSRM) has been installed and started on the destination server.

Use the same drive letters for the destination server volumes as for the source server. This is required, because FSRM migration requires the drive letter to remain the same.

Data and file share preparation on destination server

Do not allow users to access the destination server until migration is fully completed. This ensures data integrity and prevents failure when an open file on the destination server cannot be overwritten during migration.

Data integrity and security considerations on destination server

Server migration tools preserve file and folder permissions during data migration. When you are planning the migration, keep in mind that if the migrated files and folders inherit permissions from their parents, during migration it is the inheritance setting that is migrated, not the inherited permissions. Therefore, it is important to make sure that the parent folders on the source server and the destination server have the same permissions to maintain the permissions on migrated data that has inherited permissions.

For example:

1. Migrate folder **c:\A\C** from the source server to folder **c:\B\D** on the destination server.
2. Verify that on the source server, only Mary has access to folder **c:\A** and folder **c:\A\C** is specified to inherit permission from its parent.
3. Verify that on the destination server, only John has access to folder **c:\B**. After **c:\A\C** is migrated to **c:\B\D**, John will have access to folder **D**, but Mary will not.

If you use permissions inheritance for the migrated data, ensure that the parent folder for the migrated data on the destination server has the required permission set.

Prepare DFS Namespaces on destination server

The DFS Namespaces role service must be installed, and the DFS Namespace service must be running before migration. If the namespaces that you are migrating are domain-based, both source and destination servers must be in the same Active Directory Domain Services (AD DS) domain. If the namespaces are stand-alone namespaces, AD DS membership does not matter.

Back up the source server

If DFS Namespaces are being migrated, back up the source server by using a full server backup or system state backup. If the DFS Namespaces are part of an AD DS domain, you need to back up the AD DS domain to save the Active Directory configuration information for DFS Namespaces.

For each domain-based DFS namespace, you should also back up the configuration information for the namespace. Repeat the following command for each namespace and save the output file name to a safe location:

```
DFSUtil.exe root export <\\<DomainName>\Namespace> <Filename>
```

Note

DFSUtil.exe is available on computers that are running Windows Server 2012, Windows Server 2008 R2, and Windows Server 2008. It is available to download for use on Windows Server 2003 R2 and Windows Server 2003 as part of the Windows Server 2003 Service Pack 1 32-bit Support Tools.

Prepare the source server

The following sections describe how to prepare the source server for the migration.

Prepare all file services on source server

- Install Windows Server Migration Tools on the source server.

  For more information about how to install Windows Server Migration Tools, see Install, Use, and Remove Windows Server Migration Tools.
- Verify that the time and date are set correctly on the destination server and that they are synchronized with the source server.
- Open UDP port 7000 and make sure that is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to establish a data transfer connection.

  Note
  If you have changed the default behavior of Windows Firewall to block outbound traffic on computers that are running Windows Server 2012, Windows Server 2008 R2, or Windows Server 2008, you must explicitly allow outbound traffic on UDP port 7000.
- Open TCP port 7000 and make sure that it is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to perform the data transfer.

For more information about how to open UDP port 7000 and TCP port 7000, see File and Storage Services: Appendix A: Optional Procedures.

For more information about how to determine if a port is in use, see How To Determine Which Program Uses or Blocks Specific Transmission Control Protocol Ports in Windows Server 2003.

Data and file share preparation on the source server

To minimize downtime and reduce impact to users, plan your data migration to occur during off-peak hours. Use the `net share` command to list all file shares on the source server.

You can use this list during the verification step to verify that all the required file shares have migrated. Reparse points and hard links will not migrate when data is copied (versus a physical migration), and the reparse points need to be migrated manually. When you migrate hard links, a separate file is created on the destination server for each link. If your migration involves copying the data to the destination server, follow the instructions for how to detect the reparse points and hard links in File and Storage Services: Appendix A: Optional Procedures. Afterward, you can remap and recreate them during migration, as instructed in the For copy data migration scenarios section.

Prepare DFS on the source server

DFS Namespaces role services must be installed, and the DFS Namespace service must be running before migration.

For information about DFS Namespaces preparation, see Prepare DFS Namespaces on source server.

Prepare DFS Namespaces on source server

For domain-based namespaces with one namespace server, determine if you will add a temporary server to the namespace or if you will perform a manual inventory of the namespace permissions.

- Option 1 (recommended):

  Add a temporary server as a namespace server to each domain-based namespace on the source server when the source server is the only namespace server.

- Option 2:

  Inventory the permissions for managing each of the namespaces that are hosted on the source server when the source server is the only namespace server. This process can be completed by using the DFS Management MMC Snap-in.

Prepare other computers in the enterprise

Data and file share migration requires preparing other computers in the enterprise. Perform the following steps for copy data migration scenarios, and for physical data scenarios.

For copy data migration scenarios

- Notify the users that the server performance may be reduced during the first phase of data migration.
- Ask users to stop writing to the server before the second phase of data migration begins (to prevent possible data loss). We recommend that you prevent access to the file shares so that users do not ignore this advice. For example, you could temporarily set all file shares to be read-only by setting the share permissions to Everyone = Read Only.
- Notify users that they cannot access their files on the server when the second phase of the migration begins until the file server migration is fully completed.

For physical data migration scenarios

Notify the users that they cannot access the file server from the moment the storage is disconnected from the source server until the server migration is fully completed.

See also

- Migrate File and Storage Services to Windows Server 2012 R2
- File and Storage Services: Migrate the File and Storage Services Role
- File and Storage Services: Verify the Migration
- File and Storage Services: Migrate an iSCSI Software Target
- File and Storage Services: Migrate Network File System
- File and Storage Services: Post-Migration Tasks
- File and Storage Services: Appendix A: Optional Procedures
- File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Migrate the File and Storage Services Role

Migrate File Services

Perform the following tasks to migrate the File and Storage Services server role.

- Freeze administration configuration
- Export settings
- Migrate local users and groups to the destination server
- Migrate data
- Migrate the source server identity
- Export Remote VSS settings
- Import settings to the destination server

Freeze administration configuration

Administrators must stop all configuration changes to the File and Storage Services role services on the source server before starting migration. When the migration begins, you must not make any configuration changes to the source server other than those that are required for the migration. For example, no links can be added to a DFS namespace after migration starts until the migration is successfully verified.

Install the Windows Server Migration Tools

Before you can use any of the following Windows PowerShell cmdlets for migration on the source server or destination server, ensure that the Windows Server Migration Tools are added. You can do this by using Server Manager or by using Windows PowerShell.

To install the Windows Server Migration Tools

1. Log on to the computer as a member of the local Administrators security group.
2. In Server Manager, click **Add roles and features**.
3. On the **Before you begin** page, click **Next**.
4. On the **Select installation type** page, select **the Role-base or feature-based installation** option, and then click **Next**.
5. On the **Select destination server** page, click **Next**.
6. On the **Select server roles** page, accept the default selections, and then click **Next**.
7. On the **Select features** page, click **Windows Server Migration Tools**, and then click **Next**.
8. On the **Confirm installation selections** page, click **Install**.
9. After the installation is complete, click **Close**.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Install-WindowsFeature Migration
```

The following is a list of Windows Server Migration Tools cmdlets:

- **Export-SmigServerSetting**
- **Import-SmigServerSetting**
- **Get-SmigServerFeature**
- **Send-SmigServerData**
- **Receive-SmigServerData**

For more information about how to work with the Windows Server Migration Tools see Install, Use, and Remove Windows Server Migration Tools.

Export settings

Export the following settings from the source server to the destination server:

- Server Message Block (SMB)
- Offline Files (also known as called client-side caching or CSC)
- DFS Namespaces
- File Server Resource Manager (FSRM)
- Shadow Copies of Shared Folders

BranchCache for Network Files server key

The following procedure applies only if the source server is running Windows Server 2012 or Windows Server 2008 R2.

Notes

This procedure, which is used to migrate the seed value that is used by the BranchCache for Network Files component, enables data that was stored in branch office locations by using BranchCache to be used after the file server is migrated from the source server to the destination server.

For information about how to migrate a BranchCache host server, see the BranchCache Migration Guide.

To migrate BranchCache for Network Files settings from the source server

1. In your Windows PowerShell session, collect data from the source server by running the **Export-SmigServerSetting** cmdlet as a member of the Administrators security group. This step runs the **Export-SmigServerSetting** cmdlet with all parameters from a single

command line. The **Export-SmigServerSetting** cmdlet parameters can collect all source BranchCache feature data in a single file (Svrmig.mig), or you can run the **Export-SmigServerSetting** cmdlet multiple times by using one or more parameters to collect and store data in multiple Svrmig.mig files.

For more information, see the section "Prepare for migration" in File and Storage Services: Prepare to Migrate.

Review the following dependencies before you run the command.

- When you run the **Export-SmigServerSetting** cmdlet, you are prompted to provide a password to encrypt the migration store data. You must provide this same password to import data from the migration store.
- The *path* parameter can be to a folder that is empty or one that contains data. The actual data file in the folder (Svrmig.mig) is created by the **Export-SmigServerSetting** cmdlet. Therefore, the user does not have to specify a file name.
- If the path is not a shared location that the destination server can read, you must manually copy the migration store to the destination server or a location that the destination server can access.
- If a migration store location already exists and you want to rerun the **Export-SmigServerSetting** cmdlet, you must move the Svrmig.mig file from the migration store location and then store it elsewhere, or rename or delete the Svrmig.mig file first.

2. On the source server, type the following, and then press Enter, where <*storepath*> is the path that will contain the Svrmig.mig file after this step is completed. An example of the path is *\\fileserver\users\username\branchcachestore*.

```
Export-SmigServerSetting -featureID BranchCache -Path
<storepath\BranchCache> -Verbose
```

Group Policy setting or local policy setting specific to SMB and Offline Files

Most SMB and Offline Files settings are migrated as part of shared folders migration. The remaining settings that affect the server are set through Group Policy settings or local policy settings. This section describes how to inventory SMB and Offline Files settings that are controlled by Group Policy.

Server message block

Determine the policy settings that affect the SMB server. The SMB settings are controlled by Group Policy settings or local policy settings. If a Group Policy Object (GPO) is applied, these policies override the local settings. First, determine if the policy settings are controlled by a GPO, and then determine local settings for anything that is not controlled by the GPO.

To determine if a GPO is applied to the source server

1. Open the Resultant Set of Policy snap-in. To open the Resultant Set of Policy snap-in, open a command prompt, type **rsop.msc**, and then press Enter.

2. In the snap-in tree pane, click **Computer Configuration**, click **Windows Settings**, click **Security Settings**, click **Local Policies**, and then click **Security Options**.
3. Note in the SMB data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets any Group Policy setting that affects the following **Microsoft network server** settings:
   - Microsoft network server: Amount of idle time required before suspending session
   - Microsoft network server: Attempt S4USelf to obtain claim information
   - Microsoft network server: Digitally sign communications (always)
   - Microsoft network server: Digitally sign communications (if client agrees)
   - Microsoft network server: Disconnect clients when logon hours expire

   On source servers that are running the Server Core installation option of the Windows Server 2012 or Windows Server 2008 R2 operating system, run the **gpresult** command to review Group Policy settings (for more information about **gpresult**, type **gpresult /?** at a command prompt.)

Notes

For any setting that is controlled by Group Policy, you must apply the same GPO to the destination server, or you can set the local policy of the destination server for the same behavior.

For any setting that is not controlled by Group Policy, use the following procedure to determine the local policy setting. Note the local policy setting in the SMB data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets.

To determine local policy settings

1. Open the Local Group Policy Editor. To open the Local Group Policy Editor, open a command prompt, type **gpedit.msc**, and then press Enter.
2. In the snap-in tree pane, click **Computer Configuration**, click **Windows Settings**, click **Security Settings**, click **Local Policies**, and then click **Security Options**.
3. Note the following settings for **Microsoft network server**:
   - Microsoft network server: Amount of idle time required before suspending session
   - Microsoft network server: Attempt S4USelf to obtain claim information
   - Microsoft network server: Digitally sign communications (always)
   - Microsoft network server: Digitally sign communications (if client agrees)
   - Microsoft network server: Disconnect clients when logon hours expire

   On source servers that are running the Server Core installation, run the **secedit** command to export and review local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt.)

Offline Files

Note

This section applies only to source servers that are running Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2, or. Previous operating system releases do not have Offline Files settings that affect the server.

Determine the policy settings that affect shared folders on the server for which client computers use Offline Files. The Offline Files settings are controlled through Group Policy or local policy. If Group Policy is applied, these policies override local settings. First, determine if the settings are controlled through Group Policy, and then determine the local settings for anything that is not controlled by using Group Policy.

To determine if Group Policy is applied to the source server

1. Open the Resultant Set of Policy snap-in. To open the Resultant Set of Policy snap-in, open a command prompt, type **rsop.msc**, and then press Enter.
2. In the snap-in tree pane, click **Computer Configuration**, click **Administrative Templates**, click **Network**, and then click **Lanman Server**.

   Note

   If no policies are set, the preceding path will not exist. If the path does not exist, skip to the procedure To determine local policy settings. If the path exists and policies are found, proceed to the next step.
3. Note in the BranchCache data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets any Group Policy settings that control the **Hash Publication for BranchCache** and **Hash Version support for BranchCache** settings.

   On source servers that are running the Server Core installation option, run the **gpresult** command to review Group Policy settings (for more information about **gpresult**, type **gpresult /?** at a command prompt).

For any setting controlled by Group Policy, have the same Group Policy setting apply to the destination server, or you can choose to set the local policy setting of the destination server to get the same behavior.

For any setting not controlled by Group Policy, use the following instructions to determine the local policy setting.

To determine local policy settings

1. Open the Local Group Policy Editor. To open the Local Group Policy Editor, open a command prompt, type **gpedit.msc**, and then press Enter.
2. In the snap-in tree pane, click **Computer Configuration**, click **Administrative Templates**, click **Network**, and then click **Lanman Server**.
3. Note in the BranchCache data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets the value of the **Hash Publication for**

BranchCache and **Hash Version support for BranchCache** settings.

On source servers that are running the Server Core installation option, run the **secedit** command to export and review local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt).

DFS Namespace configuration

Procedures in this section describe how to migrate DFS Namespaces from the source server to the destination server.

Before the migration of the namespace begins, you can inventory the namespaces that are hosted on the source server for tracking purposes. You can do this by using DFS Management or DFSUtil.exe.

The following procedure (To inventory DFS Namespaces by using DFS Management) applies only to computers that are running at least the Windows Server 2003 R2 version of the Windows Server operating system. For computers that are running Windows Server 2003, you can perform a DFS Namespace inventory by using **DFSUtil.exe** as described in To inventory DFS Namespaces by using DFSutil.exe.

You can also perform a DFS Namespace inventory from a client computer that is running Windows 8, Windows 7, or Windows Vista, by using DFS Management that is part of Remote Server Administration Tools.

- To download Remote Server Administration Tools for Windows 8, see Deploy Remote Server Administration Tools.
- To download Remote Server Administration Tools for Windows 7, see Remote Server Administration Tools for Windows 7.
- To download Remote Server Administration Tools for Windows Vista, see Microsoft Remote Server Administration Tools for Windows Vista.

To inventory DFS Namespaces by using DFS Management

1. Under DFS Management in the left pane, right-click **Namespaces**.
2. Click **Add Namespaces to Display**.
3. In the dialog box that is displayed, select **Server** from the Scope options.
4. Type the name of source server and click **Show Namespaces**.
5. Select all namespaces listed in the list box and click **OK**.
6. Right-click the first namespace listed in the left pane.
7. Click **Properties**.
8. On the **General** tab, check the **Type** field. The type of namespace that is hosted on the server is described here. Possible values are stand-alone, domain-based (Windows Server 2000 mode), and domain-based (Windows Server 2008 mode).
9. In the case of a domain-based namespace, click the **Namespace Servers** tab to identify the number of servers that host the namespace.
10. Repeat steps 7 through 10 for the remaining namespaces listed in the left pane.

To inventory DFS Namespaces by using DFSutil.exe

1. You can inventory your DFS Namespaces using DFSUtil.exe by using the command prompt. From a command prompt, type **DFSUtil.exe server SourceServer** where *SourceServer* represents the name of the source server.
2. Identify the namespaces (DFS roots) listed for the source server.
3. Type the following command, which lists the namespace properties for the first namespace identified in step 2:

   ```
   DFSUtil.exe root <\\SourceServer\Namespace>
   ```
4. Identify the namespace type; possible values are stand-alone root, domain root (domain-based namespace in Windows 2000 Server mode), domainV2 root (domain-based namespace in Windows 2008 mode).
5. Identify the DFS folders present in the namespace in each of the **Link Name** items displayed.
6. In the case of domain-based namespaces, identify all the namespace servers by typing the following command:

   ```
   DFSUtil.exe root <\\Domain\Namespace>
   ```
7. Identify the namespace servers that host the namespace in each of the **Target** items displayed under **Root Name**.
8. Repeat steps 3 through 7 for the remaining namespaces on the source server.

Considerations for namespaces

Is the namespace stand-alone or domain-based? If the namespace is stand-alone, see the following section in this document: Stand-alone namespaces.

If the namespace is domain-based, consider the number of namespace servers for each namespace. For more information, see the following sections in this document:

- Domain-based namespaces with more than one namespace server
- Domain-based namespaces with one namespace server

Stand-alone namespaces

Complete the following procedure to export a stand-alone namespace configuration.

To export the namespace configuration to an export file

1. On the destination server, open a Command Prompt window.
2. Type **DFSUtil.exe root export \\SourceServer\Namespace FileName** to export the stand-alone namespace to a file (where *FileName* represents the exported file), and then press Enter.

Domain-based namespaces with more than one namespace server

For more than one namespace server, remove the namespace server from the namespace by using DFSUtil.exe.

To remove the namespace server from the namespace

1. On the destination server, open a Command Prompt window.
2. Type **DFSUtil.exe target remove <\\SourceServer\Namespace>**, and then press Enter.

Domain-based namespaces with one namespace server

There are two options that you can use in this scenario: export the namespace settings, or add a temporary server to the namespace.

To export namespace settings

1. On the destination server, open a Command Prompt window.
2. Type **DFSUtil.exe root export \\Domain\Namespace FileName** where *FileName* represents the file containing settings for export, and then press Enter.

   Note
   For each namespace, there must be a different file name to export settings.
3. Repeat step 2 for each namespace for which the source server is a namespace server.

You can use either of the following two options if a temporary server can be added to the namespace. This provides the ability to maintain the namespace online while the migration progresses. If this is not possible, follow the steps in To remove the namespace server from the namespace instead.

To add a temporary server to the namespace by using DFS Management

1. In the left pane, select the namespace to be migrated.
2. Click the **Namespace servers** tab.
3. Select **Add Namespace Server**.
4. In the **Namespace server** box, type the name of the temporary server, and then click **OK**.

The temporary server will be added to the namespace.

To add a temporary server to the namespace by using DFSUtil.exe

1. Create a shared folder for the namespace on the temporary server with the same permissions as on the source server.
2. On the destination server, open a Command Prompt window.
3. Type **DFSUtil.exe target add \\TemporaryServer\Namespace** and then press Enter.

```
DFSUtil.exe target add <\\TemporaryServer\Namespace>
```

The temporary server will be added to the namespace.

After the namespace settings are exported or a temporary server is added to the namespace, the namespace source server can be removed from the namespace as described in To remove the namespace server from the namespace.

Inventory advanced registry keys

This section describes the process for determining if there are any settings that have been applied to the DFS Namespace component on the source server. These settings are stored in the registry and set or viewed through the DFSUtil.exe tool. To inventory these settings, run the following commands from the destination server:

```
DFSUtil.exe server registry DfsDnsConfig <SourceServer>

DFSUtil.exe server registry LdapTimeoutValue <SourceServer>

DFSUtil.exe server registry SyncInterval <SourceServer>
```

Note the setting for any registry modification. Registry keys that have not been modified display a value similar to the following:

```
<KeyName> does not exist in the Registry.
```

DFS Replication configuration

To migrate DFS Replication settings, use the following Microsoft Enterprise Support blog series: Replacing DFSR Member Hardware or OS.

File Server Resource Manager configuration on the source server

When you migrate File Server Resource Manager, remember to use the same drive letters for the destination server volumes as for the source server. This is required because the File Server Resource Manager migration requires that the drive letter remains the same.

1. Stop the File Server Resource Manager and File Server Storage Reports Manager services. You can stop these services in Windows PowerShell by using the following command: **Stop-Service –Name “srmsvc”, “srmreports”**.
2. Export the File Server Resource Manager configuration. You can export the File Server Resource Manager configuration in Windows PowerShell by using the following command: **Export-SmigServerSetting -FeatureID FS-Resource-Manager -Path <storepath\FSRM> -Verbose**.
3. For each volume, get the configuration files by running the following commands in the Windows PowerShell session.
    a. Stop the file screen driver. Type **fltmc detach datascrn <VolumeLetter>:** and then press Enter.
    b. Stop the quota driver. Type **fltmc detach quota <VolumeLetter>:** and then press Enter.
    c. Grant Read permissions to the Administrator account for the "<*VolumeLetter*>:\System Volume information\SRM" folder and the following child objects:
        - takeown /F "<VolumeLetter>:\System Volume Information" /A /R /D Y
        - cacls "<VolumeLetter>:\System Volume Information" /T /E /G Administrators:F
        - attrib -S -H "<VolumeLetter>:\System Volume Information*" /S /D
    d. Copy the following files from the **SRM** folder to an external storage device:
        - Quota.xml
        - Quota.md

- Datascrn.md
- DataScreenDatabase.xml

e. Start the file screen driver. Type **fltmc attach datascrn <VolumeLetter>:** and then press Enter.

f. Start the quota driver. Type **fltmc attach quota <VolumeLetter>:** and then press Enter.

4. Restart the File Server Resource Manager and File Server Storage Reports Manage services. Type **Start-Service -name "srmsvc","srmreports"**, and then press Enter.

5. Configure scheduled reports.

   File Server Resource Manager reports and classification rule configurations are dependent on the drive letters and mount points. Any drives or mount points on the source server that are used by report or classification rule configurations must be available on the destination server, or the configurations must be updated during import.

   To configure scheduled reports, follow step (a). However, if you are migrating from Windows Server 2003, follow step (b).

   - To configure scheduled reports on all servers except Windows Server 2003, run the following commands in a Windows PowerShell session on the source server that was opened with elevated user rights.
     - To get a list of all the task names associated with storage reports: `storrept r l`
     - For each task name listed, run the following command on the source server: `schtasks /query /tn:"TASKNAME" /xml > "TASKNAME.xml"`
   - To configure scheduled reports when you migrate from Windows Server 2003:
     - On the source server, do the following:
       - Open File Server Resource Manager.
       - In **Storage Report Management**, for each report task, note the report task, target, and schedule.
     - On the destination server, after you import the file server resource manager configuration, do the following:
       - Open File Server Resource Manager.
       - In **Storage Report Management**, for each report task, edit the report task properties.
       - On the **Schedule** tab, manually add the appropriate schedule for the report.

6. Configure scheduled file management tasks. This step applies only to source servers that are running Windows Server 2012 or Windows Server 2008 R2.

   a. To display a list of all task names associated with file management tasks, type the following command on the source server in a Windows PowerShell session opened with elevated user rights:

   ```
   (new-object -com
   Fsrm.FsrmFileManagementJobManager).EnumFileManagementJobs()
   ```

   b. For each entry listed, locate the task element, and then type the following command:

   ```
   Schtasks /query /tn:"TASK" /xml > "TASK.xml"
   ```

7. Export the classification schedule. This is only applicable to servers running Windows Server 2012 or Windows Server 2008 R2 that already have a classification schedule configured. From an elevated command prompt, type the following command:

```
Schtasks /query /tn:"FsrmAutoClassification{c94c42c4-08d5-473d-8b2d-98ea77d55acd}" /xml > "classification.xml"
```

Shadow Copies of Shared Folders

The following procedures describe how to migrate shadow copy settings.

To migrate shadow copy settings

1. Open Windows Explorer on the source server to view shadow copy storage locations and the creation schedule.

   Important
   This procedure applies to shadow copies for a server running the full installation option of Windows Server. If your source server is running the Server Core installation option of Windows Server, skip this procedure and follow the instructions in the following section: To migrate shadow copies in a Server Core installation.

2. For each volume on the source server, right-click the volume, and then click **Configure Shadow Copies**.

   On source servers that are running Windows Server 2003, right-click the volume, click **Properties**, and then click the **Shadow Copies** tab.

3. Click **Settings**, and note the location and size of the shadow copy storage.
4. Click **Schedule** and note the details for the snapshot creation task.

To migrate shadow copies in a Server Core installation

1. Log on to the computer that is running a Server Core installation remotely as follows:
   a. In Server Manager, click **Tools**, and then click **Computer Management**.
   b. In the **Computer Management** snap-in pane, right-click the top node, and then click **Connect to another computer**.
2. Type the computer name, and then click **OK**.
3. Expand **System Tools**, right-click **Shared Folders**, click the **All Tasks** tab, and then click **Configure Shadow Copies**.
4. For each volume on the source server, right-click the volume, click **Configure Shadow Copies**, click **Settings**, and note the location and size of the shadow copy storage.
5. Click **Schedule**, and then note details for the snapshot creation task.

Migrate local users and groups to the destination server

Before migrating data and shared folders or completing your migration of the FSRM configuration, you must migrate local users and groups. Export local users and groups from the source server, and then import local users and groups to the destination server.

Important

If the source server is a domain member server, but the destination server is a domain controller, imported local users are elevated to domain users, and imported local groups become Domain Local groups on the destination server.

If the source server is a domain controller, but the destination server is not, Domain Local groups are migrated as local groups, and domain users are migrated as local users.

Export local users and groups from the source server

On the source server, export local users and groups to a migration store (as shown in the following example) in a Windows PowerShell session that has been opened with elevated user rights.

```
Export-SmigServerSetting -User All -Group -Path <storepath\UsersGroups> -Verbose
```

You can use one of the following values with the *-user* parameter:

- Enabled: Specify to export only enabled local users.
- Disabled: Specify to export only disabled local users.
- All: Specify to export enabled and disabled local users.

For more information about the attributes of local users and groups that can be migrated, see the Local User and Group Migration Guide.

You are prompted to provide a password to encrypt the migration store. Remember this password, because you must provide the same password to import from the migration store.

If the path is not a shared location that is accessible to the destination server, you must manually copy the contents of the migration store folder to the destination server or a location that is accessible to the destination server.

Import local users and groups to the destination server

On the destination server, import local users and groups from the migration store to which you exported the users and groups in Export local users and groups from the source server, as illustrated by the following example. Use a Windows PowerShell session that has been opened with elevated user rights.

```
Import-SmigServerSetting -User All -Group -Path <storepath\UsersGroups> -Verbose
```

You can use one of the following values with the *-user* parameter:

- Enabled: Specify to import only enabled local users.
- Disabled: Specify to import only disabled local users.
- All: Specify to import enabled and disabled local users.

For the list of user attributes that are supported for migration, see the Local User and Group Migration Guide.

You are prompted to provide the same password that you provided during export to decrypt the migration store.

Migrate data

To migrate data, you can copy file data or physically move it, for example, by attaching the storage drive from the source server to the destination server. If you copy the data, follow the copy data migration steps in the following section.

If you physically move the data, follow the steps described in the Physical data migration section later in this document.

Data copy migration

If you are planning a two-phase data copy migration as described in the previous section, note that if files have been deleted on the source server between the start of the first copy and the start of the final copy, copies of the deleted files might have already transferred to the destination server. So if a file is deleted between the two copy processes, the file might still be available on the destination server after the migration is complete. If this is unacceptable in your environment, perform data and shared folder migration in a single phase, and disconnect all users before starting migration.

Important
The file migration portion of the Windows Server Migration Tools is designed for smaller data sets (less than 100 GB of data). It copies files one at a time over HTTPS. For larger datasets, we recommend using the version of robocopy.exe included with Windows Server 2012 R2 or Windows Server 2012.

To copy data and shared folders and migrate all data without disconnecting users

1. Verify that the destination path has sufficient disk space to migrate the data. If NTFS or folder quota management is enabled on the destination server disk drive, verify that the NTFS or File Server Resource Manager quota limit allows for sufficient free disk space to migrate data. For more information about quota management in File Server Resource Manager, see one of the following:
   - Quota Management for Windows Server 2012, Windows Server 2008 R2, and Windows Server 2008
   - Quota Management for Windows Server 2003 R2

   For more information about NTFS quota management, see one of the following:

- Setting Disk Quotas for Windows Server 2012, Windows Server 2008 R2, and Windows Server 2008
- Enable disk quotas for Windows Server 2003 R2 and Windows Server 2003

2. Ensure that you have completed the migration of local users and groups.

   The **Send-SmigServerData** and **Receive-SmigServerData** cmdlets must be run on the source and destination server within five minutes of each other. By default, **Send-SmigServerData** and **Receive-SmigServerData** time out if a connection cannot be established within 300 seconds (five minutes). This maximum connection time-out for the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets is stored in the following registry subkey, which is user-defined.

   Subkey: HKEY_LOCAL_MACHINE\Software\Microsoft\ServerMigration

   Value: MaxConnectionTime (REG_DWORD)

   Data: Between 1 and 3600 (represents connection time-out, in seconds)

   If a value larger than 3600 is specified, 3600 seconds (1 hour) is used as the maximum connection time-out.

   For information about how to create a Windows Registry key, see Add a Registry Key on the Microsoft Web site.

3. Use the following command to run the **Receive-SmigServerData** cmdlet on the destination server. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Receive-SmigServerData
   ```

   Note
   All output for the Send and Receive operations occurs on the source server only. The destination server will appear to be done before the operation has completed.

4. Use the following command to run the **Send-SmigServerData** cmdlet on the source server to migrate data and shared folders. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Send-SmigServerData -ComputerName <DestinationServer> -
   SourcePath d:\users -DestinationPath d:\shares\users -Recurse
   -Include All -Force
   ```

The destination data location does not have to be the same as the source location, and you can change it, if desired.

Notes
The Server service startup type must be set to Automatic on the destination server for shared folder migration to complete.

Data that is transferred is encrypted automatically. You are prompted to enter a password to encrypt the transferred data on the source server, and the same password to decrypt the received data on the destination server.

After the first data copy is finished, you must freeze the source server and all data changes.

To disconnect users and migrate new or updated files

1. Make sure that users are notified that they should stop using the source server at this time to prevent any possible data loss. You can run the following command to list all the currently open files to determine the potential impact of performing this step.

   ```
   net file
   ```

2. Disconnect all users from the source server by stopping the LanMan Server service.

   ```
   Stop-Service LanmanServer -force
   ```

   Stopping the LanMan Server service invalidates all open remote files to the shared folders on the source server, which can lead to potential data loss. It is best to perform this step when the fewest users are expected to access files on this server.

3. Use the following command to run the **Receive-SmigServerData** cmdlet on the destination server. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Receive-SmigServerData
   ```

4. Use the following command to run the **Send-SmigServerData** cmdlet on the source server to migrate data and shared folders. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Send-SmigServerData -ComputerName <DestinationServer> -
   SourcePath d:\users -DestinationPath d:\shares\users -Recurse
   -Include All -Force
   ```

5. If your scenario requires migrating reparse points, hard links, and mount points, recreate them on the destination server by using the **mklink** command for reparse points and hard links, and using the **mountvol** command for mounted volumes. For more information about these commands, enter `mklink /?` or `mountvol /?` in a Command Prompt window.

It is important to maintain the same destination path that you used during the first copy of data and shared folders. The cmdlets transfer files, folders, and shared folders only if they do not exist on the destination server, or if there is a new version on the source server.

Physical data migration

The next sections describe data migration by physically moving external drives or logical unit numbers (LUNs).

Using disk drives or LUNs to migrate data from the source server to the destination server

You can migrate data from the source server by moving the disk drives. Or, if your data resides on a LUN storage device, you have the option of moving the file server data by masking the LUNs from the source server and unmasking them on the destination server.

For the ideal migration, make sure that you maintain the same mapping of the drive letters (for example, drive D) and the volume IDs (see the following explanation) so that relevant data and application information remains as consistent as possible during the move.

Caution

You should not move a disk drive or LUN if it contains both data and the operating system.

Benefits of physical migration:

- For large amounts of data, this is a faster operation.
- You maintain all data on the disk drive, such as hard links and mount points.
- Shadow copies are preserved if the shadow copies are on the migrated disk drive.

Potential issues to be aware of:

- Permissions for local users that are not default computer accounts (such as local administrators) will be lost even if the same user name is used when creating the user account on the destination server.
- Encrypted files (EFS) cannot be migrated.
- Encrypted volumes with BitLocker cannot be migrated without first decrypting the volumes.
- Remote Storage cannot be migrated.
- When you are physically migrating disk drives that have File Server Resource Manager quotas enabled on them, it is a best practice to dismount the drive gracefully to avoid marking the quotas as dirty. Otherwise, unnecessary scans may occur later.

To migrate data by physically moving the disk drive or by masking and unmasking the LUNs

1. Collect information from the source server.

   Tip

   You can use Server Manager or Windows PowerShell on a computer running Windows Server 2012 or Windows 8 to collect information from source computers running Windows Server 2012.

   a. Record the drive letter and volume label for each data volume on the source server that you would like to move to the destination server.

   b. On the source server, export the volume GUID paths by exporting the following registry subkey to a file: **\HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices**. To do this, open the Registry Editor (regedit.exe), browse to the registry subkey, right-click the registry subkey, and clicking **Export**.

   Alternatively, to export the volume GUID paths from a server running Windows

Server 2012 or Windows Server 2008 R2, open a Windows PowerShell session, and then type the following commands, where <*SourceServer*> is the name of the source server, <*Domain**User*> is a user account with administrative permissions on the source server and <*LocalPath*>\<*Filename*> is a local path and filename of the exported registry keys:

```
Enter-PSSession <SourceServer> -Credential <Domain\User>
Regedit.exe /E <LocalPath>\<Filename>.reg
"HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices"
```

Note

To use Server Manager or Windows PowerShell to remotely collect information from earlier versions of Windows Server, you must first setup the source server for remote management. For more information, see Managing Downlevel Windows-based Servers from Server Manager in Windows Server 2012.

c. Open Notepad, and copy the exported .reg file. Remove all entries that are in the following form: **\DosDevices\D:**. Save the.reg file (all remaining entries should be in the following form: **\??\Volume{ef93fe94-5dd7-11dd-961a-001e4cdb4059}**).

2. Prepare the destination server.

   a. In the Server Manager navigation pane, click **File and Storage Services**, and then click **Volumes** to display the Volumes page. Use the Volumes tile to make sure that the drive letters for the data volumes are available. If there is a drive letter that is currently assigned to an existing volume on the destination server, change the drive letter for that volume.

      Alternatively, use the Windows PowerShell **Get-Volume** and **Set-Partition** cmdlets. For example, to get any volumes with the drive letters of F, G, or H, type `Get-Volume F,G,H`. To change the drive letter of a partition with the F drive letter, type `Set-Partition -DriveLetter F -NewDriveLetter Z`

   b. To import the volume GUID paths into the destination server, copy the.reg file that you created previously to the destination server, and then double-click that file to update the destination server.

3. Move the disk drives or LUNs from the source server to the destination server.

   a. On the source server, remove the disk drives or unassign the LUNs by using Storage Manager for SANs. (To open Storage Manager for SANs, click **Start**, click **Administrative Tools**, and then click **Storage Manager for SANs**.) If the source server is running Windows Server 2012, use the File and Storage Services role in Server Manager instead to view the disks or virtual disks (when using storage pools) that you want to move. If the disk is part of a storage pool, on the **Storage Pools** page of the File and Storage Services role right-click the virtual disk, and then click **Detach Virtual Disk**. For other types of disks, on the **Disks** page, right-click the disk that you want to move and then click **Take Offline**.

   b. On the destination server, attach each disk drive or assign the LUNs, and then assign the appropriate drive letter by using the **Disks** and **Storage Pools** pages of the File

and Storage Services role in Server Manager.

4. If any files or folders on the migrated drive use local users or local groups permissions (except default users and groups), re-create these permission. Note that all domain users and groups permissions will remain intact, assuming that the source server and the destination server are members of the same domain.

Notes

You can use the `icacls` command to modify file and folder permissions (type `icacls /?` in a Command Prompt window for details). Type this same command in a Windows PowerShell session or a command prompt that has been opened with elevated user rights.

The list of the default users and groups is available in the topic Default User Accounts and Groups.

Migrate shared folders

If any of the folders on the migrated drive were shared on the source server and must be shared on the destination server, the following steps explain how to migrate shared folders.

1. If any of the migrated shared folders use local users and group permissions, ensure that you have completed the migration of local users and groups.

   The **Send-SmigServerData** and **Receive-SmigServerData** cmdlets must be started on the source server and the destination server within five minutes of each other. By default, **Send-SmigServerData** and **Receive-SmigServerData** operations terminate if a connection cannot be established within 300 seconds (five minutes). The maximum connection time-out for the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets is stored in the following registry subkey, which is user-defined.

   Subkey: **\HKLM\Software\Microsoft\ServerMigration**

   Value: MaxConnectionTime (REG_DWORD)

   Data: Between 1 and 3600 (represents connection time-out, in seconds). If a value larger than 3600 is specified, 3600 seconds (one hour) is used as the maximum connection time-out.

   For information about how to create a Windows Registry key, see Add a Registry Key.

2. Open port 7000 on the source server and destination server (if this has not already been done).

   For information about how to open a port in Windows Firewall, see File and Storage Services: Appendix A: Optional Procedures.

3. On the destination server:

   a. Open a Windows PowerShell session with elevated user rights and enter the following command: **Receive-SmigServerData**.

4. On the source server:

a. Open a Windows PowerShell session in Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003. On computers that are running Windows Server 2012, Windows Server 2008 R2, or Windows Server 2008, the Windows PowerShell session must be opened with elevated user rights. Enter the following command: **Send-SmigServerData -ComputerName <DestinationServerName> -SourcePath <SourcePath> -DestinationPath <DestinationPath> -Recurse -Include Share -Force**

Notes

The <*SourcePath*> value specifies the local path on the source server that contained the shared folder before the drive was migrated. Shared folder information is not stored on the data drive, so do not be concerned that the drive no longer resides on the source server.

The <*DestinationPath*> value specifies the local path on the destination server that contains folders that were previously shared on the source server. Unless the root drive letter or the folder structure has been changed on the migrated drive, the <*SourcePath*> and <*DestinationPath*> values should be the same.

During shared folder migration, permissions for local users and groups and domain users and groups are migrated—no manual remapping is required.

LanMan Server service automatically restarts on the destination server, and the shared folders migrate.

DFS Replication migration

If you physically migrated data, clean-up the DFS Replication configuration state, which is stored on the migrated volume.

1. To clean up volumes (for each physically migrated volume)
   a. Navigate to the path <volume>\System Volume Information.

      Note

      This is a hidden system folder. To view this folder: in File Explorer, click **View**, and then select the **Hidden Items** check box. Also ensure that local administrators are granted **Full Control** of the folder.

   b. Delete the **DFSR** folder and all content in the folder.
   c. Revert any security permissions modifications that you made to perform the migration process.
   d. Repeat this process for all physically migrated volumes.

2. To clean up replicated folders (for replicated folders on physically migrated volumes)
   a. Navigate to the root of a replicated folder.
   b. Delete the **DfsrPrivate** folder and all subfolders.
   c. If the staging folder for the replicated folder is not located in the default location, remove the staging folder and all content in the staging folder.

      Note
      The default location for the staging folder is in the DfsrPrivate folder, and this step is not required if the path is at the default location.

   d. If the **Conflict and Deleted** folder for the replicated folder is not located in the default location, remove the **Conflict and Deleted** folder and all content in the **Conflict and Deleted** folder.

      Note
      The default location for the **Conflict and Deleted** folder is in the **DfsrPrivate** folder, and this step is not required if the path is at the default location.

Use the inventoried information that you collected for the source server to detect all replication groups to which the source server belongs. Add the destination server as a member server to all these replication groups.

Migrate the source server identity

You need to rename the source server and migrate its previous identity to the destination server. You might also need to migrate the source server IP address to the destination server.

Rename the source server

Rename the source server to a temporary name.

Migrate IP address

When a static IP address is used on the source server, it is recommended that the IP address be migrated from the source server to the destination server. This is because client computers locally cache the IP address that is associated with a server name. Client computers will still attempt to access the source server even if it has been renamed.

When the server IP address is not migrated, you must stop the LanMan Server service on the source server. This is done to prevent users from accessing shared folders on the source server after they have been migrated to the destination server. Open a Windows PowerShell session with elevated user rights, and then run the following cmdlet:

```
Stop-Service LanmanServer -Force
```

For more information about IP address migration, see IP Configuration Migration Guide.

Rename destination server

Rename the destination server to the name that was originally used for the source server.

Export Remote VSS settings

Follow the procedure in this section to migrate Remote VSS settings from Windows Server 2012 R2 or Windows Server 2012.

To migrate Remote VSS from Windows Server 2012 R2 or Windows Server 2012, you must first export the remote VSS settings using the configuration information that is included in the registry and in Group Policy. There are two configuration Group Policy settings for Remote VSS:

- Computer Policy->Administrator Templates->System->File Share Shadow Copy Provider
- Computer Policy->Administrator Templates->System->File Share Shadow Copy Agent

You can configure these settings using either local or domain-based Group Policy. It is recommended that you use a domain-based policy setting because it does not require migration steps—you simply ensure that the policy setting applies to the new destination computer. If you are using a local policy setting, you must document the current settings for these two policy settings by running gpedit.msc. For the remaining policy settings, export the following registry key (using reg.exe from a command prompt with Administrative privileges), and then copy the rvss.reg file to the destination server:

```
Reg.exe export "HKLM\SYSTEM\CurrentControlSet\Services\fssagent\Settings" rvss.reg
```

If you migrated the data by copying it

Follow this procedure to add a replication connection between the source server and the destination server for each replication group on the source server:

1. In Server Manager, click **Tools**, and then click **DFS Management**.
2. In the console tree, under the **Replication** node, select **Add Replication Groups to Display**, enter the name of the source, and then click **Show Replication Groups**. Select all of the replication groups that are displayed, and then click **OK**.
3. For each replication group, do the following:
    a. Click the replication group, and then click **New Member**. The **New Member Wizard** appears. Follow the instructions in the wizard to add the destination server to the replication group by using the information from row #2 in the DFS Replication data collection worksheet (File and Storage Services: Appendix B: Migration Data Collection Worksheets).
    b. In the console tree, under the **Replication** node, right-click the replication group to which you just added the destination server, and then click **New Connection**.
    c. Specify the source server and destination server as sending and receiving members, and specify a schedule so that the connection is always enabled. At this point, the replication is one-way.
    d. Select **Create a second connection in the opposite direction** to create a second connection for two-way replication between the sending and receiving members.

If you migrated the data by physically moving it

Follow this procedure to add a replication connection between the destination server and the closest server to the destination server other than the source server:

1. In Server Manager, click **Tools**, and then click **DFS Management**.
2. In the console tree, under the **Replication** node, select **Add Replication Groups to Display**, enter the name of the source, and then click **Show Replication Groups**. Select all of the replication groups that are displayed, and then click **OK**.
3. For each replication group:
   a. Click the replication group, and then click **New Member**. The **New Member Wizard** appears. Follow the instructions in the wizard to add the destination server to the replication group by using the information from row #2 in the DFS Replication data collection worksheet (File and Storage Services: Appendix B: Migration Data Collection Worksheets).
   b. In the console tree, under the **Replication** node, right-click the replication group to which you just added the destination server, and then click **New Connection**.
   c. Specify the destination server as the sending member, and then specify any other server except the source server as the receiving member. Specify the schedule to use for the connection. It is recommended that you select a server that has a good network connection to the destination server as the receiving member.
   d. Select **Create a second connection in the opposite direction** to create a connection for two-way replication between the sending and receiving members.

Notes

The folder does not begin to replicate immediately. The new DFS Replication settings must be replicated to all domain controllers, and each member in the replication group must poll its closest domain controller to obtain these settings. The amount of time this takes depends on Active Directory Domain Services (AD DS) replication latency and the polling interval (60 minutes) on each member. The **dfsrdiag /pollad** command can be used to force DFS Replication on the source server and destination server to poll AD DS and retrieve the latest configuration information instead of waiting for the next normal polling interval which could be up to 60 minutes.

After DFS Replication on the destination server polls AD DS, it begins to replicate the folders that it configured, and it performs an initial synchronization. Event ID 4102 (MSG_EVENT_DFSR_CS_INITIAL_SYNC_NEEDED) is registered in the event log on the destination server for each replicated folder.

During initial sync, DFS Replication downloads all files in the replicated folders from the source server and builds up a local copy of the database per volume. This process can be time consuming. It is possible to speed up the initial sync by seeding the data from the source server onto the destination server (from the backup that was taken prior to

commencing migration).

When the initial sync completes, event ID 4104 (MSG_EVENT_DFSR_CS_INITIAL_SYNC_COMPLETED) is registered for each replicated folder on the destination server. Monitor each replicated folder on the destination server and check to ensure that all of them have completed the initial sync.

Import settings to the destination server

Follow the procedures in this section to import settings to the destination server.

Note

If the source server is not running Windows Server 2012 or Windows Server 2008 R2, the first procedure in this section does not apply. (This procedure is used to migrate the seed value that is used by BranchCache for the Network Files component, and it enables data that is stored in BranchCache on the source server to be used after it is migrated to the destination server. For information about how to migrate a BranchCache host server, see the BranchCache Migration Guide.

To set up BranchCache for Network Files migration on the destination server

1. On the destination server, open a Windows PowerShell session with elevated user rights.
2. Type the following command, where *storepath* is the available path that contains the Svrmig.mig file, and then press Enter.

```
Import-SmigServerSetting -featureid BranchCache -Path
<storepath\BranchCache> -Force -Verbose
```

Group Policy or local policy specific to server message block and Offline Files

Use a Group Policy Object or a local policy setting on the destination server to change the settings to the same values as the source server. These settings are recorded in the SMB and BranchCache data collection worksheets in File and Storage Services: Appendix B: Migration Data Collection Worksheets.

To import SMB settings

1. Do one of the following:
   - If the policy settings are set by using Group Policy Objects, use the Group Policy editing tools to apply appropriate policy settings to the destination server.
   - If the policy settings are set by using a local policy setting, do the following:
     i. On the destination server, open the Local Group Policy Editor snap-in.
     ii. In the snap-in tree pane, click **Computer Configuration**, click **Windows**

Settings, click **Security Settings**, click **Local Policies**, and then click **Security Options**.

2. Use a Group Policy Object or a local policy setting to set the following settings to the same values as noted in Export settings. Set the destination server settings to the same values as were noted on the source server for the following **Microsoft network server** settings:
   - Microsoft network server: Amount of idle time required before suspending session
   - Microsoft network server: Attempt S4USelf to obtain claim information
   - Microsoft network server: Digitally sign communications (always)
   - Microsoft network server: Digitally sign communications (if client agrees)
   - Microsoft network server: Disconnect clients when logon hours expire

   Note
   For any setting that is controlled by Group Policy, you must have the same Group Policy Object apply to the destination server, or you can set the local policy of the destination server to get the same behavior.

   On destination servers that are running the Server Core installation, run the **secedit** command to change local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt).

Note

The following procedure applies only if the source server is Windows Server 2012 or Windows Server 2008 R2.

To import Offline Files settings

1. Do one of the following:
   - If the policy settings are set by using Group Policy, use the Group Policy editing tools to apply appropriate policy settings to the destination server.
   - If the policy settings are set by using local policy, do the following:
     i. On the destination server, open the Local Group Policy Editor snap-in.
     ii. In the snap-in tree pane, click **Computer Configuration**, click **Windows Settings**, click **Administrative Templates**, click **Network**, and then click **LanMan Server**.
2. Use a Group Policy Object or a local policy setting to set the destination server policy settings to the same values as the source server policy settings for **Hash Publication for BranchCache** and **Hash Version support for BranchCache** settings.

   On destination servers that are running the Server Core installation, run the **secedit** command to change local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt).

DFS Namespace configuration

Complete the configuration of namespaces on the destination server. The procedure you use depends on whether you want a stand-alone or a domain-based namespace.

- Stand-alone namespaces
- Domain-based namespaces with more than one namespace server
- Domain-based namespaces with one namespace server

Stand-alone namespaces

If you want a stand-alone namespace, you must first create the namespace on the destination server. You can do this by using DFS Management, or the **DFSUtil.exe** command-line utility.

To create the namespace on the destination server

1. Do one of the following:
   - On the destination server, open DFS Management, and create the namespace by using the same name as on the source server.
   - On the destination server, in a Command Prompt window opened with elevated user rights, type the following, and then press Enter.

     ```
     Dfsutil.exe root addstd <\\DestinationServer\Namespace>
     ```

To import a namespace configuration from the export file

1. On the destination server, in a Command Prompt window opened with elevated user rights, type the following (in which *filename* represents the file name into which you exported namespace settings from the source server in To export the namespace configuration to an export file), and then press Enter.

   ```
   Dfsutil.exe root import set <filename>
   <\\DestinationServer\Namespace>
   ```

Domain-based namespaces with more than one namespace server

If you have more than one domain-based namespace server, you can add namespace servers to your destination server by using DFS Management or the **DFSUtil.exe** command-line utility.

To use DFS Management

1. Select the namespace being migrated in the left pane.
2. Click the **Namespace servers** tab in the right pane.
3. Select **Add Namespace Server**.
4. In the dialog box that opens, type the name of the destination server, and then click **OK**.

The destination server is added to the namespace.

To use DFSUtil.exe

1. On the destination server, open a Command Prompt window.
2. Type the following command, and then press Enter.

```
DFSUtil.exe target add <\\DestinationServer\Namespace>
```

Domain-based namespaces with one namespace server

This section applies only if a temporary server was not added to the namespace. If you added a temporary server to the namespace as part of your export process, see Domain-based namespaces with more than one namespace server.

To create the namespace on the destination server

1. Do one of the following:
   a. In DFS Management on the destination server, create the namespace with the same name that was used on the source server.
   b. Type the following command at a command prompt, and then press Enter.

```
Dfsutil.exe root adddom <\\DestinationServer\Namespace>
```

To import a namespace configuration from the export file

1. On the destination server, open a Command Prompt window.
2. Type the following command (in which *Filename* represents the export file names you created in To export namespace settings). Run this command for each of the namespaces for which the source server was a namespace server.

```
DFSUtil.exe root import set <Filename>
\\DestinationServer\Namespace
```

> **Note**
> For each namespace, there must be a file name from which settings are imported.

To manually reset delegation permissions on the namespace

1. On the destination server, open DFS Management.
2. Set the permissions that you inventoried in DFS Namespace configuration. When complete, close DFS Management.

If any advanced registry keys were configured on *SourceServer*, use **DFSUtil.exe** to configure *DestinationServer* to have the same registry key settings. Run the following commands on the destination server to set the advanced registry keys.

To set advanced registry keys

1. On the destination server, open a Command Prompt window.
2. Run the following commands to set the advanced registry keys by using **DFSUtil.exe**.

```
DFSUtil.exe server registry DfsDnsConfig set
<DestinationServer>
DFSUtil.exe server registry LdapTimeoutValue set <Value>
<DestinationServer>
DFSUtil.exe server registry SyncInterval set <Value>
<DestinationServer>
```

File Server Resource Manager configuration on the destination server

When you are migrating File Server Resource Manager, remember to use the same drive letters for the destination server volumes as for the source server. This is required because File Server Resource Manager migration requires that the drive letter remains the same.

1. Stop the File Server Resource Manager and File Server Storage Reports Manager services.

   Open a Windows PowerShell session with elevated user rights, and then run the following command:

   ```
   Stop-Service -name "srmsvc","srmreports"
   ```

2. Type the following in the Windows PowerShell session, and then press Enter.

   ```
   Import-SmigServerSetting -FeatureID FS-Resource-Manager -Path
   <storepath\FSRM> -Force
   ```

   Notes

   If the Windows features that you are migrating have not been installed on the destination server, the **Import-SmigServerSetting** cmdlet installs them as part of the import process, along with any Windows features that they depend on. Some Windows features might require that you restart the destination server to complete the installation. After restarting the computer, you must run the cmdlet again with the **-Force** parameter to complete the import operation.

   Importing FSRM settings to the destination server replaces any global FSRM configuration information that is already on the destination server.

3. Set the configuration files for each volume.

   Type the following commands in a Windows PowerShell session, and then press Enter.

   Note

   Running the following commands on a clean computer returns an error message. It is safe to ignore this error message.

   a. Type the following command to stop the file screen driver:

```
fltmc detach datascrn <VolumeLetter>:
```

b. Type the following command to stop the quota driver:

```
fltmc detach quota <VolumeLetter>:
```

c. Add administrator Write permissions to the "<VolumeLetter>:\System Volume information\SRM" folder and the following subfolders:

- takeown /F "<VolumeLetter>:\System Volume Information" /A /R /D Y
- cacls "<VolumeLetter>:\System Volume Information" /T /E /G Administrators:F
- attrib -S -H "<VolumeLetter>:\System Volume Information*" /S /D

d. Copy the following files from the external storage to the **SRM** folder:

- Quota.xml
- Quota.md
- Datascrn.md
- DataScreenDatabase.xml

e. Type the following command to start the file screen driver:

```
fltmc attach datascrn <VolumeLetter>:
```

f. Type the following command to start the quota driver:

```
fltmc attach quota <VolumeLetter>:
```

4. Restart the File Server Resource Manager and File Server Storage Reports Manager services.

   Type the following command in a Windows PowerShell session, and then press Enter.

   ```
   Start-Service -name "srmsvc","srmreports"
   ```

5. Configure scheduled reports and file management tasks.

   For each scheduled report, you need to create a scheduled task on the destination server.

   Note
   File Server Resource ManagerReports and classification rule configurations are dependent on the drive letters and mount points. Any drives or mount points on the source server that are used by report or classification rule configurations must be available on the destination server or the configurations must be updated during import.

   After you have an XML file for each task, copy them to the destination server and run the following command in a Windows PowerShell session on the destination server for each task:

   ```
   schtasks /create /xml:"TASKNAME.xml" /tn:"TASKNAME"
   ```

6. Import the classification schedule. The classification schedule requires a scheduled task on the destination server.

   ```
   schtasks /create /xml:"classification.xml"
   /tn:"FsrmAutoClassification{c94c42c4-08d5-473d-8b2d-
   98ea77d55acd}"
   ```

Note that *classification.xml* is the name of the XML file that was exported from the target server.

Shadow Copies of Shared Folders

Apply the same settings from the source server to the corresponding volumes on the destination server.

To migrate shadow copy settings for Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003

1. To configure shadow copies, right-click each volume on the destination server that had shadow copies configured on the source server, right-click the volume, and then click **Configure Shadow Copies**.
2. Click **Settings** and verify that the location and size of shadow copy storage matches the settings from the source server.
3. Click **Schedule** and verify that the details for the snapshot creation task match the settings from the source server.

To migrate shadow copy settings for a Server Core installation

1. Log on to the destination server that is remotely running the Server Core installation by doing the following:
   a. In Server Manager, click **Tools**, and then click **Computer Management**.
   b. In the **Computer Management** snap-in tree pane, right-click the top node, and then click **Connect to another computer**.
2. Enter the computer name, and then click **OK**.
3. Expand **System Tools**, right-click **Shared Folders**, click the **All Tasks** tab, and then click **Configure Shadow Copies**.
4. For each volume on the destination server that had shadow copies configured on the source server, right-click the volume, click **Configure Shadow Copies**, click **Settings**, and verify that the location and size of shadow copy storage match the settings from the source server.
5. Click **Schedule**, and verify that these details for the snapshot creation task match the settings from the source server.

Deduplication

Use the following section to migrate Deduplication.

Migrating Deduplication from Windows Server 2012 to Windows Server 2012

All configuration information needed for migration is included on the deduplicated volume.

If a disk is physically moved, or if a deduplicated volume is restored from a backup onto a different Windows Server 2012 computer, install the Deduplication role service using Server Manager on the new computer. If the Deduplication role service is not installed on the new server, only normal nondeduplicated files will be accessible. After a volume has been mounted, the deduplication filter will detect that the volume is deduplicated and will redirect input/output requests appropriately.

Note

Any previous custom deduplication job schedules that were created using Task Scheduler must be created again on the new computer using Task Scheduler.

Migrating SIS from Windows Storage Server 2008 to Windows Server 2012

Volumes that have been created and optimized using the down-level Windows Storage Server version of deduplication, Single Instance Storage (SIS), should not be enabled for data deduplication. Microsoft recommends that SIS be removed from the volume by using SISAdmin.exe within Windows Storage Server to remove SIS or by copying the data to a different volume that is not running SIS prior to migrating the volume.

Windows Server 2012 supports reading and writing to SIS-controlled volumes, but you cannot continue to SIS files using Windows Server 2012. You can install the SIS filter driver on Windows Server 2012 by installing the SIS-Limited feature using the following command syntax:

```
dism /online /enable-feature:SIS-Limited
```

The SIS filter driver can be loaded so that you can read SIS-controlled volumes and the data can be copied to a non-SIS controlled volume so that data deduplication can be installed on the volume. Note that Windows Server 2012 does not support sisadmin.exe and cannot be used to remove SIS from a volume.

1. You should remove SIS from your volumes before installing the Windows Server 2012 data deduplication feature. (This process is also known as un-SIS.)
2. Do not restore SIS links from a backup to a Windows Server 2012 deduplicated volume.
3. Restoring SIS volumes to Windows Server 2012 is supported if you load the SIS-Limited filter.

Migrating SIS volumes

You have several options when it comes to migrating Windows Storage Server 2008 volumes to Windows Server 2012 to take advantage of the new Data Deduplication feature.

You can migrate your existing SIS-installed Windows Storage Server 2008 volumes to Windows Server 2012; however, migration is not automatic. SIS and data deduplication are mutually-exclusive technologies.

Caution

You will need to open the volumes in Windows Storage Server 2008 first, un-SIS them, and then uninstall SIS before migrating to Windows Server 2012 as described in the procedures below.

To unSIS a Windows Storage Server 2008 R2 or 2008 SIS volume, type **sisadmin.exe [/m <server>] [/u <volumes>]** where:

1. **/m <server>** shifts the focus of the command line to a remote server. If the */m* option is not specified, the command line is applied to the local server. <*server*> can be expressed as a host name, fully qualified domain name (FQDN), or IP address.
2. **/u <volumes>** is used to un-SIS a volume (that is, to restore all file copies and remove reparse points).

For each command option that uses <*volumes*> as a parameter, <*volumes*> represents a space-delimited list of volume names (for example, d:, e:, f:, and g:).

To unSIS or remove SIS entirely from the F: volume of a remote server using the IP address of the server, you might use the following command: **sisadmin.exe /m 192.168.1.50 /u F:**

Import Remote VSS settings

Follow the procedure in this section to migrate Remote VSS settings from Windows Server 2012 R2 or Windows Server 2012.

To finish migrating Remote VSS from Windows Server 2012 R2 or Windows Server 2012, import the remote VSS settings using the configuration information that is included in the registry and in Group Policy. There are two configuration Group Policy settings for Remote VSS:

- Computer Policy->Administrator Templates->System->File Share Shadow Copy Provider
- Computer Policy->Administrator Templates->System->File Share Shadow Copy Agent

You can configure these policy settings using either local or domain-based Group Policy. It is recommended that you use a domain-based policy setting because it does not require migration steps—you simply ensure that the policy setting applies to the new destination computer. If you are using a local policy setting, open gpedit.msc and recreate the policy settings that you documented in Export Remote VSS settings.

For the remaining policy settings, export the registry key that you previously exported by using reg.exe from a command prompt with Administrative privileges:

For the remaining policy settings, import the registry key by using reg.exe from a command prompt with Administrative privileges:

```
Reg.exe import rvss.reg
```

See also

- Migrate File and Storage Services to Windows Server 2012 R2
- File and Storage Services: Prepare to Migrate
- File and Storage Services: Verify the Migration
- File and Storage Services: Migrate an iSCSI Software Target

- File and Storage Services: Migrate Network File System
- File and Storage Services: Post-Migration Tasks
- File and Storage Services: Appendix A: Optional Procedures
- File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Verify the Migration

To verify that the migration was successful, follow the appropriate verification steps based on the File and Storage Services role services that have been migrated.

The following overview describes the steps to verify the migration.

Verify the File Services migration

Perform the following tasks to verify the File and Storage Services role migration.

- Verify the File Services migration (only if running Windows Server 2012 or Windows Server 2008 R2)
- Verify migration of local users and groups
- Verify data and shared folder migration
- Verify the migration of DFS Namespaces
- Verify the configuration on other computers
- Verify the File Server Resource Manager migration

Verify migration of BranchCache for Network File Services server key

Perform this step only if the source server is running Windows Server 2012 or Windows Server 2008 R2:

Verify that the server key was migrated correctly by checking the key value, and ensure that the key values are identical on the source server and destination server, as shown in the following example:

```
Key: HKLM\Software\Microsoft\WindowsNT\CurrentVersion\PeerDist\SecurityManager\Restricted
Value: Seed
```

Verify migration of local users and groups

Check that all the local users and groups you expected to migrate are present on the destination server by comparing the list of users and groups on the Local Users and Groups snap-in on the source server with the list on the destination server.

To open the Local Users and Groups

1. In Server Manager, click **Tools**, and then click **Computer Management**.

Alternatively, you can compare the list of users and groups on the source server and destination server by typing **net** commands in a Command Prompt window.

- To get the list of all local users and save it in a text file, type the following command:

```
net user > localusers.txt
```

- To get the list of all local groups and save it in a text file, type the following command:

```
net localgroup > localgroups.txt
```

Verify data and shared folder migration

1. Check that all the data you expected to migrate is present at the correct location on the destination server and that the data has the correct permissions associated with it.

   To list files and folders with their permissions, type the following command in a Command Prompt window or in a Windows PowerShell session opened with elevated user rights:

```
icacls <path>
```

2. Verify that all the expected shared folders have migrated and that they have the correct permissions associated with them. To list all shared folders and their permissions, type the following command in a Windows PowerShell session opened with elevated user rights:

```
gwmi win32_share | %{net share $_.name}
```

Verify the migration of DFS Namespaces

The procedure that you use to verify the migration of DFS Namespaces depends on whether your namespaces are stand-alone or domain-based.

To verify the migration of a stand-alone namespace

1. Open DFS Management on the destination server.
2. Right-click **Namespaces**, or click the **Action** menu.
3. Click **Add Namespaces to Display**.
4. Type the name of the destination server, and then click the **Show Namespaces** button. Select the namespace that you migrated, and then click **OK**.
5. In the namespaces tree, click the namespace that you migrated.
6. Click the **Namespace** tab, and check that all the namespace links are present.
7. Click the **Namespace server** tab, and check that the destination server is listed.
8. Right-click the destination server name, and then click **Open in Windows Explorer**. All namespace links should be visible in the new window.
9. Using DFSUtil.exe on the destination server, type the following command for each stand-alone namespace:

```
Dfsutil.exe root \\DestinationServer\Namespace
```

The information displayed should contain the destination server and all the namespace links.

To verify the migration of a domain-based namespace

1. Open DFS Management, and then right-click **Namespaces** or click the **Action** menu.
2. Click **Add Namespaces to Display**.
3. Type the name of the domain where the namespace is located, and then click the **Show Namespaces** button. Select the namespace that you migrated, and click **OK**.
4. In the namespaces tree, click the namespace that you migrated.
5. Click the **Namespace** tab, and check that all the namespace links are present.
6. Click the **Namespace** server tab, and check that all the namespace servers are listed.
7. Right-click the destination server name, and then click **Open in Windows Explorer**. All namespace links should be visible in the new window.
8. Using DFSUtil.exe on the destination server, type the following command in a Command Prompt window, where *Domain**Namespace* is the name of the appropriate domain and namespace that you migrated.

   ```
   Dfsutil.exe root <\\Domain\Namespace>
   ```

   The information displayed should contain all namespace servers and namespace links.

Verify the configuration on other computers

To verify that File and Storage Services migration completed successfully on other computers, you must test the configuration on the client computers in your enterprise.

To verify DFS Namespaces on a client computer

1. Log on to a client computer with the credentials of a user who has access to the migrated namespace.
2. Verify that you can access the namespace by using File Explorer, a command prompt window, or another application, by entering the same name that you used before the migration.

Verify the File Server Resource Manager migration

Follow these steps to ensure that File Server Resource Manager migrated:

1. If any custom actions are configured for quota notification or file management tasks, the user should ensure that the folders that contain the executable files configured for the actions and the working folders have the correct access control lists.
2. As a best practice, ensure that all email message text for notifications, reports, and other purposes migrated correctly.

3. Administrators should send a test email message through the File Server Resource Manager console to verify that the Simple Mail Transfer Protocol (SMTP) server is configured correctly for the destination server.
4. Ensure that expiration folders that are used by File Management Tasks are reachable on the destination server.
5. Ensure that executable files that are used by custom actions (such as quota notifications and file management tasks) are accessible or executable on the destination server.

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Migrate an iSCSI Software Target

This section describes how to migrate Microsoft iSCSI Software Target 3.2 or 3.3 settings and data from an existing Windows Storage Server 2008 R2 or Windows Storage Server 2008 computer to a destination server that is running the iSCSI Target Server role service that is included with Windows Server 2012 R2 or Windows Server 2012 and Windows Storage Server 2012.

The naming for iSCSI Software Target has changed. To reduce the potential for confusion, in the context of this document, any naming that refers to “iSCSI Software Target”, refers to prior product versions installed on Windows Storage Server 2008 R2 and Windows Storage Server 2008, which are source servers. By contrast, any naming that refers to “iSCSI Target Server” refers to the new role service included with Windows Server 2012 R2, Windows Server 2012, and Windows Storage Server 2012, which are destination servers.

Note

This section contains only iSCSI-specific migration information. For generic information, such as the use of Windows Server Migration Tools, refer to the application section in the main File and Storage Services Migration Guide.

Supported migration scenarios

This section details both supported and unsupported migration scenarios.

Supported operating systems

The versions of operating systems that are listed are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are supported.

Migrations between physical operating systems and virtual operating systems are supported.

Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shared resources from a computer that is running Windows Server 2008 in the French system UI language to a computer that is running Windows Server 2012 R2 or Windows Server 2012 in the German system UI language.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x64-based	Windows Server 2008 R2	Windows Server 2012 R2 and Windows Storage Server 2012	x64-based
x64-based	Windows Storage Server 2008 R2	Windows Server 2012 R2 and Windows Storage Server 2012	x64-based
x64-based	Windows Server 2012	Windows Server 2012 R2 and Windows Storage Server 2012	x64-based
x64-based	Windows Storage Server 2012	Windows Server 2012 R2 and Windows Storage Server 2012	x64-based

x64-based migrations are supported for Windows Storage Server 2012 R2 and Windows Server 2012 R2. All editions of Windows Storage Server 2008 R2 and Windows Server 2008 R2 are x64-based.

x86-based migrations are not supported because Windows Storage Server 2012 R2 is not offered in the x86 platform.

Note

The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Supported role configurations

This migration guide is applicable to stand-alone and clustered configurations, with certain limitations.

The following general restrictions are applicable to all the supported configurations:

- Authentication settings for iSCSI initiators that use CHAP and Reverse CHAP settings are not automatically migrated.
- Snapshot storage settings for each virtual disk in the configuration are not automatically migrated.
- Configuration settings for virtual disks that are derived from snapshots are not automatically migrated.
- For clustered configurations, the migration process includes iSCSI target settings that are scoped to the virtual computer object, to a cluster node, or to the cluster node that owns the code cluster group.
- For clustered configurations, the migration of resource groups, network name resources, IP addresses, and cluster disks that are associated with resource groups is outside of the scope of this guide, and the migration needs to be performed independently as a preliminary step.
- iSCSI Naming Services (iSNS) settings for iSCSI Software Target are not automatically migrated.
- iSCSI target portal settings (such as IP addresses that are used by the iSCSI target service to listen for incoming network connections) are not automatically migrated
- The schedule for snapshots of virtual disks is not migrated.

The following configurations are supported:

- Migration from a stand-alone configuration to stand-alone configuration
- Migration from a clustered configuration to a stand-alone configuration (with the restrictions listed previously regarding the scope of the settings).
- Migration from a clustered configuration to a clustered configuration (with the restrictions listed previously regarding the scope of the settings).

Supported role services and features

iSCSI Target Server (as included with Windows Storage Server 2012 and Windows Server 2012 R2) does not have role dependencies or feature dependencies.

It is possible to install iSCSI Target Server with failover clustering, and this configuration is supported with the migration limitations listed previously.

Migrating multiple roles

If you are migrating one clustered configuration to a different clustered configuration, the Failover Clustering feature needs to be migrated or set up prior to migrating iSCSI target settings.

Migration scenarios that are not supported

The following migration scenarios are not supported:

- Migration from Windows Unified Storage Server 2003 R2.
- Migration from a stand-alone configuration to a clustered configuration. This migration is not supported because there is no default mechanism to associate target and virtual disk settings to resource groups without knowing how the file paths are mapped to the cluster disk and how IP addresses are mapped to resource groups.
- Snapshots of virtual disks are not automatically migrated. Snapshots are based on a snapshot of the volume that contains the virtual hard disk (VHD) file at the time the snapshot was taken. Their existence and implementation depends on the volume of the computer from which the migration process happens, and it cannot be replicated or exported.
- Snapshot storage settings for virtual disks are not automatically migrated. The snapshot storage settings (such as volume and maximum size per volume) are dependent on the hardware and software configuration of the computer to which the settings are being migrate, and they cannot automatically be migrated. For detailed information about how to manually migrate the snapshot storage settings, see Migrating iSCSI Target.
- The configuration settings of the iSCSI target portal are not automatically migrated. This configuration is based on the IP addresses of the destination server, and those settings cannot be migrated outside the knowledge of the network configuration of the computer to which the settings are being migrate. For detailed information about how to manually configure the portal settings, see Migrating iSCSI Target.
- iSNS settings are not automatically migrated. The iSNS settings are based on the network infrastructure and configuration of the destination server, and those settings cannot be migrated outside the knowledge of the network configuration of the computer to which the settings are being migrated. For detailed information about how to manually configure iSNS settings, see Migrating iSCSI Target.
- Settings for virtual disks that are attached as local disks on the source server are not automatically migrated. The ability to attach a disk locally is expected to be a temporary operation that can be replicated if. For detailed information about how to configure settings for virtual disks that are to be attached as local disks, see Migrating iSCSI Target.
- The schedule for snapshots of virtual disks is not migrated. Those settings must be manually discovered and replicated from the source to the destination server.

Migration overview

This section describes the high-level migration process, which involves harvesting configuration settings from the source, moving the virtual disks from the source server to the destination server, and restoring the configuration settings.

Migration process

This section describes the high-level migration process.

Migration planning

The migration planning phase involves gathering information based on the following questions:

- Are the source server and destination server configured in a cluster?

- If the servers are configured in a cluster, what are the virtual computer objects or client access points that contain the iSCSI target resources?
- Is the storage system of the destination server capable and configured appropriately to host the virtual disks of the source server, and does it have appropriate space to store the volume snapshots?
- Are there any iSCSI initiators that have a critical dependency on iSCSI targets for the duration of the migration process (such as a computer that uses iSCSI boot nodes, or clusters that use shared storage)?
- Are there any IP address or portal settings that are unique to the source server that need to be accounted for (such as IP addresses that are known to the firmware of devices)?
- Are there any iSNS settings that need to be manually recorded and migrated?
- Are there any virtual disks attached as local disks that might need to be exposed?

Preparing to migrate

The preparation to migrate data from the source server to the destination server involves the following steps:

1. If the destination server will have a clustered configuration, install the Failover Clustering feature and form a cluster before performing the migration.
2. If the destination server will have a clustered configuration, create a number of cluster resource groups with client access points and cluster disks as appropriate to replicate the existing configuration. If possible, use the same resource group names for the source clusters and the destination clusters.
3. Install the iSCSI Target Server role service on the destination server.
4. Disconnect all the iSCSI initiators. This step is required to maintain consistent data on the virtual disks while they are being moved.
5. Run the Windows PowerShell script, iSCSITargetSettings.ps1, to capture the existing settings on the source server in an XML file. For a cluster, run the script on each node in the cluster or on each virtual computer object, as appropriate for the scope of the planned migration.

   The Windows PowerShell script displays the virtual disks that are eligible for migration and those that are not (for the snapshot-based reasons discussed previously).

Migration

The actual migration process includes the following steps:

1. Move the files for all the virtual disks that are eligible for migration from the source server to the destination server. If there are any file path changes, note the source to destination mapping.
2. In a cluster configuration, ensure that the destination path of the file copy is on a cluster disk and that the cluster disk has been assigned to a resource group. Note the resource group that owns the path.
3. If the file paths have changed between the source and the destination servers, open the settings .xml file in a text editor, and identify the **<MigrationDevicePath>** tags that need to be changed to reflect the new path.

4. In a cluster configuration, if the file path or the resource group name have changed between the source server and the destination server, open the settings .xml file in a text editor, and identify the **<MigrationResourceGroup>** tags that need to be changed to reflect the new resource group.
5. Run the Windows PowerShell script, iSCSITargetSettings.ps1, to import the settings to the destination server. In a cluster configuration, the destination server can be specified as a cluster node or as a virtual computer object. The cluster node or virtual computer object must be the owner of the resource group that is indicated in the settings .xml file.
6. If there are snapshot storage settings relevant to the new configuration, apply those settings manually.
7. If there are virtual disks that need to be attached as local disks, perform those actions.
8. If there are any iSNS settings that are relevant to the new configuration, apply those settings manually.
9. If there are any iSCSI target portal settings that are relevant to the new configuration, apply those settings manually.
10. If there are any iSCSI initiators that are configured to authenticate by using CHAP and Reverse CHAP, manually restore those settings.

Verification

The verification process for the migration involves the following steps:

1. Validate the iSCSI target portal settings by opening a Command Prompt window and typing **netstat.exe –nao | findstr 3260**. (This assumes that the default TCP port for the iSCSI protocol 3260 is used.) Alternatively, type **Get-WmiObject –Namespace root\wmi –Class WT_Portal** to cross-check the results.
2. Inspect the iSCSI Target Server configuration by using the Windows PowerShell cmdlet, **Get-IScsiServerTarget**
3. Inspect the iSCSI virtual disk configuration by using the Windows PowerShell cmdlet, **Get-IScsiVirtualDisk**
4. Validate the configuration for each iSCSI initiator that you expect to use with iSCSI Target Server by using the iscsicpl.exe UI tool or the iscsicli.exe command-line tool.

Impact of migration

The migration process does not impact or affect the source server. There are no resources or configuration settings that are altered or deleted as part of the migration process.

No servers in the enterprise, other than the destination servers, will be affected by the migration.

Client computers that are running as iSCSI initiators are expected to be explicitly disconnected during the migration to ensure data integrity. During the migration, the source server will be unavailable. When the migration process is complete, it is expected that the iSCSI initiators will log on to the destination server without any issues.

The downtime for the iSCSI initiators is expected to be proportionate to the time it takes to move the virtual disk files from the source server to the destination server, plus the time needed to restore the configuration settings and to establish the network identity of the destination server.

Permissions required for migration

Local Administrator permissions are required on the source and the destination server.

If the iSNS server has additional access control policies, permission to alter the iSNS settings are required as appropriate for the iSNS server.

To perform the migration process for the iSCSI initiators, permissions to log on and log off iSCSI sessions are required. For the iSCSI initiator, Local Administrator permissions are required.

For iSCSI initiators that are firmware based, such as a network interface with the option to boot from iSCSI, being at the actual console may be required to configure logon credentials or the network identity of the destination server if the authentication settings (CHAP and Reverse CHAP) have changed.

Estimated time duration

This section detail the various factors that impact how long a migration may take to complete.

Planning

The planning phase is expected to be influenced by the following factors:

- Stand-alone versus a cluster configuration. A cluster setup may require one to two hours to configure if all the validations are performed.
- Storage configuration. Understanding and configuring a storage array to host potentially huge files requires that you plan the spindle and volume configurations so that they use the tools that are provided by the storage subsystem vendor.
- Network identity. This planning involves understanding if the source server has specially or purposely configured IP addresses, if configuring Level-2 components (such as switches) is required, and if specific DNS or NetBIOS names need to be known to and cached by the iSCSI initiators.

Preparation

The preparation process involves understanding which settings (that are specific to the source server) cannot be automatically migrated, and gathering those settings. For each step in the preparation phase, the mechanism that is used to retrieve the settings depends on which step is applicable and which tool is used to recover those settings.

- Cluster resource group names and configuration. These settings can be gathered from the cluster administration tools and the user interfaces.
- iSCSI target portal configuration. These settings can be gathered by typing the following code at a command prompt: `PS > Get-WmiObject –Namespace root\wmi –Class WT_Portal`
- iSNS Server settings. These settings can be gathered by typing the following code at a command prompt: `PS > Get-WmiObject –Namespace root\wmi –Class WT_ISnsServer`
- CHAP and Reverse CHAP authentication settings. These settings cannot be automatically retrieved because the iSCSI target server does not offer a mechanism to retrieve passwords. These settings have been stored elsewhere in the enterprise, and they need to be retrieved independently.
- Locally mounted virtual disk settings.

Migration

The estimated time for the actual migration process is largely dominated by the time that it takes to move the virtual disk files from the source server to the destination server.

A network-based file copy, using a 1 GB link used at 50% for 1 TB of data, is estimated to take over five hours. Techniques that use a file transfer process involving external media, such as an External Serial Advanced Technology Attachment (eSATA) device, may take less time.

The execution of the Windows PowerShell import script is estimated to take few minutes for approximately 100 resources (with a combination of iSCSI target settings and virtual disk settings).

Verification

The estimated time for the verification is proportionate to the time it takes to reconnect or log on to the iSCSI initiators.

For each iSCSI initiator, the target portal needs to be reconfigured, credentials related to authentication settings must be entered (if required), and the sessions have to be logged on.

The estimated time is 5 to 15 minutes to verify each iSCSI initiator, depending on the process that is being used. iSCSI initiators can be verified through the iscsicpl.exe UI, through the iscsicli.exe command-line tool, or through ad hoc Windows Management Instrumentation (WMI)-based scripts).

See Also

Migrate File and Storage Services to Windows Server 2012 R2

Prepare to Migrate iSCSI Software Target

Migrate iSCSI Software Target

Verify the iSCSI Software Target Migration

Troubleshoot the iSCSI Software Target Migration

Roll Back a Failed iSCI Software Target Migration

Prepare to Migrate iSCSI Software Target

This topic discusses the tasks that are necessary before you start the migration process. The first step is to install the Windows Server Migration Tools. For more information, see File and Storage Services: Prepare to Migrate.

Prepare the destination server

The destination server is a computer that is configured and shipped by an OEM with Windows Storage Server 2012 pre-installed, or that is running Windows Server 2012 R2.

iSCSI Target Server hardware requirements for the destination server are the following:

- The amount of free disk space on the destination server must be sufficient to host the iSCSI virtual disk from the source server with adequate room for the snapshot storage.
- For clustered configurations, the resource groups that are created in the destination server must have associated cluster disks with adequate free space to host the iSCSI virtual disk from the source server.
- The destination server must have one or more network interfaces to be utilized for the iSCSI network traffic.

Installing the Failover Cluster feature in Windows Server 2012 R2 or Windows Storage Server 2012 or is required if the source server was configured with failover clusters. For more information, see the Failover Clustering Overview.

Back up the source server

Before you start migration, as a best practice, it is recommended that you back up the source server. For more information, see Windows Server Backup.

Prepare the source server

The following tasks are performed on the source server.

Cluster resource group configuration

Use the following steps to obtain the cluster resource groups:

1. Gather the resource groups that have iSCSI Software Target resources by using the following Windows PowerShell commands:

```
PS > Import-Module FailoverClusters
PS > $iSCSITargetResources = Get-ClusterResource | Where-Object
{ ( $_.ResourceType.Name -eq "Host" ) –or ($_.ResourceType.Name
-eq "WTDisk") }
PS > $iSCSITargetResources
```

2. From the cluster resources obtained in the previous step, gather the cluster disk dependencies by using the following Windows PowerShell commands:

```
PS > $Dependencies = &{ $iSCSITargetResources | Get-
ClusterResourceDependency  }
PS > $Dependencies
```

If the source server is running Windows Storage Server 2008, the following steps can be followed to gather the equivalent information:

1. Gather the iSCSI Software Target resources, and then gather the groups by using the following Windows PowerShell commands:

```
PS > $iSCSITargetResources = Get-WmiObject -NameSpace
root\mscluster -Authentication PacketPrivacy -Class
MsCluster_Resource -Filter "Type = `"WTDisk`" or Type =
`"Host`""
PS > $iSCSITargetResources
PS > $Groups = &{foreach($res in $iSCSITargetResources) { Get-
WmiObject -NameSpace root\mscluster -Authentication
PacketPrivacy -Query "associators of {$($res.__RELPATH)}  WHERE
ResultClass = MSCluster_ResourceGroup" }}
PS > $Groups
```

2. From the cluster resources obtained in the previous step, gather the cluster disk dependencies by using the following Windows PowerShell commands:

```
PS > $Dependencies = &{foreach($res in $iSCSITargetResources) {
Get-WmiObject -NameSpace root\mscluster -Authentication
PacketPrivacy -Query "associators of {$($res.__RELPATH)}  WHERE
ResultClass = MSCluster_Resource ResultRole = Dependent" }}
PS > $Dependencies
```

The resource groups obtained in step 1 have network name resources and IP addresses that need to be migrated to the destination server.

For information about how to migrate these settings, see Migrate IP Configuration to Windows Server 2012.

The cluster disk that you obtained in step 2 is the physical disk where the volumes reside that are hosting the iSCSI Software Target virtual disks.

To obtain the volumes from the cluster disk, use the following steps:

1. Obtain the disk signature of the cluster disk by using the following Windows PowerShell command:

```
PS > & cluster.exe res "<cluster resource name>" /priv
```

2. Obtain the **Win32_DiskDrive** object from the disk signature by using the following Windows PowerShell command:

```
PS > $DiskObj = Get-WmiObject -Namespace root\cimv2 -Class
Win32_DiskDrive -Filter "Signature = <disk signature>"
PS > $DiskObj
```

3. Obtain the **Win32_DiskDriveToDiskPartition** association by using the following Windows PowerShell command:

```
PS > $DiskToDiskPartition = Get-WmiObject -Namespace root\cimv2
-Class Win32_DiskDriveToDiskPartition | Where-Object {
$_.Antecedent -eq $DiskObj.__PATH }
PS > $DiskToDiskPartition
```

4. Obtain the **Win32_LogicalDiskToDiskPartition** association that points to the volume association by using the following Windows PowerShell command:

```
PS > Get-WmiObject -Namespace root\cimv2 -Class
Win32_LogicalDiskToPartition | Where-Object { $_.Antecedent -eq
$ DiskToDiskPartition.Dependent }
```

Steps 2–4 must be applied on the source server cluster node that currently owns the physical disk cluster resource.

iSCSI Target portal configuration

Use the following steps to obtain the portal associations:

1. Gather the configured portals association for the iSCSI target portal by using the following Windows PowerShell command:

```
PS> Get-WmiObject -Namespace root\wmi -Class WT_portal | Format-
List -Property Address,Listen,Port
```

2. The IP addresses that have the **Listen** state set to **True** are the IP addresses that an iSCSI initiator can use to reach the server. For more information about migrating the IP addresses, see Migrate IP Configuration to Windows Server 2012.

iSNS configuration

Gather the configured iSCSI Naming Services (iSNS) association for the server by using the following Windows PowerShell command:

```
PS> Get-WmiObject -Namespace root\wmi -Class WT_ISnsServer | Format-List -Property
ServerName
```

The server names that are listed need to be added to the list of iSNS servers that can be used to retrieve information about the iSCSI initiators in the enterprise.

CHAP and Reverse CHAP configuration

Gather the CHAPUserName and ReverseCHAPUserName association for the servers that are configured with CHAP and Reverse CHAP by using the following Windows PowerShell command:

```
PS > Get-WmiObject -Namespace root\wmi -Class WT_Host | Where-Object { ( $_.EnableCHAP )
-or ( $_.EnableReverseCHAP )  } | Format-List -Property
Hostname,CHAPUserName,ReverseCHAPUserName
```

The passwords that are used in conjunction with the credentials listed previously cannot be retrieved, and they must be known through other mechanisms.

Snapshot storage configuration

The snapshot storage configuration can be obtained by using the following Windows PowerShell command:

```
PS > & vssadmin.exe list shadowstorage
```

This command shows the volume snapshot shadow storage configuration for the entire source server. Not all the volumes listed may be relevant to the current iSCSI Software Target server configuration.

For the volumes that are relevant (that is, the volumes that host iSCSI virtual disks), the associated shadow storage volume is listed, in addition to the amount of disk space used with the maximum amount of configured space.

Disconnect the iSCSI initiators

Follow the instruction in the following section to disconnect the iSCSI initiators: Prepare other computers in the enterprise.

Capture the existing settings: stand-alone configuration

All of the settings on the iSCSI Software Target source server that are not hardware configuration specific and are not dependent on an IP address and the network identity of the server can be captured with the following Windows PowerShell commands:

Windows Server 2008 R2 and Windows Server 2008 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"

PS> Export-IscsiTargetServerConfiguration –FileName <settings XML file>
```

Windows Server 2012 R2 file path:

```
PS > cd "$ENV:SystemRoot\System32\WindowsPowerShell\V1.0\Modules\IscsiTarget"

PS> Export-IscsiTargetServerConfiguration –FileName <settings XML file>
```

If the procedure is performed on a source server that is running iSCSI Target 3.3 from a destination server that is prepared as illustrated in the previous sections, the settings can be captured using the following Windows PowerShell commands:

Windows Server 2012 R2 file path:

```
PS > cd "$ENV:SystemRoot\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget"

PS> Export-IscsiTargetServerConfiguration –FileName <settings XML file> -ComputerName
<source server computer name>
```

Windows Server 2008 R2 and Windows Server 2008 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"
```

```
PS> Export-IscsiTargetServerConfiguration –FileName <settings XML file> -ComputerName
<source server computer name>
```

At the end of the settings capture process, the Windows PowerShell script will display the set of VHD files that are eligible for migration. This list is needed for the destination server during migration.

Capture the existing settings: clustered configuration

Before capturing the iSCSI Software Target source server settings that are not hardware configuration specific, we recommend that all the resource groups with iSCSI target resources are moved to a single node in the cluster.

This can be accomplished by using the following Windows PowerShell commands. These commands assume that you previously followed the steps in Cluster resource group configuration.

```
PS > $iSCSITargetResources | Format-List -Property OwnerGroup
```

```
PS > foreach($Res in $iSCSITargetResources) { & cluster group  $Res.OwnerGroup
/moveto:$ENV:COMPUTERNAME }
```

After all the resource groups have been moved to a single node, the settings can be gathered by using the following Windows PowerShell commands:

Windows Server 2012 R2 file path:

```
PS > cd "$ENV:Programfiles\ISCSI Target"
```

```
PS> .\ Export-IscsiTargetServerConfiguration –FileName <settings XML file>
```

Windows Server 2008 R2 and Windows Server 2008 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"
```

```
PS> .\ Export-IscsiTargetServerConfiguration –FileName <settings XML file>
```

If the procedure is performed on a source server that is running iSCSI Target 3.2, the resources can be moved to a single node by using the following Windows PowerShell commands:

```
PS >  $Groups = &{foreach($res in $iSCSITargetResources) { Get-WmiObject -NameSpace
root\mscluster –Authentication PacketPrivacy -Query "associators of {$($res.__RELPATH)}
WHERE ResultClass = MSCluster_ResourceGroup" }}
```

```
PS > foreach($Group in $Groups) { & cluster group $Group.Name /moveto:<node name source
server> }
```

The iSCSI Target Server settings need to be gathered from a destination server that is prepared as illustrated in the previous sections. Run the script from a source server that is running iSCSI Target 3.3 by using the following Windows PowerShell command:

Windows Server 2012 R2 file path:

```
PS > cd "$ENV:Programfiles\ISCSI Target"

PS> .\ Export-IscsiTargetServerConfiguration –FileName <settings XML file> -ComputerName
<source server computer name>
```

Windows Server 2008 R2 and Windows Server 2008 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"

PS> .\ Export-IscsiTargetServerConfiguration –Export –FileName <settings XML file> -
ComputerName <source server computer name>
```

In the previous example, the source server computer name is the name of the node. At the end of the settings capture process, the Windows PowerShell script will display the set of VHD files that are eligible for migration. This list is needed for the destination server during migration.

Remove the network identity of the iSCSI Software Target computer

In a network with an iSCSI Software Target source computer, the identity of the server is known to iSCSI initiators in the form of NetBIOS names, fully qualified domain names (FQDN), or IP addresses. When a server is being replaced, as part of planning, a strategy to replace the server network identity must be devised. Possible scenarios include:

- Transfer the NetBIOS and FQDNs to the destination server, and then assign new IP addresses to the destination server.
- Create new NetBIOS and FQDNs for the destination server, and then assign the existing IP addresses to the destination server.
- Create new NetBIOS and FQDNs for the destination server, and then assign new IP addresses to the destination server.

Each scenario requires potentially updating information in the DNS server, Active Directory, or DHCP server, according to the methodology that is used to assign IP addresses and network names to the servers in the enterprise.

The intent of this step is to ensure that upon completion of the migration steps, the iSCSI initiators are able to locate the destination server (either through explicit reconfiguration, or implicitly through the computer name or IP address re-assignment).

For more information, see Migrate IP Configuration to Windows Server 2012.

Prepare the iSCSI initiator computers

The other computers in the enterprise that are affected by migration are the iSCSI initiators. The users of the computers that are acting as iSCSI initiators should be sent an outage notification. If

the iSCSI Software Target is being used as a boot node for the iSCSI initiator computers, the computers may be completely unusable for the duration of the migration.

Capture the session information

The information regarding the active session for an iSCSI Software Target source server can be obtained by using the following Windows PowerShell command:

```
PS > & iscsicli.exe sessionlist
```

This information is needed to disconnect the session.

Disconnect the session

The session can be disconnected by using the following Windows PowerShell command:

```
PS > & iscsicli.exe LogoutTarget <session id>
```

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Migrate an iSCSI Software Target

Migrate iSCSI Software Target

Verify the iSCSI Software Target Migration

Troubleshoot the iSCSI Software Target Migration

Roll Back a Failed iSCI Software Target Migration

Migrate iSCSI Software Target

This topic discusses the actual migration steps for iSCSI Software Target 3.2 or iSCSI Software Target 3.3 for both the stand-alone configuration and the clustered configuration:

Migrating iSCSI Software Target in a standalone configuration

The migration of iSCSI Target Server 3.3 or iSCSI Target Server 6.2 has equivalent steps, whether you are migrating from Windows Storage Server 2008 R2 or Windows Server 2012 or Windows Storage Server 2012 to Windows Server 2012 R2.

Establish network identity of the iSCSI Target Server computer

As part of the planning process, a strategy should have been devised regarding how iSCSI Target Server will be accessed from the network, based on key questions including but not limited to:

- Which computer name will be used?
- Which IP addresses on which subnet or set of network interfaces will be used?
- What relationship should be maintained between the IP addresses and computer name of the source server and the destination server? Will you keep the same addresses and names or create new ones?

Based on the desired final configuration, configuration changes are potentially needed in the following areas:

- The DHCP Server that might assign IP addresses to the destination iSCSI Target servers
- The DHCP Server that might assign IP addresses to the iSCSI initiators
- The DNS Server or Active Directory domain controller that might perform naming resolution services for the computers in the enterprise

Configure the iSCSI Target Server portal

After you have configured IP addresses for the network interfaces of the iSCSI Target Server computer, it is possible to verify the existing configuration by using the following Windows PowerShell command:

```
PS > $Portals = Get-WmiObject -Namespace root\wmi -Class WT_Portal | Where-Object {
$_.Listen }

PS > $Portals
```

The configuration of the access surface for iSCSI Target Server from the network can be restricted by disabling certain portals. For example, you can disable the fourth portal in the array returned in the previous step by using the following Windows PowerShell commands:

```
PS > $Portals[3].Listen = 0

PS > $Portals[3].Put()
```

The default port can also be changed from 3260 to any available TCP port on the destination server.

Configure iSNS settings

The iSNS servers that were configured for the source server can be configured for the destination server by using the following Windows PowerShell commands:

```
PS > $WT_ISnsServerClass =  Get-WmiObject -namespace root\wmi -class meta_class -filter
"__CLASS = 'WT_ISnsServer'"

PS > $WtiSNSInstanace = $WT_ISnsServerClass.CreateInstance()

PS > $WtiSNSInstanace.ServerName = "<iSNS computer name or IP>"

PS > $WtIsnsInstanace.Put()
```

Note

The set of iSNS servers that are configured for iSCSI Target Server was obtained during the preparation of the source server.

Configure storage

The destination server is expected to have sufficient storage space to host all of the virtual disks that are present on the source server.

The space does not need to be contiguous or in a single volume, and it does not need to replicate the same file system structure or volume mount point structure of the source server. The storage that is prepared to host the virtual disks must not be a nested volume, and it must be formatted with the NTFS file system.

Configure the Volume Shadow Copy Service

For the storage that was prepared in the previous step, it is appropriate to configure the Volume Shadow Copy Service, in case the default per-volume settings are not adequate. To inspect that current configuration, use the following Windows PowerShell command:

```
PS > & vssadmin.exe list shadowstorage
```

To modify the current configuration, use the following Windows PowerShell commands:

```
PS > & vssadmin.exe add ShadowStorage

PS > & vssadmin.exe delete ShadowStorage

PS > & vssadmin.exe resize ShadowStorage
```

Transfer the virtual disk

For all the files in the list of files that was captured in the source server preparation step, copy the files from the source server to the destination server. For more information, see Capture the existing settings: stand-alone configuration.

You will need the destination paths in the following steps. So if the absolute file path is different between the source server and the destination server, create a table with the mapping; for example:

Source path	Destination path
G:\WS08R2_OpsMgr2007_R2.vhd	H:\VHDS\WS08R2_OpsMgr2007_R2.vhd
F:\Dynamic_Spanned_GPT_2.vhd	D:\DYNVHDS\Dynamic_Spanned_GPT_2.vhd

Import the iSCSI Software Target settings in a stand-alone configuration

To import the iSCSI Software Target settings in a stand-alone configuration, you need the .xml file that you previously created. For more information, see Capture the existing settings: stand-alone configuration.

If there is no change in the absolute path of the virtual disk files, the import process can be performed by using the following Windows PowerShell commands:

```
PS > cd "$ENV:SystemRoot\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget"

PS> .\ iSCSITargetSettings.PS1 -Import -FileName <settings XML file>
```

If the absolute path is different between the source server and the destination server, before you import the settings, the settings .xml file needs to be altered to reflect the new path.

Locate the records for the virtual disk, and alter the path in the <MigrationDevicePath> tag to reflect the absolute file path in the destination server:

```
<iSCSIVirtualDisk>

 <DevicePath>F:\Dynamic_Spanned_GPT_2.vhd</DevicePath>

  <MigrationDevicePath>D:\DYNVHDS\Dynamic_Spanned_GPT_2.vhd</MigrationDevicePath>

</iSCSIVirtualDisk>
```

After the XML has been altered to reflect the path in the destination server, you can import the settings by using the Windows PowerShell commands previously presented.

Configure shadow storage for the virtual disks

If certain virtual disks have shadow storage requirements that are different than the ones configured in the section Configure the Volume Shadow Copy Service, it is possible to alter the default or previously configured settings by using the following Windows PowerShell commands:

```
PS > $VirtDisk = Get-WmiObject -Namespace root\wmi -Class WT_Disk | Where-Object {
$_.DevicePath -eq '<full path of virtual disk>' }

PS > $VirtDisk.SnapshotStorageSizeInMB = <new size>

PS > $VirtDisk.Put()
```

Configure CHAP and Reverse CHAP

The authentication settings for iSCSI Target Server that are configured with CHAP and Reverse CHAP need to be manually configured. The list of targets that require CHAP and Reverse CHAP configuration is listed at the end of the import script, as described in the section Import the iSCSI Software Target settings in a standalone configuration.

To configure the CHAP and Reverse CHAP settings, use the following Windows PowerShell commands:

```
PS > $Target = Get-WmiObject -Namespace root\wmi -Class WT_Host | Where-Object {
$_.HostName -eq '<name of the target>' }

PS > $Target.EnableCHAP = 1

PS > $Target.CHAPUserName = "<user name>"

PS > $Target.CHAPSecret = "<CHAP Secret>"

PS  $Target.Put()
```

Migrating iSCSI Software Target in a failover cluster

The migration process for the failover cluster configuration is largely identical to migrating a stand-alone configuration, with the following differences:

- You will migrate resource groups instead of merely establishing the network identity of the server.
- You will use different Windows PowerShell commands to import the resource groups.

Migrate resource groups

This step replaces the “Establishing the network identity of iSCSI Target Server” step when you migrate a stand-alone configuration because the network identity of an iSCSI Target server in a cluster is given by the union of the client access point. (A client access point in the cluster is the logical union of a network name resource and one or more IP addresses that are assigned to the network name resource.)

Assuming the initial cluster resource groups and network names were configured in the default state, those can be recreated by using the following Windows PowerShell command:

```
PS > Add-ClusterServerRole – Name <resource group name>
```

Use this command for each of the resource groups that were in the original configuration. If the default client access point configuration does not match the initial configuration (for example, because the network name is bound to the incorrect cluster network, or the configuration required statically assigned IP addresses), modifications can be made. For more information, see Migrate Cluster Roles to Windows Server 2012 R2.

After the resource groups have been created, clustered disks must be assigned to the network resources to match the configuration that you captured. For more information, see the Cluster resource group configuration section.

Import the iSCSI Software Target settings in a failover cluster

Follow these instructions to import settings in a failover cluster configuration. (This information differs from the how you would import settings in a stand-alone configuration.)

A prerequisite for the import phase is to have all of the resource groups that will host iSCSI Target Server resources owned by the same cluster node. Use the following Windows PowerShell command to validate the current ownership:

```
PS > Get-ClusterGroup
```

If there is no change in the absolute path of the virtual disk files, the import process can be performed by using the following commands:

```
PS > cd "$ENV:SystemRoot\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget"

PS> .\iSCSITargetSettings.PS1 -Import -FileName <settings XML file>
```

If the absolute path is different between the source server and the destination server, before you import the settings, the settings .xml file needs to be altered to reflect the new path.Locate the records for the virtual disk, and alter the path in the **<MigrationDevicePath>** tag to reflect the absolute file path in the destination server:

```
<iSCSIVirtualDisk>

      <DevicePath>F:\Dynamic_Spanned_GPT_2.vhd</DevicePath>

      <MigrationDevicePath>D:\DYNVHDS\Dynamic_Spanned_GPT_2.vhd</MigrationDevicePath>

    </iSCSIVirtualDisk>
```

After the XML has been altered to reflect the path in the destination server, you can import the settings by using the Windows PowerShell commands.

Migrate iSCSI Target Server Providers

This section provides details about migrating iSCSI Target Server Virtual Disk Service (VDS), Volume Shadow Copy Service (VSS), and SMI-S providers.

Migrate VDS and VSS hardware providers

If you are upgrading from Windows Server 2012 to Windows Server 2012 R2, the previous storage provider is automatically upgraded to Windows Server 2012 R2, and no additional action is required.

If you are upgrading from Windows Server 2008 R2 to Windows Server 2012 R2, you must first manually uninstall the currently installed iSCSI Target storage provider. Because iSCSI Target storage provider was installed separately from Windows Server 2008 R2, the provider cannot be automatically upgraded. When the iSCSI Target storage provider is uninstalled, do the following:

a. Upgrade the server to Windows Server 2012 R2.
b. Install the **iSCSI Target Storage Provider (VDS and VSS hardware providers)** role service on the upgraded server. You can do this using **Server Manager dashboard**.
c. The iSCSI VDS and VSS storage providers must be configured to run under the administrative credentials of the iSCSI Target Server. For more information, see iSCSI Target Block Storage, How To.

Migrate SMI-S providers

You must first manually uninstall the currently installed SMI-S provider for Windows Server 2012. Because the SMI-S provider was installed separately from Windows Server 2012, the provider cannot be automatically upgraded. When the SMI-S provider is uninstalled, do the following:

1. Upgrade the server to Windows Server 2012 R2. The SMI-S provider is automatically installed along with the iSCSI Target Server role service.
2. From any System Center Virtual Machine Manager (VMM) or SMI-S management client, unregister and reregister using the appropriate credentials. For information on configuring the SMI-S provider using VMM, see Configuring an SMI-S Provider for iSCSI Target Server. For information about configuring the SMI-S provider using the SMI-S management client, see Register-SmisProvider.

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Migrate an iSCSI Software Target

Prepare to Migrate iSCSI Software Target

Verify the iSCSI Software Target Migration

Troubleshoot the iSCSI Software Target Migration

Roll Back a Failed iSCI Software Target Migration

Verify the iSCSI Software Target Migration

This topic discusses the steps you can use to verify that the migration successfully completed.

Verifying the destination server configuration

To verify that the destination server has been properly configured after migration, you can verify the listening endpoints and connectivity and run a scan with the Best Practices Analyzer.

Verify the listening endpoints

On the iSCSI Target destination server, you can validate that the target portals have been configured as desired by using the following Windows PowerShell command:

```
PS > & netstat.exe -nao | findstr 3260 | findstr LISTENING
```

```
  TCP    10.121.26.107:3260     0.0.0.0:0              LISTENING       1560

  TCP    10.121.26.126:3260     0.0.0.0:0              LISTENING       1560

  TCP    [2001:4898:0:fff:0:5efe:10.121.26.126]:3260  [::]:0                 LISTENING
1560

  TCP    [2001:4898:f0:1001:f063:8fc5:52e6:2310]:3260  [::]:0                 LISTENING
1560
```

The list of IP addresses and port pairs in the listening state needs to match the desired set of target portals.

Note
If ports other than the default 3260 are being used, the command needs to be altered to reflect the alternate IP ports.

Verify the basic connectivity

To validate that the iSCSI Target Server computer is reachable from other computers on the network, from a computer that has the Telnet Client feature installed, use the following Windows PowerShell command:

```
PS > telnet.exe <iSCSI Software Target machine name or IP> 3260
```

If there is a successful connection, Telnet Client will show a blinking cursor at the top of the window. Press any key to close Telnet Client.

Perform a Best Practices Analyzer scan

To verify that iSCSI Target Server is optimally configured on Windows Server 2012 or Windows Storage Server 2012 after migration, we recommend that you run a Best Practices Analyzer (BPA) scan on the role.

BPA is a server management tool that is available in Windows Server 2012. After the migration of iSCSI Target 3.3 is complete, BPA can help you ensure that your server is configured according to best practices. You can use the Server Manager console UI or Windows PowerShell to perform BPA scans and view results. For detailed information about how to scan your role and view results, see Run Best Practices Analyzer Scans and Manage Scan Results.

Verifying the configuration of iSCSI initiator computers

Validating the migration of iSCSI Software Target to the destination server includes ensuring that the iSCSI initiators can discover and fully access all features of the iSCSI protocol.

Verify that the iSCSI initiators can discover iSCSI Target Server

To verify that the iSCSI initiators can discover iSCSI Target Server, use the following Windows PowerShell commands:

```
PS > & iscsicli AddTargetPortal <ip-address> 3260
```

```
PS > & iscsicli.exe ListTargets
```

If the commands execute without errors, the initiator is capable of discovering the targets that are offered by the server

Verify that the iSCSI initiators can log on

The second step is to verify that the iSCSI initiators are able to log on to the iSCSI targets that are exposed by iSCSI Target Server. This can be accomplished by using the following Windows PowerShell command:

```
PS > & iscsicli.exe  LoginTarget <target IQN> T <ip address> 3260 Root\ISCSIPRT\0000_0 *
* * * * * * * * * * * * *
```

Note

If you are using CHAP and Reverse CHAP authentication, you may need to specify more parameters. For more information, consult the documentation in the iscsicli.exe.

If the command executes without errors, the iSCSI initiator has successfully logged on to the target, and the disks are exposed to iSCSI Target Server.

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Migrate an iSCSI Software Target

Prepare to Migrate iSCSI Software Target

Migrate iSCSI Software Target

Troubleshoot the iSCSI Software Target Migration

Roll Back a Failed iSCI Software Target Migration

Troubleshoot the iSCSI Software Target Migration

Troubleshooting iSCSI Software Target migration issues involves first viewing the contents of the Windows Server Migration Tools deployment log and the result objects. For more information, see Locate the deployment log file and View the content of Windows Server Migration Tools result objects.

Understanding the messages from the iSCSI Target Migration tool

The iSCSI migration tool (iSCSITargetSettings.ps1) does not produce a log file, but it prints diagnostics messages on the console. These messages show the outcome of the operations that are being attempted and performed.

For example, the following message shows information about saved settings:

```
PS C:\Windows\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget>

.\iSCSITargetSettings.PS1 -Export -FileName $env:temp\test00000000.xml


Number of Target(s) saved in the Export settings: 4.

Target Names:

   test000

   test001

   test002

   test1111


Number of Virtual Disk(s) saved in the Export settings: 3.

Virtual Disk DevicePaths:

   s:\test000.vhd

   t:\test000.vhd

   z:\test000.vhd


Number of Virtual Disk(s) NOT saved in the Export settings: 0.

Virtual Disk DevicePaths:
```

The following message shows that not all the virtual disks are eligible for migration:

```
PS D:\Program Files\ISCSI Target> .\iSCSITargetSettings.PS1 -Export -FileName
$env:temp\test00000001.xml


Number of Target(s) saved in the Export settings: 4.

Target Names:

   test000

   test001

   test002
```

```
   test1111

Number of Virtual Disk(s) saved in the Export settings: 3.
Virtual Disk DevicePaths:
   s:\test000.vhd
   t:\test000.vhd
   z:\test000.vhd

Number of Virtual Disk(s) NOT saved in the Export settings: 1.
Virtual Disk DevicePaths:
    \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy{B6B3C77C-93CC-11DF-B3FE-
001CC0C60A6E}\test000.vhd
```

The following message shows information about the settings restore phase:

```
PS C:\Program Files\ISCSI Target> .\iSCSITargetSettings.PS1 -Import -file
$env:temp\test00000000.xml

Importing settings from file
'E:\Users\administrator\AppData\Local\Temp\test00000001.xml'.
The operation may take a long time.

Number of Target(s) imported from the Import settings: 4.
Targets:
    test000
    test001
    test002
    test1111

Number of Virtual Disk(s) imported from the Import settings: 3.
Virtual Disk:
   s:\test000.vhd
   t:\test000.vhd
   z:\test000.vhd
```

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Migrate an iSCSI Software Target

Prepare to Migrate iSCSI Software Target

Migrate iSCSI Software Target

Verify the iSCSI Software Target Migration

Roll Back a Failed iSCSI Software Target Migration

Roll Back a Failed iSCSI Software Target Migration

If iSCSI initiators have successfully reconnected to the iSCSI Target Server computer, the migration is successful and complete. This topic discusses the tasks that should be performed in the event of a failed migration.

Restoring the role if the migration failed

If migration does not complete successfully, a rollback procedure is required to undo any changes to the source server, other servers, and client computers, and then restore the source server back into service.

Rollback requirements

The rollback procedure requires that the source server is available in the same state as it was after the “Remove the network identity of the iSCSI Software Target server” step in the “Prepare your source server” section. For more information, see Remove the network identity of the iSCSI Software Target server.

During the source server preparation steps, none of the steps performed permanently changed the existing configuration of the server because all of the operations were substantially read operations.

The estimated time to complete the rollback is equivalent to the time that it takes to re-establish the network identity of the source server. This operation may require rolling back changes to the DHCP servers, DNS server, or Active Directory Domain controllers.

Roll back iSCSI initiators on other computers

The other computers in the enterprise that are affected by migrating ISCSI Software Target are the iSCSI initiators.

In the case of a rollback, the iSCSI initiators that were configured to log on to the destination server need to be rolled back to the source server. Use the following Windows PowerShell commands:

1. To log out of an existing iSCSI session:

   ```
   PS > & iscsicli.exe sessionlistPS > & iscsicli.exe LogoutTarget <session id>
   ```

2. To discover the iSCSI Software Target source server:

   ```
   PS > & iscsicli AddTargetPortal <source server ip address> 3260PS > iscsicli.exe
   ListTargets
   ```

3. To log on to the targets on the iSCSI Software Target source server:

   ```
   PS > & iscsicli.exe  LoginTarget <target IQN> T < source server ip address> 3260
   Root\ISCSIPRT\0000_0 * * * * * * * * * * * * *
   ```

Roll back iSCSI Software Target on a stand-alone source server

This step will undo the network identity removal that is described in “Remove the network identity of the iSCSI Software Target server”.

Possible scenarios include:

- Restore the NetBIOS fully qualified domain name to the source server, and assign the required IP addresses to the source server.
- Restore any DNS assignments (for example, reverse lookup and DHCP assignment).
- Restore any identities that were previously assigned in Active Directory.

Each scenario requires potentially updating information in the DNS server, Active Directory, or DHCP server, according to the methodology that is used to assign IP addresses and network names to the servers in the enterprise.

The intent of this step is to ensure that upon completion of the migration steps, the iSCSI initiators are able to locate the source server (either through explicit reconfiguration, or implicitly through the computer name or IP address reassignment).

Roll back iSCSI Software Target on a clustered source server

Rolling back iSCSI Software Target on a clustered source server requires two steps:

Step 1: Roll back cluster network name changes

This step will undo the network identity removal described in “Remove the network identity of the iSCSI Software Target server”.

In a clustered configuration, network names are established by the server principal name that is assigned in Active Directory to the cluster when the cluster was formed.

To re-establish network names that were possibly deleted or retired, the cluster administration utilities must be used. For more information, see Migrating Settings to a Failover Cluster Running Windows Server 2008 R2.

Step 2: Move resource groups to the preferred owner node

After the client access points have been re-established, the resource groups need to be moved back to their preferred owner node.

The resource groups were moved to a single node as part of the steps performed in “Capture the existing settings: clustered configuration”.

To move the resource groups back to their preferred owner node, use the following Windows PowerShell command:

```
PS > & cluster.exe /cluster:<cluster name> GROUP <group name> /moveto:<node name>
```

Note

The group name and the node names were obtained during the previous preparation tasks.

Roll back iSCSI Target Server on a stand-alone destination server

To roll back iSCSI Target Server on a stand-alone destination server that is running Windows Server 2012 or Windows Storage Server 2012, uninstall the **iSCSI Target Server** role service using Server Manager.

Roll back iSCSI Target Server on a clustered destination server

To roll back iSCSI Target Server on a destination server that is running Windows Server 2012 or Windows Storage Server 2012 in a clustered configuration, first remove any client access point that was created for iSCSI Target Server and then uninstall the **iSCSI Target Server** role service using Server Manager.

Retiring iSCSI Software Target on a source server

Retiring iSCSI Software Target 3.2 or iSCSI Software Target 3.3 on your source server requires using the following Windows PowerShell commands:

Retire iSCSI Software Target

1. Find the package GUID:

   ```
   PS > Get-WmiObject -Class Win32_product | Where-Object { $_.packageName -match
   'iscsitarget'}
   ```

2. Uninstall the package:

   ```
   PS > & msiexec /uninstall <package GUID> /qr
   ```

Retiring a source server

In a stand-alone configuration, there are no particular procedures for retiring the source server. In a clustered configuration, the client access points that are devoted to iSCSI Software Target access can be removed by using the following Windows PowerShell command:

```
PS > Remove-ClusterGroup -Name <resource group name> -RemoveResources -Force
```

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Migrate an iSCSI Software Target

Prepare to Migrate iSCSI Software Target

Migrate iSCSI Software Target

Verify the iSCSI Software Target Migration

Troubleshoot the iSCSI Software Target Migration

File and Storage Services: Migrate Network File System

This topic describes how to migrate Network File System shares and settings from previous versions of Windows Server to Windows Server 2012 R2.

Network File System Migration overview

You can migrate Network File System (NFS) from a server running Windows Server 2012, Windows Server 2008 R2, Windows Server°2008, or Windows Server°2003°R2 to a server running Windows Server 2012 R2 using the procedures described in this topic. Some of the methods that are available for migrating NFS are the following:

- You can gather output from NFS servers running on previous versions of Windows Server, and then modify and use this information to input into the new NFS server running Windows Server 2012 R2. This can be done using the NFS cmdlets in Windows PowerShell, or using command tools such as **nfsshare** and **nfsadmin**.
- You can gather output from the NFS servers running on previous versions of Windows Server, and then use this information as a reference when manually configuring NFS settings for the new NFS server running Windows Server 2012 R2. This can be done using the NFS cmdlets in Windows PowerShell or the File Server Administration Console in Server Manager.

Migrating NFS Server from Windows Server°2012 to Windows Server°2012°R2

This section explains how to migrate NFS shares and permissions from Windows Server 2012 to Windows Server 2012 R2. Introduced in Windows Server 2012, the NFS cmdlets in Windows PowerShell allow you to manage NFS shares and settings, export shares and configuration metadata to .xml files, and then import the files into Windows Server 2012 R2. During this process, UNIX or Linux-based style group and password files are copied to Windows Server 2012 R2. If you used Active Directory Lightweight Directory Services (AD°LDS) to configure name mapping, see Active Directory Lightweight Directory Services Overview.

Export the server configuration

Before starting the export process, you must first create a directory (for example C:\tmp) where all the files will be exported.

Open Windows PowerShell, and to export the NFS server configuration information, type:

```
PS C:\tmp> Get-NfsServerConfiguration | Export-Clixml NfsServerConfig.xml
```

Export NFS shares

To export the NFS share settings information, open Windows PowerShell, and type the following. Note that this procedure does not include exporting the NFS share permissions.

```
PS C:\tmp> Get-NfsShare | Export-Clixml NfsShares.xml
```

Next, update the host configuration information using the following steps. You can ignore these steps if the net name and host names are going to remain the same.

1. Open the file where the exported shares are located (for example, c:\tmp\NfsShares.xml).
2. Find the network name and host name, and then rename them as appropriate.
3. If necessary, update the location of the directory path.
4. Save the file that contains the exported shares (such as c:\tmp\NfsShares.xml).

Export NFS share permissions

To export the NFS share permissions for all the NFS shares, type:

```
PS C:\tmp> Get-NfsShare | Get-NfsSharePermission | Export-Clixml NfsSharePermission.xml
```

Next, update the host configuration information using the following steps. You can ignore these steps if the net name and host names are going to remain the same.

1. Open the file where the exported permissions are located (for example, c:\tmp\NfsSharePermission.xml).
2. Find the network name and host name, and then rename them as appropriate.

3. If necessary, update the location of the directory path.
4. Save the file that contains the exported permissions (such as c:\tmp\NfsShares.xml).

Copy local mapping data

If you are using the UNIX or Linux-based local password and group files to map between UNIX and Linux-based users and Windows, copy the following files from Windows Server 2012. You can ignore this step if you are not using UNIX or Linux-based local password and group files.

```
PS C:\tmp> COPY %SystemRoot%\system32\drivers\etc\passwd C:\tmp

PS C:\tmp> COPY %SystemRoot%\system32\drivers\etc\group C:\tmp
```

Export identity mapping

In Windows PowerShell, type the following to display identity mapping information (such as Lightweight Directory Access Protocol or AD LDS) used by the NFS Server. This information must be manually recreated in Windows Server 2012 R2. If no identity mapping stores are configured, you can ignore this step.

```
PS C:\tmp> Get-NfsMappingStore | Export-Clixml nfsmappingstore.xml
```

Note

The group and user identity mapping are expected to remain the same after the migration.

Export netgroups and client groups

Configuring netgroups and client groups makes it easier to manage computer and user authentication. In Windows PowerShell, type the following to display information about netgroups and client groups, which can then be exported to Windows Server 2012 R2.

```
PS C:\tmp> Get-NfsNetgroup | Export-Clixml nfsnetgroup.xml

PS C:\tmp> Get-NfsNetgroupStore | Export-Clixml nfsnetgroupstore.xml

PS C:\tmp> Get-NfsClientGroup | Export-Clixml nfsclientgroup.xml
```

Importing NFS shares and settings from Windows Server°2012 to Windows Server°2012°R2

This section describes how to import NFS shares and settings that you exported from Windows Server 2012 to Windows Server 2012 R2. First, create a directory (for example C:\tmp) on the computer running Windows Server 2012 R2 and copy all the files exported from Windows Server 2012.

Note

The settings for NFS shares are metadata used over existing volumes and directories. Therefore, you should make sure the data and directory structure are correct before NFS

share settings are applied. After the directory structure is in place, you can proceed to the following procedure. For more information about data migration, see Impact of data migration by copying data and shared folders.

Import the server configuration

Before importing the server configuration, make sure that you have installed the **Server for NFS** role service in Server Manager. To import the server configuration, open Windows PowerShell, and type:

```
PS C:\tmp> Import-Clixml NfsServerConfig.xml | Set-NfsServerConfiguration
```

Restart **Server for NFS** by using either **Control Panel** or by typing `Restart-Service NfsService` at a command prompt.

Import NFS shares

Before performing this step, make sure that the directory structure is already in place and that the Nfsshares.xml file is updated with the appropriate location, server names, and any additional important information.

To import NFS share settings, open Windows PowerShell, and type:

```
PS C:\tmp> Import-Clixml NfsShares.xml | %{New-NfsShare -Name $_.Name -Path $_.Path -
NetworkName $_.NetworkName -EnableAnonymousAccess
$_.AnonymousAccess -AnonymousUid $_.AnonymousUid -AnonymousGid $_.AnonymousGid -
EnableUnmappedAccess
$_.UnmappedUserAccess -Authentication $_.Authentication}
```

You should resolve any errors before proceeding to the next step.

Import NFS share permissions

Before performing this step, make sure that the Nfssharepermission.xml file is updated with the correct server names. To import NFS share permissions, open Windows PowerShell, and type:

```
PS C:\tmp> Import-Clixml NfsSharePermission.xml | foreach { $_ |Grant-NfsSharePermission}
```

Import local mapping data

If UNIX and Linux-based local password and group files are used for mapping between UNIX and Linux users and Windows, copy the following files (which were exported from Windows Server 2012). You can ignore this step if you do not use UNIX and Linux-based password and group files.

```
PS C:\tmp> COPY  C:\tmp\passwd %SystemRoot%\system32\drivers\etc\passwd
PS C:\tmp> COPY C:\tmp\group %SystemRoot%\system32\drivers\etc\group
```

Import non-local identity mapping

If you are using methods, such as LDAP or AD LDS, to configure identity mapping, use the following Windows PowerShell script to import the .xml file:

```
PS C:\tmp> Import-Clixml nfsmappingstore.xml | Set-NfsMappingStore
```

Import netgroups and client groups

In Windows PowerShell, type the following to export netgroups and client groups to Windows Server 2012 R2:

```
PS C:\tmp> Import-Clixml nfsnetgroup.xml | Set-NfsNetgroup

PS C:\tmp> Import-Clixml nfsnetgroupstore.xml | Set-NfsNetgroupStore

PS C:\tmp> Import-Clixml nfsclientgroup.xml | Set-NfsClientGroup
```

After the netgroups and client groups are defined, permission to access shares that an NFS server exports can be configured using the Grant-NfsSharePermission Windows PowerShell cmdlet. Some examples for granting share permissions are shown in the following generated information in Windows PowerShell:

```
PS C:\> New-NfsClientgroup -ClientGroupName MIGRATION -AddMember 'MACHINE1','MACHINE2'

PS C:\> Get-NfsClientgroup MIGRATION

ClientGroupName                                          ClientGroupMembers

---------------                                          ------------------

MIGRATION                                                {MACHINE1, MACHINE2}

PS C:\> Grant-NfsSharePermission -Name NfsTestShare1 -ClientName MIGRATION -ClientType
clientgroup -Permission readonly

PS C:\> Get-NfsSharePermission NfsTestShare1

Name                  ClientName             Permission        AllowRootAccess

----                  ----------             ----------      ---------------

NFSTestShare1          MIGRATION                READ              False
```

If you are using Unmapped UNIX User Access (UUUA), see NFS Identity Mapping in Windows Server 2012, which provides information about the various methods of identity mapping. You should note that both Windows Server 2012 R2 and Windows Server 2012 support UNIX and Linux-based password and group files.

NFS server and share settings migration from Windows Server 2012 to Windows Server 2012 R2 is complete.

Migrating NFS Server from Windows Server°2008°R2, Windows Server°2008, or

Windows Server°2003°R2 to Windows Server°2012°R2

This section describes how to migrate NFS shares and permissions from Windows Server 2008 R2 and earlier versions of the Windows Server operating system to Windows Server 2012 R2. Using the command-line tools, **nfsshare** and **nfsadmin**, you can export NFS shares and settings, and then import the files into Windows Server 2012 R2.

Get server configuration

To retrieve information from the NFS server configuration, type the following at a command prompt:

```
C:\tmp> nfsadmin server
```

After running the command, create a copy of the information that is generated. An example of output from Windows Server 2008 R2 follows:

```
The following are the settings on localhost


Locking Daemon Grace Period : 45 seconds

Activity logging Settings   :

Protocol for Portmap        : TCP+UDP

Protocol for Mount          : TCP+UDP

Protocol for NFS            : TCP+UDP

Protocol for NLM            : TCP+UDP

Protocol for NSM            : TCP+UDP

Protocol for Mapping Server : TCP+UDP

Protocol for NIS            : TCP+UDP

Enable NFS V3 Support       : Enabled

Renew Authentication        : Enabled

Renew Authentication Interval : 600 seconds

Directory Cache             : 128 KB

Translation File Name       :

Dot Files Hidden            : Disabled

Case Sensitive Lookups      : Enabled

NTFS Case                   : Preserve Case

NetGroup Source             : none

NIS Server                  :

NIS Domain                  :
```

```
LDAP Server or AD Domain    :

LDAP naming context (DN)    :
```

Collect NFS shares information

You can display NFS share settings by using the following commands. You should note that running this command does not display NFS share permissions.

To retrieve the list of NFS shares configured in the server, type the following at a command prompt:

```
C:\tmp> nfsshare
```

To retrieve detailed information for each NFS share listed after using the preceding command, type:

```
C:\tmp> nfsshare <share-name>
```

Example output:

```
C:\tmp> nfsshare

        share1 = C:\shares\share1

        share2 = C:\shares\share2

C:\tmp> nfsshare share1

   Alias = share1

   Path = C:\shares\share1

   Supported security flavors are SYS:KRB5:KRB5I

   Encoding = ansi

   UNMAPPED UNIX USER access allowed

   ANONYMOUS access disallowed

   Anonymous UID = -2

   Anonymous GID = -2

   HOST ACCESS :

      ALL MACHINES            read-write     Root Access Allowed      ansi
```

Note

Kerberos authentication was introduced in Windows Server 2008 R2 for use with NFS, and therefore, so it is not available in earlier versions of the operating system.

Collect identity mapping and group identifier information

To display identity mapping settings (such as the Network Information Service [NIS] server, NIS domain, and LDAP or AD LDS information), type the following at a command prompt:

```
C:\tmp> nfsadmin server
```

To display the identity mapping methods that are used, type:

```
C:\tmp> nfsadmin mapping
```

Example output:

```
C:\tmp> nfsadmin mapping

The following are the settings on localhost

Mapping Server Lookup       : Disabled

Mapping Server              :

AD Lookup                   : Disabled

AD Domain                   :
```

To display the names of all client groups, type:

```
C:\tmp\nfsadmin server listgroups

C:\tmp\nfsadmin server listmembers <client group name>
```

Reconfiguring NFS shares and settings from Windows Server°2008°R2, Windows Server°2008, or Windows Server°2003°R2 to Windows Server°2012°R2

This section explains the process of reconfiguring the NFS shares and settings that you exported from Windows Server 2008 R2 or previous versions of Windows Server to Windows Server 2012 R2. You can reconfigure NFS shares and settings using the **nfsshare** or **nfsadmin** command tools, or, if you are migrating from Windows Server 2008 R2, you can use the NFS cmdlets in Windows PowerShell.

Before you import NFS shares and settings, make sure that you have installed the **Server for NFS** role service in **Server Manager** on the computer running Windows Server 2012 R2.

The settings for NFS shares are metadata used over existing volumes and directories. Therefore, you should make sure the data and directory structure are correct before NFS share settings are applied. After the directory structure is in place, you can proceed to the following procedure. For more information about data migration, see Impact of data migration by copying data and shared folders.

Set up the NFS server configuration

To configure the NFS server, type the following in Windows PowerShell. Instructions for setting up the NFS server using the **nfsadmin** command are provided later in this section.

In Windows PowerShell, type:

```
PS C:\tmp> Set-NfsServerConfiguration  -[parameters as displayed below]
```

Windows Server°2008°R2 output	Equivalent Windows PowerShell cmdlet parameters in Windows Server°2012°R2
Locking daemon grace period	GracePeriodSec
Protocol for Portmap	PortmapProtocol
Protocol for Mount	MountProtocol
Protocol for NFS	NfsProtocol
Protocol for NLM	NlmProtocol
Protocol for NSM	NsmProtocol
Protocol for Mapping Server	MapServerProtocol
Protocol for NIS	NisProtocol
Enable NFS V3 support	EnableNFSV3
Renew Authentication	EnableAuthenticationRenewal
Renew Authentication Interval	AuthenticationRenewalIntervalSec
Directory Cache	DirectoryCacheSize
Translation File Name	CharacterTranslationFile
Dot Files Hidden	HideFilesBeginningInDot
Activity Logging Setting	LogActivity

Notes

In Windows Server 2012 R2, there is a new parameter *LeasePeriodSec* for the Set-NfsServerConfiguration Windows PowerShell cmdlet. When setting the *GracePeriodSec* value, make sure that the *LeasePeriodSec* value is set to 50 percent of *GracePeriodSec*.

Case-sensitive file lookups and case-sensitive preservation can no longer be configured in Windows Server 2012 R2 because they are now system-wide Windows settings.

If you prefer to use the **nfsadmin** command tool, type the following at a command prompt:

```
C:\tmp> nfsadmin server config config_options
```

For a detailed list of configuration options for **nfsadmin**, type **nfsadmin server /?**.

Configure NFS shares

To configure NFS shares using the information you previously gathered on the shares, type the following in Windows PowerShell. Instructions for configuring NFS shares using the **nfsshare** command are provided later in this section.

```
PS C:\tmp> New-NfsShare <parameters>
```

Windows Server output	Equivalent NfsShare cmdlet parameters in Windows Server°2012°R2
Alias	-Name
Path	-Path
Encoding	-LanguageEncoding
Anonymous access	-EnableAnonymousAccess
Anonymous UID	-AnonymousUID
-Anonymous GID	-AnonymousGID
Host access	-Permission, -AllowRootAccess, …

An example of configuring an NFS share follows:

```
PS C:\> New-NfsShare –Name roshare – Path C:\shares\roshare =AnonymousUid -2 –
AnonymousGid -2 –LanguageEncoding ANSI –EnableAnonymousAccess $false –
EnableUnmappedAccess $false –AllowRootAccess $false
```

If you prefer to use the **nfsshare** command tool, type the following at a command prompt:

```
C:\tmp> nfsshare sharename=drive:path [ -o options ]
```

For a detailed list of configuration options for **nfsshare**, type **nfsshare server /?**.

Configure identity mapping and group identifier information

Using the information you gathered earlier for identity mapping, type the following in Windows PowerShell to configure ID mapping:

```
PS C:\tmp> Set-NfsMappingStore <Parameters>

PS C:\tmp> Set-NfsMappedIdentity <Parameters>
```

An example of configuring identity mapping follows:

```
PS c:\tmp> Set-NfsMappingStore -EnableADLookup $true -ADDomainName "Contoso.com"
```

If you prefer to use the **nfsadmin** command tool, type the following at a command prompt:

```
C:\tmp> nfsadmin server  <parameter for NIS server or LDAP server information>

C:\tmp> nfsadmin mapping <parameters>
```

For a detailed list of configuration options for **nfsadmin**, type **nfsadmin server /?**.

An example of configuring identity mapping using **nfsadmin** follows:

```
c:\tmp> nfsadmin mapping

The following are the settings on localhost

Mapping Server Lookup       : Disabled

Mapping Server              :

AD Lookup                   : Enabled

AD Domain                   : Contoso.com
```

Using the information you gathered earlier for group identifiers, type the following in Windows PowerShell to configure groups:

```
Set-NfsgroupStore

Set-NfsClientGroup  (or) New-NfsClientGroup

Set-NfsNetGroup (or) New-NfsNetGroup
```

After you have configured the netgroup and client group, you can set the NFS share permissions using the Grant-NfsSharePermission Windows PowerShell cmdlet . Some examples of configuring share permissions follow:

```
PS C:\> New-NfsClientgroup -ClientGroupName MIGRATION -AddMember 'MACHINE1','MACHINE2'

PS C:\> Get-NfsClientgroup MIGRATION

ClientGroupName                                          ClientGroupMembers

---------------                                          ------------------

MIGRATION                                                {MACHINE1, MACHINE2}

PS C:\> Grant-NfsSharePermission -Name NfsTestShare1 -ClientName MIGRATION -ClientType
clientgroup -Permission readonly

PS C:\> Get-NfsSharePermission NfsTestShare1

Name                  ClientName             Permission        AllowRootAccess

----                  ----------             ----------      ---------------

NFSTestShare1          MIGRATION                READ              False
```

If you are using Unmapped UNIX User Access (UUUA), see NFS Identity Mapping in Windows Server 2012, which provides information about the various methods of identity mapping. You should note that both Windows Server 2012 R2 and Windows Server 2012 support UNIX and Linux-based password and group files.

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Post-Migration Tasks

This topic explains how to complete the migration if it was successful, and how to roll back or troubleshoot the migration if it failed.

Completing the migration

After you verify the migration, you can retire the source server.

Retire File and Storage Services on the source server

After you complete and verify the migration, the source server can be shut down or disconnected from the network.

Remove DFS Namespaces from the source server

The procedure you use to remove DFS Namespaces from the source server depends on whether the namespaces are stand-alone or domain-based. If you want to remove the namespace from the source server, you must use **DFSUtil.exe**.

Note
By default, clients cache the list of namespace servers for 300 seconds (five minutes), so we recommend that you do not run the **DFSUtil.exe remove** command within five minutes of completing verification of the DFS namespace migration. During migration, clients have only the temporary server in the cache of namespace servers. Waiting five minutes after you add the destination server to the namespace allows clients to list the destination server in their cache.

To remove stand-alone namespaces

1. Open a Command Prompt window on the destination server.
2. Type the following code, and then press Enter.

```
Dfsutil.exe root remove <\\SourceServer\Namespace>
```

To remove domain-based namespaces with one namespace server

1. On the destination server, open a Command Prompt window.
2. Type the following, and then press **Enter**.

```
DFSUtil.exe target remove <\\TemporaryServer\Namespace>
```

Note

This procedure applies only if a temporary server was added to the namespace for migration purposes. For domain-based namespaces with more than one namespace server, no additional actions are required.

Restoring File and Storage Services in the event of migration failure

The following sections describe how to restore the File and Storage Services server role in the event of migration failure.

Roll back DFS Namespaces

The steps that you perform to roll back DFS Namespaces depend on whether the namespaces are stand-alone or domain-based, and whether a temporary namespace was created during the migration process.

To roll back DFS Namespaces (do one of the following)

1. For stand-alone namespaces, no action is required other than migrating the identity back to the source server.
2. For domain-based namespaces with more than one namespace server, or if a temporary server was added to a namespace that initially had only one namespace server, do the following:
   a. Remove the destination server from the namespace.
   b. Migrate identity and shared folder information to the source server.
   c. Add the source server to the namespace.
3. For domain-based namespaces with only one namespace server, where no temporary namespace server was added during migration, do the following:
   a. Migrate identity and shared folder information to the source server.
   b. Verify that the export file for the namespace that was created during migration is still available.
   c. Delete the namespace.
   d. Create the namespace on the source server.
   e. Import the namespace configuration from the export file created during the migration.

f. Manually reset delegation permissions to the namespace.

Note

Another option to migrate domain-based namespaces with one namespace server is to temporarily add a second namespace server before the migration, and then remove the temporary server after the migration.

Roll back data and shared folders

If no changes have been made to migrated files, folders, and shared folders on the destination server and this data has not been deleted from the source server, no additional steps to roll back data and shared folders are required.

If the migrated files, folders, or shared folders may have been modified on the destination server by the administrators or users, perform the following steps to synchronize the changes from the destination server back to the source server:

1. Type the following command in a Command Prompt window to copy the updated migrated data (files and folders) from the destination server back to the source server:

   ```
   robocopy <copy from path> <copy to path> /E
   ```

   This command can be executed on the source server or on the destination server, and it will recursively copy updated data. Type `robocopy /?` in a Command Prompt window for additional copy options, including options to copy file and folder permissions.

   Caution

   Permissions that you set for nondefault local users and groups will not copy properly and need to be created manually.

2. Compare the lists of shared folders and their permissions on the source server and destination server and manually synchronize any changes.

   To list all shared folders and their permissions, type the following command in a Windows PowerShell session that has been opened with elevated user rights:

   ```
   gwmi win32_share | %{net share $_.name}
   ```

Roll back migration on the other computers in the enterprise

If the migration failed, verify that the other computers in the enterprise can access the source server after you roll back the migration data.

Troubleshooting migration issues

Troubleshooting tips include the following:

- For physical migration issues:

  When some files are migrated physically and others are copied, there is a chance that the File Server Resource Manager configuration is not synchronized. To remedy this, delete and create new copies of the Quota.md and Datascrn.md files.

- For domain-joined computers:

  If a custom action (quota notification or file management task) fails to execute with an access-denied failure and a corresponding event log, you should remove the custom action and create it on the destination server.

Troubleshoot data migration that does not complete

If the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets run indefinitely without completing, your destination server might not have sufficient disk space or a large enough File Server Resource Manager or NTFS quota limit to allow for data migration to finish. To determine whether insufficient disk space is preventing the data send-receive process from completing, do the following on the destination server.

1. Open %*localappdata*%/Svrmig/Log/SetupAct.log.
2. Review the most recent log entries. If the following exception occurs, your destination server has insufficient disk space or File Server Resource Manager or NTFS quota limits to complete data migration.

   ```
   Win32Exception: unable to write to FileStream: There is not enough space on the disk.
   ```

 **To resolve this issue, do the following**

1. Press Ctrl+C to cancel **Send-SmigServerData** and **Receive-SmigServerData** on both source and destination servers.
2. Check for sufficient disk space on the destination server's hard disk drive. If the destination server's hard disk drive has insufficient space, do one of the following:
   - Clear additional space.
   - Identify a different hard disk drive that has sufficient space.
3. If the destination server's hard disk drive, the destination path, or any folders that contain the destination path have a File Server Resource Manager or NTFS quota enabled, and the quota limit does not allow for sufficient disk space to migrate data, do one of the following:
   - Increase the quota limit to set sufficient disk space to migrate the data. For more information about FSRM quota management, see one of the following.
     - Quota Management for Windows Server 2012, Windows Server 2008 R2, and Windows Server 2008
     - Quota Management for Windows Server 2003 R2

     For more information about NTFS quota management, see one of the following.
     - Setting Disk Quotas for Windows Server 2012, Windows Server 2008 R2, and Windows Server 2008
     - Enable disk quotas for Windows Server 2003 R2 and Windows Server 2003

- Identify a different hard disk drive that already has sufficient space and File Server Resource Manager or NTFS quota limits.

4. Run the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets again, specifying a destination path that has sufficient disk space, and large enough File Server Resource Manager or NTFS quota limits, if applicable.

Troubleshoot data migration connectivity

If the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets cannot establish connectivity, verify the following conditions and then try again:

1. In the **Send-SmigServerData** command on the source server, the *ComputerName* parameter correctly specifies the name of the destination server.
2. The **Receive-SmigServerData** and **Send-SmigServerData** commands are entered on the destination server and the source server respectively within five minutes of one another. This is the default maximum connection timeout for **Send-SmigServerData** and **Receive-SmigServerData**. You can change the maximum connection timeout for the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets by modifying the following user-defined registry subkey on the source server and destination server.

   Subkey: **\HKEY_Local_Machine\Software\Microsoft\ServerMigration**

   Value: MaxConnectionTime (REG_DWORD)

   Data: Between 1 and 3600 (represents the connection time-out in seconds). If a value larger than 3600 is specified, 3600 seconds is used as the maximum connection time-out.

   For information about how to create a Windows Registry key, see Add a Registry Key.
3. The same password is entered on the source server and destination server.
4. The source server and destination server are available on the same subnet:
   a. On the destination server, in a Command Prompt window, type `ipconfig` and note the subnet mask value.
   b. On the source server, in a Command Prompt window, type `ipconfig` and note the subnet mask value.
   c. Ensure that the subnet mask values are the same on the source server and destination server.
5. Port 7000 is open on the source and destination server and is not in use by another application.
   a. To check if port 7000 is open, in a Command Prompt window, enter the command:

      ```
      netsh firewall show portopening
      ```

      If port 7000 is not in the list, follow the instructions in File and Storage Services: Appendix A: Optional Procedures to open port 7000.
   b. If port 7000 is open, type the following command to check if port 7000 is being used by another application:

      ```
      netstat
      ```

      - In the Local Address column, you will see *<IP Address>*: *<port number>*.

- If port 7000 is in the list, it is being used by another application.

Troubleshoot unexpected Windows PowerShell session closure

If a migration cmdlet fails and the Windows PowerShell session closes unexpectedly with an access violation error message, look for a message similar to the following example in the %*localappdata*%\SvrMig\Logs\setuperr.log file.

```
FatalError [0x090001] PANTHR Exception (code 0xC0000005: ACCESS_VIOLATION) occurred at 0x000007FEEDE9E050 in C:\Windows\system32\migwiz\unbcl.dll (+000000000008E050). Minidump attached (317793 bytes).
```

This failure occurs when the server cannot contact domain controllers that are associated with domain users or groups who are members of local groups, or who have rights to files or shares that are being migrated. When this happens, each domain user or group is displayed in the GUI as an unresolved security identifier (SID). An example of a SID is **S-1-5-21-1579938362-1064596589-3161144252-1006**.

To prevent this problem, verify that required domain controllers or global catalog servers are running, and that network connectivity allows communication between both source and destination servers and required domain controllers or global catalog servers. Then, run the cmdlets again.

If connections between either the source or destination servers and the domain controllers or global catalog servers cannot be restored, do the following

1. Before you run **Export-SmigServerSetting**, **Import-SmigServerSetting**, or **Get-SmigServerFeature** again, remove all unresolved domain users or groups who are members of local groups from the server on which you are running the cmdlet.
2. Before you run **Send-SmigServerData** or **Receive-SmigServerData** again, remove all unresolved domain users or groups who have user rights to files, folders, or shares on the migration source server.

Locate the deployment log file

The Windows Server Migration Tools deployment log file is located at %windir%\Logs\SmigDeploy.log. Additional Windows Server Migration Tools log files are created at the following locations:

- `%windir%\Logs\ServerMigration.log`
- On Windows Server 2012, Windows Server 2008 R2, and Windows Server 2008: `%localappdata%\SvrMig\Log`
- On Windows Server 2003: `%userprofile%\Local Settings\Application Data\SvrMig\Log`

If migration log files are not created in the preceding locations, ServerMigration.log and SmigDeploy.log are created in `%temp%`, and other logs are created in `%windir%\System32`.

View the content of Windows Server Migration Tools result objects

All Windows Server Migration Tools cmdlets return results as objects. You can save result objects and query them for more information about the settings and data that were migrated. You can also use result objects as input for other Windows PowerShell commands and scripts.

Result object descriptions

The **Import-SmigServerSetting** and **Export-SmigServerSetting** cmdlets in Windows Server Migration Tools return results in a list of **MigrationResult** objects. Each **MigrationResult** object contains information about the data or setting that the cmdlet processes, the result of the operation, and any related error or warning messages. The following table describes the properties of a **MigrationResult** object.

Property Name	Type	Definition
ItemType	Enum	The type of item being migrated. Values include **General**, **WindowsFeatureInstallation**, **WindowsFeature**, and **OSSetting**.
ID	String	The ID of the migrated item. Examples of values include **Local User**, **Local Group**, and **DHCP**.
Success	Boolean	The value **True** is displayed if the migration was successful; otherwise, **False** is displayed.
DetailsList	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.

Send-SmigServerData and **Receive-SmigServerData** cmdlets return results in a list of **MigrationDataResult** objects. Each **MigrationDataResult** object contains information about the data or shared folder that the cmdlet processes, the result of the operation, any error or warning messages, and other related information. The following table describes the properties of a **MigrationDataResult** object.

Property Name	Type	Definition
ItemType	Enum	The type of migrated item. Values include **File**, **Folder**, **Share**, and **Encrypted File**.

Property Name	Type	Definition
SourceLocation	String	The source location of the item, shown as a path name.
DestinationLocation	String	The destination location of the item shown as a path name.
Success	Boolean	The value **True** is displayed if the migration was successful; otherwise, **False** is displayed.
Size	Integer	The item size, in bytes.
ErrorDetails	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.
Error	Enum	Errors enumeration for errors that occurred.
WarningMessageList	List <String>	A list of warning messages.

The following table describes the properties of objects within the **MigrationResultDetails** object that are common to **MigrationResult** and **MigrationDataResult** objects.

Property name	Type	Definition
FeatureId	String	The name of the migration setting that is related to the item. Examples of values include **IPConfig** and **DNS**. This property is empty for data migration.
Messages	List <String>	A list of detailed event messages.
DetailCode	Integer	The error or warning code associated with each event message.
Severity	Enum	The severity of an event, if events occurred. Examples of values include **Information**, **Error**, and **Warning**.

Property name	Type	Definition
Title	String	Title of the result object. Examples of values include the physical address of the network adapter for IP configuration, or the user name for local user migration.

Examples

The following examples show how to store the list of the result objects in a variable, and then use the variable in a query to return the content of result objects after the migration is complete.

To store a list of result objects as a variable for queries

1. To run a cmdlet and save the result in a variable, type a command in the following format, and then press **Enter**.

   $*VariableName* **=** $(*Cmdlet*)

   The following is an example.

   $ImportResult = $(Import-SmigServerSetting -FeatureId DHCP -User all -Group -Path D:\rmt\DemoStore -force -Verbose)

   This command runs the **Import-SmigServerSetting** cmdlet with several parameters specified, and then saves result objects in the variable **ImportResult**.

2. After the **Import-SmigServerSetting** cmdlet has completed its operations, return the information contained in the result object by typing a command in the following format, and then pressing **Enter**.

   $*VariableName*

   In the following example, the variable is named **ImportResult**.

   $*ImportResult*

   This command returns information contained in the result objects that were returned by **Import-SmigServerSetting** in the example shown in step 1. The following is an example of the output that is displayed by calling the **ImportResult** variable:

```
            ItemType  ID                                  Success
DetailsList
            --------  --                                  -------
-----------
          OSSetting  Local User                            True
{Local User, Loc...
          OSSetting  Local Group                           True
```

```
{Local Group, Lo...
     WindowsFeature  DHCP                                    True
{}
```

Each line of the preceding example is a migration result for an item that was migrated by using the **Import-SmigServerSetting** cmdlet. The column heading names are properties of **MigrationResult** objects. You can incorporate these properties into another command to return greater detail about result objects, as shown by the examples that follow in steps 3 and 4.

3. To display a specific property for all result objects in the list, type a command in the following format, and then press **Enter**.

   $<VariableName>| Select-Object -ExpandProperty <PropertyName>

   The following is an example.

   $importResult | Select-Object -ExpandProperty DetailsList

4. You can run more advanced queries to analyze result objects by using Windows PowerShell cmdlets. The following are examples:

   - The following command returns only those details of result objects that have the ID **Local User**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList

   - The following command returns only those details of result objects with an ID of **Local User** that have a message severity equal to **Warning**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Severity -eq "Warning") {$_} }

   - The following command returns only the details of result objects with an ID of **Local User** that also have the title **Remote Desktop Users**.

     $ImportResult | Where-Object { $_.ID -eq "Local Group" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Title -eq "Remote DesktopUsers") {$_} }

More information about querying results

For more information about the cmdlets that are used in the preceding examples, see the following additional resources:

- Using the Where-Object Cmdlet
- Using the Select-Object Cmdlet
- Using the ForEach-Object Cmdlet

For more information about Windows PowerShell scripting techniques, see What Can I Do With Windows PowerShell?

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Migrate an iSCSI Software Target

File and Storage Services: Migrate Network File System

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix A: Optional Procedures

Opening ports in Windows Firewall

The following instructions are for opening ports in Windows Firewall. If you have a non-Microsoft firewall installed, consult the guide for that firewall about how to open ports. Opening ports in Windows Firewall can be done through the command line.

Important
Opening ports in your firewall can leave your server exposed to malicious attacks. Make sure that you understand firewall systems before you open ports.

To open Windows Firewall ports by using the command line (do one of the following):

1. Open a Command Prompt window with elevated user rights, type the following, and then press Enter.
   - On computers that are running Windows Server 2003, type:

     ```
     netsh firewall add allowedprogram
     program=%windir%\System32\WindowsPowerShell\v1.0\powershell.ex
     e name="ServerMigration" mode=ENABLE
     ```
   - On computers that are running Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2, or Windows Server 2008, type the following commands in order, and press Enter after each command.

     i.

     ii.
2. If you have changed the default behavior of Windows Firewall to block all outbound traffic on computers that are running Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2, or Windows Server 2008, you must explicitly allow outbound

traffic on UDP port 7000. To do this, open a Command Prompt window with elevated user rights, type the following, and then press Enter.

```
netsh advfirewall firewall add rule name=ServerMigration(UDP-Out) dir=out program=%windir%\System32\WindowsPowerShell\v1.0\powershell.exe action=allow protocol=UDP localport=7000
```

Closing ports in Windows Firewall

As a best practice, we recommend that you close Windows Firewall ports after the data transfer operation is completed.

To close Windows Firewall ports by using the command line

- Do one of the following:
  - On computers that are running Windows Server 2003, open a Command Prompt window, type the following, and then press Enter.

    ```
    netsh firewall delete allowedprogram program=%windir%\System32\WindowsPowerShell\v1.0\powershell.exe
    ```
  - On computers that are running Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2, or Windows Server 2008, open a Command Prompt window with elevated user rights, type the following two commands, and press Enter after each command.

    ```
    netsh advfirewall firewall delete rule name=ServerMigration(TCP-In)
    ```

    ```
    netsh advfirewall firewall delete rule name=ServerMigration(UDP-Out)
    ```

Detect reparse points and hard links

The following commands can be used to detect reparse points and mounted volumes in any folder and its subfolders. Open a Command Prompt window, type the following commands to detect reparse points, in which **D:\Test** represents the hard disk drive and folder that you want to search, and then press Enter.

```
dir D:\Test\* /S /A:L
```

The option **/A:L** specifies that only reparse points need to be enumerated. The output is similar to the following:

```
Volume in drive D has no label.

Volume Serial Number is 3AE4-E412
```

```
Directory of D:\Test\Links

10/07/2008  03:44 PM    <JUNCTION>     JunctionMSIT [d:\test\targets\msit]
10/07/2008  03:42 PM    <SYMLINK>      LinkMSIT [d:\test\targets\msit]
10/07/2008  03:41 PM    <SYMLINKD>     SymLinkMSIT [d:\test\targets\msit]
1 File(s)              0 bytes

Directory of D:\Test\Targets

10/07/2008  05:35 PM    <JUNCTION>     Volume [\??\Volume{0674413f-760d-11dd-beb3-
806e6f6e6963}\]
0 File(s)              0 bytes

Total Files Listed:
1 File(s)              0 bytes
3 Dir(s)  17,918,840,832 bytes free
```

To enumerate hard links on a file on Windows Server 2012 R2, Windows Server 2012, or Windows Server 2008 R2, open a Command Prompt window with elevated user rights, type the following command, and then press Enter.

```
fsutil hardlink list D:\Test\File.txt
```

To enumerate hard links on all files in a folder on Windows Server 2012 R2, Windows Server 2012, or Windows Server 2008 R2, run the following command in a Windows PowerShell session that has been opened with elevated user rights:

```
Get-ChildItem D:\* | %{'Links for: ' + $_.FullName; fsutil hardlink list $_.FullName; ""}
```

For more information about enumerating hard links on computers that are running Windows Server 2008 or Windows Server 2003, see FindFirstFileNameW function on MSDN.

Migrated and nonmigrated attributes for local users and groups

For more information about the attributes of local users and groups that can be migrated, see the Local User and Group Migration Guide.

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Prepare to Migrate

File and Storage Services: Appendix B: Migration Data Collection Worksheets

SMB data collection worksheet

Use this server message block (SMB) data collection worksheet to record data for SMB policies that are set on the source server.

#	Source Server Essential Settings	Setting Identification
01	Idle time The setting name is: **Microsoft network server: Amount of idle time required before suspending a session**.	Idle time (in minutes): _____ Group or Local Policy: _____
02	S4USelf The setting name is: **Microsoft network server: Attempt S4USelf to obtain claim information**.	Claim information: __ Default __ Enabled or __ Disabled Group or Local Policy: _____
03	Sign (always) The setting name is: **Microsoft network server: Digitally sign communications (always)**.	Sign always: __ Enabled or __ Disabled Group or Local Policy: _____
04	Sign (if client agrees) The setting name is: **Microsoft network server: Digitally sign communications (if client agrees)**.	Sign if client agrees: __ Enabled or __ Disabled Group or Local Policy: _____
05	Disconnect when logon hours	Disconnect: __ Enabled or __

#	Source Server Essential Settings	Setting Identification
	expire The setting name is: **Microsoft network server: Disconnect clients when logon hours expire**.	Disabled Group or Local Policy: _____

BranchCache data collection worksheet

Use this BranchCache data collection worksheet to record data for the BranchCache policies that are set on the source server.

#	Source Server Essential Settings	Setting Identification
01	BranchCache The setting name is: **Hash Publication for BranchCache**.	BranchCache: __ Not configured, __ Enabled, or __ Disabled Group or Local Policy: _____
	BranchCache The setting name is: **Hash Version support for BranchCache**.	BranchCache: __ Not configured, __ Enabled, or __ Disabled Group or Local Policy: _____

See Also

Migrate File and Storage Services to Windows Server 2012 R2

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Migrate an iSCSI Software Target

File and Storage Services: Migrate Network File System

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

Migrate Remote Desktop Services to Windows Server 2012 R2

Remote Desktop Services is a role in the Windows Server operating system that provides multi-user access to applications and desktops for non-administrative purposes. This guide describes how to migrate Remote Desktop Services, what Remote Desktop Services role services will be migrated, and tasks that apply to migrating the role services.

About this guide

This guide describes how to migrate the Remote Desktop Services role by providing preparation, migration and verification steps.

Migration documentation and tools ease the migration of server role settings and data from an existing server to a destination server that is running Windows Server 2012 R2. By using the process described in this guide, you can simplify the migration process, reduce migration time, increase the accuracy of the migration process, and help eliminate possible conflicts that might otherwise occur during the migration process.

Note

Your detailed feedback is very important, and it helps us make Windows Server Migration Guides as reliable, complete, and easy-to-use as possible. Please take a moment to rate this topic and add comments that support your rating. Click **Rate this topic** at the top of the page. Describe what you liked, did not like, or want to see in future versions of the topic. To submit additional suggestions about how to improve Windows Server migration guides or tools, please write a post on the Windows Server Migration forum.

Target audience

This guide is intended for the following audiences:

- IT architects who are responsible for computer management and security throughout an organization
- IT operations engineers who are responsible for the day-to-day management and troubleshooting of networks, servers, client computers, operating systems, or applications
- IT operations managers who are accountable for network and server management

What this guide does not provide

This guide does not cover migration of the following:

- Customizations made to any Remote Desktop Services role service. In particular, this may apply to the RD Session Host, RD Virtualization Host, RD Web Access, or RD Connection Broker role services.
- Third-party application settings, programs, or plug-ins

- More than one server role at the same time
- More than one role service at a time
- Group Policy settings
- User profiles, including roaming profiles
- Event history
- Microsoft applications or application settings
- RD Connection Broker servers that are configured in a clustered or load-balanced environment (except High-Availability mode)

This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, it is recommended that you design a custom migration procedure that is specific to your server environment, based on the information provided in other role migration guides. Migration guides for additional server roles are available at Migrate Roles and Features to Windows Server.

Caution

If your source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other roles that are running on the source server to fail.

Supported migration scenarios

This guide provides you with instructions for the following:

- Migrating a server that is running Remote Desktop Services on Windows Server 2012 to a server that is running Remote Desktop Services on Windows Server 2012 R2
- Migrating between two servers running Remote Desktop Services on Windows Server 2012 R2

For the migration scenarios that are described in this guide, each of the Remote Desktop Services role services is migrated separately. You can migrate one, some, or all role services by following the steps in this guide. For information about the order of migration, see Order of migration for multiple role services later in this topic.

Supported operating systems

The Remote Desktop Services role services are available in Windows Server 2012 R2 as follows:

- All Remote Desktop Services role services are available in Windows Server 2012 R2 Standard, Windows Server 2012 R2 Enterprise, and Windows Server 2012 R2 Datacenter.
- RD Web Access is available in Windows Web Server 2012 R2.
- RD Session Host, RD Licensing, RD Web Access, and RD Gateway are available in Windows Server 2012 R2 Foundation.

Physical to virtual machine migration

Migration between physical operating systems and virtual operating systems are supported for the RD Connection Broker, RD Session Host, RD Web Access, RD Licensing, and RD Gateway

role services. However, the RD Virtualization Host role services and the Hyper-V role do not run on virtual machines.

Backward compatibility

You can migrate the Remote Desktop Services role services from computers running Windows Server 2012 or Windows Server 2012 R2 to a computer running Windows Server 2012 R2.

Policy and configuration settings

Some Remote Desktop Services settings can be configured by using Group Policy. Information about migrating Group Policy settings is not included in this migration guide.

Supported role services and features

This migration guide describes how to migrate the Remote Desktop Services role services from a source server running Windows Server 2012 or Windows Server 2012 R2 to a destination server running Windows Server 2012 R2.

Following are the Remote Desktop Services role services that can be migrated to a computer running Windows Server 2012 R2:

1. RD Connection Broker
2. RD Virtualization Host
3. RD Session Host
4. RD Web Access
5. RD Licensing
6. RD Gateway

Migration scenarios that are not supported

The following scenarios are not supported:

- Upgrading Windows Server 2008 Terminal Services or Windows Server 2012 R2 Remote Desktop Services server role or role services
- Migrating or upgrading from Windows Server 2003 or Windows Server 2003 R2
- Migrating from a source server to a destination server that is running an operating system with a different system UI language installed
- Migrating the RD Virtualization Host or RD Session Host role services from physical computers to virtual machines
- Migrating any applications or application settings from the RD Session Host server

Order of migration for multiple role services

The steps in this guide are based on migrating the role services in the following order when you are migrating more than one role service:

1. RD Connection Broker

2. RD Session Host
3. RD Virtualization Host
4. RD Web Access

The following role services can be migrated at any time during the migration:

- RD Licensing
- RD Gateway

In a Remote Desktop Services deployment, RD Connection Broker servers must be migrated first. All other services can be migrated independently. If you do not have RD Connection Broker servers, you can migrate other role services by following the steps provided in this document.

The Remote Desktop license server can be migrated at any time, but if the new license server does not have the same name as the source server, the Remote Desktop deployments and standalone RD Session Host servers that use that license server must be configured after migration to use the new license server.

The RD Gateway server migration is not dependent on the other role services for migration. It can be migrated at any time.

Impact of migration on Remote Desktop Services

A Remote Desktop Services role service will not be available during migration. This is also the case for any role services that are dependent on it. In addition, applications and add-ons on the affected servers will not be available.

Migration times will be affected by the dependencies between role services. For example, RD Session Host servers, RD Virtualization Host servers, and RD Web Access servers are dependent on RD Connection Broker servers. These dependencies should be considered when you are estimating downtime.

Plan your data migration to occur during off-peak hours to minimize downtime and reduce impact to users. Notify users that the resources will be unavailable during that time.

In some deployments, replication may extend the length of time that the services are unavailable.

If there is more than one role service on the source server, after you remove the source server from the domain, you will not have access to role services that you didn't migrate.

The following table details the expected impacts during the migration process.

Role service	Dependent role services	Impact of migration	Downtime estimates
RD Connection Broker	RD Virtualization Host, RD Session Host, RD Web Access	Users will not have access to any resources that are managed by the RD Connection Broker or TS Session Broker	Three hours

Role service	Dependent role services	Impact of migration	Downtime estimates
		server that is being migrated. These resources include RemoteApp programs, virtual desktop pools, and personal virtual desktops.	
RD Session Host	RD Web Access may be dependent on RD Session Host in your deployment.	Session collections will not be available until migration of all destination servers in the virtual desktop collection is complete. RemoteApp programs will not be available until they are installed on the destination servers.	One hour
RD Virtualization Host	RD Virtualization Host is dependent on RD Connection Broker.	Virtual desktop collections will not be available until migration of all destination servers in the virtual desktop collection is complete.	Three hours or more depending on the number of virtual machines being migrated
RD Web Access	RD Web Access cannot serve connections to session collections or virtual desktop collections while they are being migrated.	Resources that are accessed by RD Web Access and managed by the associated RD Connection Broker server will not be available. These resources include session collections and virtual desktop collections.	One hour
RD Licensing	Remote Desktop deployments and standalone RD	Remote Desktop deployments and standalone RD	One hour

Role service	Dependent role services	Impact of migration	Downtime estimates
	Session Host servers must be configured with at least one Remote Desktop license server that is available to serve licenses. If not, users cannot connect to the RD Session Host servers while they are being migrated.	Session Host servers that are configured to use the license server may not be able to receive licenses during the migration.	
RD Gateway	RD Gateway	Users cannot access the network with the RD Gateway server that is being migrated. The Remote Desktop Gateway service may be slow or not available.	One hour

Additional references

- **You are here in this migration process document:** Migrate Remote Desktop Services to Windows Server 2012
- Remote Desktop Services: Prepare to Migrate
- Remote Desktop Services: Migrate Remote Desktop Services Role Services
- Remote Desktop Services: Verify the Migration
- Remote Desktop Services: Post-Migration Tasks

- Windows Server Migration forum
- Windows Server Migration Portal

Remote Desktop Services: Prepare to Migrate

This topic explains how to prepare to migrate the Remote Desktop Services role services from a source server running Windows Server 2012 or Windows Server 2012 R2 to a destination server running Windows Server 2012 R2. It assumes that you are migrating some or all of the role services, including dependencies, from a functional deployment of Remote Desktop Services.

The general preparation instructions provided in this topic apply to the following role services in Remote Desktop Services.

- RD Connection Broker
- RD Session Host
- RD Virtualization Host
- RD Web Access
- RD Licensing
- RD Gateway

Assign permissions required to migrate Remote Desktop Services

At a minimum, you must be a member of the **Administrators** group on the source server and the destination server to install, remove, or set up Remote Desktop Services.

Migration dependencies

Remote Desktop Services role services have dependencies or prerequisites for migration, as described in this section.

Prerequisite features to migrate separately

The following features in Remote Desktop Services must be migrated separately:

- DNS Server
- Active Directory Domain Services

  Remote Desktop User Profiles are stored in Active Directory.:

  - To migrate Active Directory Domain Services, see Active Directory Domain Services and DNS Server Migration Guide
  - To deploy user profiles, see User Profiles on Windows Server 2008 R2 Remote Desktop Services
- Active Directory Certificates Services

- If you are migrating an enterprise certification authority (CA) within the same domain, before you migrate Remote Desktop Services, follow the instructions in AD CS Migration: Migrating the Certification Authority.
- If you are migrating an enterprise CA within the same domain, before you migrate Remote Desktop Services, follow the instructions in AD CS Migration: Migrating the Certification Authority.
- Group Policy

  You can migrate Group Policy objects (GPOs) by using the Import Settings Wizard in the Group Policy Management Console (GPMC). For more information, see Import Settings from a Group Policy Object.
-

Prerequisite features already installed

Remote Desktop Services role services require the following roles and features in Windows Server 2012 R2. With the exception of Network Policy and Access Services (NPAS), these roles and features are installed automatically when the role service is installed, if they are not already installed on the server.

- RD Web Access requires Web Server (IIS)
- RD Virtualization Host requires Hyper-V
- RD Gateway requires Web Server (IIS), RPC over HTTP Proxy, and Network Policy and Access Services (NPAS)

Prepare your source server

To prepare your source server for migration, you need to back it up and gather data.

Back up your source server

Migrating some Remote Desktop Services role services require import or export of registry settings. You should back up the computer before working with the registry.

You can find information about backing up Windows Server 2012 and Windows Server 2012 R2 in the following topics:

- Backup and Recovery.
- Registry Editor.

Gather data from your source server

Settings for applications on the Remote Desktop Session Host server will not be gathered or recorded during this migration. Before you retire the RD Session Host server, gather and record the data that you will migrate from the source server into a data collection worksheet for each role service.

Prepare your destination server

The following steps are necessary to prepare all destination servers for the migration of Remote Desktop Services role services.

Hardware requirements for the destination server

Verify that the computer meets the hardware requirements for the role service and its prerequisites. Minimally, you should migrate to servers with comparable memory, disk space, processors, and GPUs.

The RD Virtualization Host server must meet the hardware requirements for the Hyper-V server role. For more information about Hyper-V hardware requirements, see Hardware Considerations.

RD Session Host, RD Web Access, and RD Virtualization Host cannot run on virtual machines.

Software requirements for the destination server

Remote Desktop Services is a server role in Windows Server 2012 R2. Windows Server 2012 R2 must be installed on the destination server before you migrate any of the Remote Desktop Services role services.

RD Session Host, RD Virtualization Host, RD Connection Broker, and RD Web Access require that the name of the destination server is the same as the name of the source server.

Other servers and client computers in the enterprise

Within the domain, if the destination server has the same name as the source server, no preparations are needed on other computers in the deployment.

To migrate Remote Desktop Services role services across domains, RD Session Host, RD Virtualization Host, RD Connection Broker, and RD Web Access must have accounts with permissions to join the new domain.

When you migrate RD Gateway and Remote Desktop license servers across domains, domain trust relationships are required.

Additional references

- Migrate Remote Desktop Services to Windows Server 2012 R2
- **You are here in this migration process document ->** Remote Desktop Services: Prepare to Migrate
- Remote Desktop Services: Migrate Remote Desktop Services Role Services
- Remote Desktop Services: Verify the Migration
- Remote Desktop Services: Post-Migration Tasks

- Windows Server Migration forum
- Windows Server Migration Portal

Remote Desktop Services: Migrate Remote Desktop Services Role Services

Migration for a Remote Desktop Services deployment is supported from source servers running Windows Server 2012 or Windows Server 2012 R2 to destination servers running Windows Server 2012 R2. Migration from any other major or minor releases to Windows Server 2012 R2 is not supported.

Following are the steps for migrating a Remote Desktop Services deployment:

1. Migrate the RD Connection Broker server
2. Migrate session collections
3. Migrate virtual desktop collections
4. Migrate RD Web Access servers
5. Migrate RD Gateway servers
6. Migrate RD Licensing servers
7. Migrate standalone Remote Desktop Services servers
8. Migrate certificates

Migrate the RD Connection Broker server

This is the first and most important step for migrating to a destination server running Windows Server 2012 R2.

- The Remote Desktop Connection Broker (RD Connection Broker) destination server must be configured for high availability to support migration.

  For more information, see RD Connection Broker High Availability in Windows Server 2012.
- If you have more than one RD Connection Broker server in the high availability setup, remove all the RD Connection Broker servers except the one that is currently active.
- Upgrade the remaining RD Connection Broker server to Windows Server 2012 R2.
- After the server is upgraded, add it to the high availability deployment.

Notes

A mixed high availability configuration with Windows Server 2012 and Windows Server 2012 R2 is not supported for RD Connection Broker servers.

An RD Connection Broker running Windows Server 2012 R2 can serve session collections with RD Session Host servers running Windows Server 2012, and it can serve virtual desktop collections with RD Virtualization Host servers running Windows Server 2012.

Migrate session collections

Follow these steps to migrate a session collection in Windows Server 2012 to a session collection in Windows Server 2012 R2.

Important
Migrate session collections only after successfully completing the previous step, Migrate the RD Connection Broker server.

1. Upgrade the session collection from Windows Server 2012 to Windows Server 2012 R2.
2. Add the new RD Session Host server running Windows Server 2012 R2 to the session collection.

   Tip
   Use drain mode when you are setting the RD Session Host servers. For more information about drain mode, see Introducing Terminal Services Server Drain Mode.
3.
4. Sign out of all sessions in the RD Session Host servers, and remove the servers that require migration from the session collection.

Notes
If the UVHD template (UVHD-template.vhdx) is enabled in the session collection and the file server has been migrated to a new server, update the **User Profile Disks: Location** collection property with the new path. The User Profile Disks must be available at the same relative path in the new location as they were on the source server.

A session collection of RD Session Host servers with a mix of servers running Windows Server 2012 and Windows Server 2012 R2 is not supported.

Migrate virtual desktop collections

Follow these steps to migrate a virtual desktop collection from a source server running Windows Server 2012 to a destination server running Windows Server 2012 R2.

Important
Migrate virtual desktop collections only after successfully completing the previous step, Migrate the RD Connection Broker server.

1. Upgrade the virtual desktop collection from the server running Windows Server 2012 to Windows Server 2012 R2.
2. Add the new Windows Server 2012 R2 RD Virtualization Host servers to the session collection.

   Tip
   Use drain mode when you set the RD Session Host servers that need to be migrated.
3.

4. Migrate all virtual machines in the current virtual desktop collection that are running on RD Virtualization Host servers to the new servers.
5. Remove all RD Virtualization Host servers that required migration from the virtual desktop collection in the source server.

Notes

If the UVHD template (UVHD-template.vhdx) is enabled in the session collection and the file server has been migrated to a new server, update the **User Profile Disks: Location** collection property with the new path. The User Profile Disks must be available at the same relative path in the new location as they were on the source server.

A session collection of RD Session Host servers with a mix of servers running Windows Server 2012 and Windows Server 2012 R2 is not supported.

Migrate RD Web Access servers

To migrate the RD Web Access servers, see Remote Desktop Web Access Role Service Migration.

Migrate RD Gateway servers

To migrate the RD Gateway Servers, see Remote Desktop Gateway Role Service Migration.

Migrate RD Licensing servers

Follow these steps to migrate an RD Licensing server from a source server running Windows Server 2012 or Windows Server 2012 R2 to a destination server running Windows Server 2012 R2.

1. Migrate the Remote Desktop Services client access licenses (RDS CALs) from the source server to the destination server.

   For more information, see Migrate Remote Desktop Services Client Access Licenses (RDS CALs).
2. Use the Deployment Properties Wizard to list the new RD Licensing servers on the server running Windows Server 2012 R2.
3. Remove the RDS CALs from the source RD Licensing server.

   For more information, see Remove Remote Desktop Services Client Access Licenses.
4. Remove the source RD Licensing servers from the deployment.

Migrate standalone Remote Desktop Services servers

The following list contains the complete migration guides for each role service. Each guide include information about preparing to migrate, verifying the migration, and post-migration tasks:

- Remote Desktop Session Host Role Service Migration
- Remote Desktop Virtualization Host Role Service Migration
- Remote Desktop Web Access Role Service Migration
- Remote Desktop Licensing Role Service Migration
- Remote Desktop Gateway Role Service Migration

Migrate certificates

Migrating certificates simply requires updating certificate information in **Deployment Properties: Manage certificates**

Remote Desktop Services features that use certificates

Although this guide does not describe how to migrate Remote Desktop Services features, the following list of features that use certificates is included for reference. Each of the following features uses certificates for at least one role service:

- Single sign-on (SSO) for RemoteApp and Desktop Connection
- Web Single Sign-On (Web SSO)
- HTTPS connections to RD Web Access
- Digital signing of Remote Desktop Protocol (.rdp) files for personal virtual desktops and virtual desktop pools
- Digital signing of Remote Desktop Protocol files for Remote App programs
- RD Gateway connections to Remote Desktop Services
- RD Session Host server connections in a farm configuration

Preparing certificates for migration

In most cases, the migration of certificates for Remote Desktop Services requires you to export the certificate with the private key. After export, you should store the certificate in a safe location.

A certificate with a private key can be migrated by using the following steps:

1. To export the certificate to a PFX file, see Export a certificate with the private key.
2. To import the certificate from a PFX file, see Import a certificate.

After you have imported the certificate to the certificate store on the destination server, follow the instructions for configuring the certificate for the specific role service.

Additional references

- Migrate Remote Desktop Services to Windows Server 2012 R2
- Remote Desktop Services: Prepare to Migrate
- **You are here in this migration process document ->**Remote Desktop Services: Migrate Remote Desktop Services Role Services
- Remote Desktop Services: Verify the Migration

- Remote Desktop Services: Post-Migration Tasks

- Windows Server Migration forum
- Windows Server Migration Portal

Remote Desktop Services: Verify the Migration

Verifying the destination server configuration is best done by running a pilot program. Use an Administrator account and an account for a valid user.

Run a pilot program

We recommend that you create a pilot program in the production environment to ensure that the migration of all role services was successful. Run the program on the servers before you put the migrated role services into production to verify that your deployment works as you expect. Depending on the role service that you migrated, you should limit connections at first, and slowly increase the number of users or connections.

Additional references

- Migrate Remote Desktop Services to Windows Server 2012 R2
- Remote Desktop Services: Prepare to Migrate
- Remote Desktop Services: Migrate Remote Desktop Services Role Services
- **You are here in this migration process document ->**Remote Desktop Services: Verify the Migration
- Remote Desktop Services: Post-Migration Tasks

- Windows Server Migration forum
- Windows Server Migration Portal

Remote Desktop Services: Post-Migration Tasks

This topic contains information about general post-migration tasks that you can perform after you migrate Remote Desktop Services role services from a source server running Windows Server 2012 or Windows Server 2012 R2 to a destination server running Windows Server 2012 R2.

The post-migration tasks include:

1. Retire the source servers

Retire the source servers

In each case, the source server is retired by removing it from the domain. After you complete and verify the migration, the source server can be shut down or disconnected from the network.

Caution
If there is more than one role service on the server, after removing the source server from the domain, you will not have access to the other role services on the computer.

- Migrate Remote Desktop Services to Windows Server 2012 R2
- Remote Desktop Services: Prepare to Migrate
- Remote Desktop Services: Migrate Remote Desktop Services Role Services
- Remote Desktop Services: Verify the Migration
- **You are here in this migration process document ->**Remote Desktop Services: Post-migration Tasks

- Windows Server Migration forum
- Windows Server Migration Portal

Migrate Cluster Roles to Windows Server 2012 R2

This guide provides step-by-step instructions for migrating clustered services and applications to a failover cluster running Windows Server 2012 R2 by using the Copy Cluster Roles Wizard. Not all clustered services and applications can be migrated using this method. This guide describes supported migration paths and provides instructions for migrating between two multi-node clusters or performing an in-place migration with only two servers. Instructions for migrating a highly available virtual machine to a new failover cluster, and for updating mount points after a clustered service migration, also are provided.

Operating system requirements for clustered roles and feature migrations

The Copy Cluster Roles Wizard supports migration to a cluster running Windows Server 2012 R2 from a cluster running any of the following operating systems:

- Windows Server 2008 R2 with Service Pack 1 (SP1)
- Windows Server 2012
- Windows Server 2012 R2

Migrations are supported between different editions of the operating system (for example, from Windows Server Enterprise to Windows Server Datacenter), between x86 and x64 processor architectures, and from a cluster running Windows Server Core or the Microsoft Hyper-V Server R2 operating system to a cluster running a full version of Windows Server.

The following migrations scenarios are not supported:

- Migrations from Windows Server 2003, Windows Server 2003 R2, or Windows Server 2008 to Windows Server 2012 R2 are not supported. You should first upgrade to Windows Server 2008 R2 SP1 or Windows Server 2012, and then migrate the resources to Windows Server 2012 R2 using the steps in this guide. For information about migrating to a Windows Server 2012 failover cluster, see Migrating Clustered Services and Applications to Windows Server 2012. For information about migrating to a Windows Server 2008 R2 failover cluster, see Migrating Clustered Services and Applications to Windows Server 2008 R2 Step-by-Step Guide.
- The Copy Cluster Roles Wizard does not support migrations from a Windows Server 2012 R2 failover cluster to a cluster with an earlier version of Windows Server.

Important

Before you perform a migration, you should install the latest updates for the operating systems on both the old failover cluster and the new failover cluster.

Target audience

This migration guide is designed for cluster administrators who want to migrate their existing clustered roles, on a failover cluster running an earlier version of Windows Server, to a Windows Server 2012 R2 failover cluster. The focus of the guide is the steps required to successfully migrate the clustered roles and resources from one cluster to another by using the Copy Cluster Roles Wizard in Failover Cluster Manager.

General knowledge of how to create a failover cluster, configure storage and networking, and deploy and manage the clustered roles and features is assumed.

It is also assumed that customers who will use the Copy Cluster Roles Wizard to migrate highly available virtual machines have a basic knowledge of how to create, configure, and manage highly available Hyper-V virtual machines.

What this guide does not provide

This guide does not provide instructions for migrating clustered roles by methods other than using the Copy Cluster Roles Wizard.

This guide identifies clustered roles that require special handling before and after a wizard-based migration, but it does not provide detailed instructions for migrating any specific role or feature. To find out requirements and dependencies for migrating a specific Windows Server role or feature, see Migrate Roles and Features to Windows Server 2012 R2.

This guide does not provide detailed instructions for migrating a highly available virtual machine (HAVM) by using the Copy Cluster Roles Wizard. For a full discussion of migration options and requirements for migrating HAVMs to a Windows Server 2012 R2 failover cluster, and step-by-step instructions for performing a migration by using the Copy Cluster Roles Wizard, see Hyper-V: Hyper-V Cluster Migration.

Planning considerations for migrations between failover clusters

As you plan a migration to a failover cluster running Windows Server 2012 R2, consider the following:

- For your cluster to be supported by Microsoft, the cluster configuration must pass cluster validation. All hardware used by the cluster should be Windows logo certified. If any of your hardware does not appear in the Windows Server Catalog in hardware certified for Windows Server 2012 R2, contact your hardware vendor to find out their certification timeline.

  In addition, the complete configuration (servers, network, and storage) must pass all tests in the Validate a Configuration Wizard, which is included in the Failover Cluster Manager snap-in. For more information, see Validate Hardware for a Failover Cluster.
- Hardware requirements are especially important if you plan to continue to use the same servers or storage for the new cluster that the old cluster used. When you plan the migration, you should check with your hardware vendor to ensure that the existing storage meets certification requirements for use with Windows Server 2012 R2. For more information about hardware requirements, see Failover Clustering Hardware Requirements and Storage Options.
- The Copy Cluster Roles Wizard assumes that the migrated role or feature will use the same storage that it used on the old cluster. If you plan to migrate to new storage, you must copy or move of data or folders (including shared folder settings) manually. The wizard also does not copy any mount point information used in the old cluster. For information about handling mount points during a migration, see Cluster Migrations Involving New Storage: Mount Points.
- Not all clustered services and features can be migrated to a Windows Server 2012 R2 failover cluster by using the Copy Cluster Roles Wizard. To find out which clustered services and applications can be migrated by using the Copy Cluster Roles Wizard, and operating system requirements for the source failover cluster, see Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012 R2.

Migration scenarios that use the Copy Cluster Roles Wizard

When you use the Copy Cluster Roles Wizard for your migration, you can choose from a variety of methods to perform the overall migration. This guide provides step-by-step instructions for the following two methods:

- **Create a separate failover cluster running Windows Server 2012 and then migrate to that cluster**. In this scenario, you migrate from a multi-node cluster running Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2. For more information, see Migrate Between Two Multi-Node Clusters: Migration to Windows Server 2012 R2.
- **Perform an in-place migration involving only two servers**. In this scenario, you start with a two-node cluster that is running Windows Server 2008 R2 SP1 or Windows Server 2012, remove a server from the cluster, and perform a clean installation (not an upgrade) of Windows Server 2012 R2 on that server. You use that server to create a new one-node failover cluster running Windows Server 2012 R2. Then you migrate the clustered services and applications from the old cluster node to the new cluster. Finally, you evict the remaining node from the old cluster, perform a clean installation of Windows Server 2012 R2 and add the Failover Clustering feature to that server, and then add the server to the new failover cluster. For more information, see In-Place Migration for a Two-Node Cluster: Migration to Windows Server 2012 R2.

Note

We recommend that you test your migration in a test lab environment before you migrate a clustered service or application in your production environment. To perform a successful migration, you need to understand the requirements and dependencies of the service or application and the supporting roles and features in Windows Server in addition to the processes that this migration guide describes.

In this guide

Related references

What's New in Failover Clustering in Windows Server 2012 R2

Failover Clustering Overview

Failover Clustering Hardware Requirements and Storage Options

Create a Failover Cluster

Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012 R2

This topic provides guidance for migrating specific cluster roles to a failover cluster running the Windows Server 2012 R2 operating system by using the Copy Cluster Roles Wizard in Failover Cluster Manager. The topic covers supported migration paths, provides an overview of wizard-based migration, and notes which cluster roles require special handling during migration.

Migration paths for specific migrations

The following table lists the operating system versions on a source failover cluster that can be migrated to a failover cluster running Windows Server 2012 R2 for each clustered service or application. Migrations between failover clusters created with physical computers and failover clusters that are created from virtual machines (also known as *guest clusters*) are supported.

Supported migrations for clustered roles and resources to a Windows Server 2012 R2 failover cluster

Clustered role or resource	From Windows Server 2008 R2 SP1	From Windows Server 2012	From Windows Server 2012 R2
Cluster Registry settings	Yes	Yes	Yes
Cluster Shared Volume (CSV) volumes	Yes	Yes	Yes
DFS Namespace (DFS-N)	Yes	Yes	Yes
DFS Replication (DFS-R)	Yes	Yes	Yes
DHCP Server	Yes	Yes	Yes
Distributed Network Name (DNN)	No	Yes	Yes
File Server	Yes	Yes	Yes
Scale-Out File Server for application data	No	Yes	Yes
Generic Application	Yes	Yes	Yes
Generic Script	Yes	Yes	Yes
Generic Service	Yes	Yes	Yes

Clustered role or resource	From Windows Server 2008 R2 SP1	From Windows Server 2012	From Windows Server 2012 R2
Virtual Machine	Yes	Yes	Yes
Hyper-V Replica Broker	No	Yes	Yes
IP addresses (IPV4, IPV6, IPv6 tunnel addresses)	Yes	Yes	Yes
iSCSI Target Server	Yes	Yes	Yes
Internet Storage Name Service (iSNS)	Yes	Yes	Yes
Message Queuing (MSMQ), MSMQ triggers	Yes	Yes	Yes
Network Name resources	Yes	Yes	Yes
NFS shares	Yes	Yes	Yes
Other Server	Yes	Yes	Yes
Physical Disk resource	Yes	Yes	Yes
WINS Server	Yes	Yes	Yes

Note

In Windows Server 2012 R2, you can designate a virtual hard disk (.vhdx file) as shared storage for multiple virtual machines that are configured as a guest failover cluster. This new type of guest cluster, known as a *shared VHDX guest cluster*, enables scenarios such as Microsoft SQL Server Failover Cluster Instance (FCI) guest clusters. The Copy Cluster Roles Wizard supports migration of the roles in the table above (except for the Virtual Machines role, which cannot exist in a guest cluster) between shared-VHDX guest clusters running the released version of Windows Server 2012 R2. However, if you created a shared-VHDX guest cluster in Windows Server 2012 R2 Preview, you cannot use the wizard to copy the cluster roles to a shared VDX guest cluster running the released version of Windows Server 2012 R2.

Cluster roles that cannot be migrated

Some services and applications that can run in a failover cluster on Windows Server 2012 R2 cannot be migrated by using the Copy Cluster Roles Wizard—in some cases because they were not supported on earlier versions of clustering. The Copy Cluster Roles Wizard in Windows Server 2012 R2 cannot be used to migrate the following clustered roles:

- Microsoft SQL Server - For upgrade guidance for SQL Server, see the whitepaper SQL Server 2012 Upgrade Technical Guide.
- Microsoft Exchange Server - For upgrade guidance for Exchange Server, see Understanding Upgrade to Exchange 2010.
- Print Spooler from Windows Server 2008 R2 – In Windows Server 2012 R2 and Windows Server 2012, the print spooler is no longer a clustered resource. Instead, high availability is defined as a highly available virtual machine running on a single cluster node. The Print Server role is installed on a single virtual machine, which can be migrated to other nodes automatically or manually. For more information, see High Availability Printing Overview.
- Remote Desktop Connection Broker from Windows Server 2008 R2 - In Windows Server 2012 R2 and Windows Server 2012, the active/passive clustering model for the RD Connection Broker role service, used in earlier versions of Windows Server, is replaced by the Active/Active Broker feature, which eliminates the need for clustering and provides a fully active/active model. For more information, see the blog entry RD Connection Broker High Availability in Windows Server 2012.
- Volume Shadow Copy Service tasks
- Task Scheduler tasks (Windows Server 2012 R2 and Windows Server 2012 only)
- Cluster Aware Updating (CAU) settings (Windows Server 2012 R2 and Windows Server 2012 only)

Roles restricted to a single instance per cluster

For the following roles, only one instance per failover cluster is supported:

- DHCP Server
- WINS Server
- iSCSI Target Server
- Hyper-V Replica Broker (Windows Server 2012 R2 and Windows Server 2012 only)

For those roles, the Copy Cluster Roles Wizard will not attempt to create a second role instance if one instance already exists on the target cluster.

Migrations for which the Copy Cluster Roles Wizard performs most or all steps

For the following clustered services or applications, The Copy Cluster Roles Wizard performs most or all steps for a migration to a Windows Server 2012 R2 failover cluster:

- Distributed File System (DFS) Namespace

- Generic Application
- Generic Script
- Generic Service
- IPv4 Address, when migrating within the same subnet
- IPv6 Address or IPv6 Tunnel Address
- Internet Storage Name Service (iSNS)
- Network Name (other than the cluster name)

  If Kerberos authentication is enabled for the Network Name resource, the migration wizard prompts you for the password for the Cluster service account that is used by the old cluster.
- NFS
- Physical Disk (resource settings only; does not copy data to new storage)
- Windows Internet Name Service (WINS) (Extra steps might be required if you migrate to new storage, and you use a different drive letter on the path to the new database.)

For more information about the Copy Cluster Roles Wizard, see Create a Failover Cluster. For step-by-step instructions for performing a migration between two multimode failover clusters, see Migrate Between Two Multi-Node Clusters: Migration to Windows Server 2012 R2. For step-by-step instructions for performing a stand-alone migration while upgrading a single failover cluster, see In-Place Migration for a Two-Node Cluster: Migration to Windows Server 2012 R2.

Migration within mixed environments

The Copy Cluster Roles Wizard can migrate clustered resources within mixed environments. For example, the wizard accommodates the following differences in the source and destination environments:

- Migrate static IP addresses to a cluster using DHCP.
- Migrate IPv4 resources into an IPv6 environment.
- Migrate across routed subnets.
- Migrate a physical cluster to a guest (virtual) cluster (with the exception of Hyper-V clusters, which must run on physical computers).
- Migrate between different editions of the operating system (for example, from Windows Server Enterprise to Windows Server Datacenter), between x86 and x64 processor architectures, and from a cluster running Windows Server Core or Microsoft Hyper-V Server to a cluster running a full version of Windows Server.

During migration, the wizard allows you to address name conflicts between resource groups, resources, and share names and to address drive letter collisions. The wizard resolves the conflicts as part of the post-migration repair process.

Important

The Copy Cluster Roles Wizard moves resources, not data. If you plan to migrate to new storage, you must move the data and folders yourself.

Additional steps for a wizard-based migration

Some additional steps typically are needed before or after you run the wizard, including the following:

- Install server roles and features that are needed in the new cluster. In most cases, you must install the role or feature on all nodes of the cluster. For example, before you migrate a highly available virtual machine, you must install the Hyper-V server role on each cluster node.
- Copy or install any associated applications, services, or scripts on the new cluster (all nodes).
- If a migrated role or feature uses the same storage, take the services and storage offline on the old cluster and then make the storage available to the new cluster.
- If a migrated role or feature uses new storage, ensure that any data and folders are copied to new storage. Verify permissions on any shared subfolders that were migrated.
- If the new cluster is on a different subnet, provide static IP addresses.
- If the new cluster uses a different volume letter, update drive path locations for applications.
- Configure Task Manager tasks on the new cluster. (Windows Server 2012 R2 or Windows Server 2012 only)
- For a virtual machine, install the latest integration services on the virtual machine. Configure Volume Shadow Copy Service (VSS) backups. For a migration from Windows Server 2012 R2 or Windows Server 2012, configure Hyper-V Replica settings.
- Configure Cluster Aware Updating (CAU). (Windows Server 2012 R2 and Windows Server 2012 only)

Failover Cluster Copy Roles reports

The wizard provides a Failover Cluster Copy Roles Pre-Copy Report (formerly the Pre-Migration Report) and a Failover Cluster Copy Roles Post-Copy Report (formerly the Post-Migration Report), which provide important information. We recommend that you review both reports while performing a migration:

- The Pre-Copy Roles Report explains whether each resource that you plan to migrate is eligible for migration.
- The Post-Copy Roles Report contains information about the success of the migration, and describes additional steps that might be needed before you bring the migrated resources online.

Note

Two resource groups are never migrated: **Cluster Core Resources Group** and **Available Storage Group**. You can ignore these resource groups in the Failover Cluster Copy Roles reports.

Clustered role and feature migrations that require extra steps

This section provides guidance for migrating clustered roles and features that require additional steps before or after you run the Copy Cluster Roles Wizard to perform a migration between clusters.

- Clustered DFS Replication migrations
- Clustered DHCP migrations
- Clustered DTC migrations
- Clustered File Server and Scale-out File Server migrations
- Clustered FSRM migrations
- Clustered Message Queuing (MSMQ) migrations
- Other Server migrations involving resource types not built into failover clusters
- Migration of highly available virtual machines

Clustered DFS Replication migrations

Before you migrate clustered Distributed File System (DFS) Replication (also known as DFS-R or DFSR) to a cluster running Windows Server 2012 R2, you must add the new cluster to the DFS replication group to which the old cluster belongs, and then wait until DFS Replication synchronizes the data to the new cluster. After data synchronization is complete, you can decommission the old cluster. For step-by-step guidance, see Migrate File and Storage Services to Windows Server 2012 R2 and File and Storage Services: Post-Migration Tasks.

To migrate clustered instances of DFS Replication to a cluster running Windows Server 2012 R2

1. Obtain the name of the cluster to which you will migrate. In Active Directory, this is the name that is used for the computer account of the cluster itself (also called the cluster name object or CNO). Add this name to the replication group that you will migrate. For more information, see Add a member to a replication group.
2. Wait until DFS Replication finishes synchronizing the replicated data to the cluster to which you will migrate.
3. If you plan to decommission the cluster from which you migrated, remove its network name from the replication group. If necessary, destroy the cluster.

For more information about DFS Replication in Windows Server 2012 R2, see DFS Namespaces and DFS Replication Overview. For step-by-step instructions for migrating DEF Replication, see Migrate File and Storage Services to Windows Server 2012 R2.

Clustered DHCP migrations

When migrating clustered Dynamic Host Configuration Protocol (DHCP) to a cluster running Windows Server 2012 R2, the Copy Cluster Roles Wizard migrates resources and settings, but

not the DHCP database. For information about how to migrate the DHCP database, see DHCP Server Migration: Migrating the DHCP Server Role. The information in the topic also applies to migrations from Windows Server 2008 R2 or Windows Server 2012 to Windows Server 2012 R2. The topic includes information about migrating from a cluster.

Note

Although the migration of the clustered DHCP role is supported, in Windows Server 2012 R2 there is the option to use DHCP failover. DHCP failover provides redundancy and load balancing without clustered DHCP. For more information, see Migrate to DHCP Failover and Understand and Deploy DHCP Failover.

Clustered DTC migrations

Before you begin the migration of clustered Distributed Transaction Coordinator (DTC) to a cluster running Windows Server 2012 R2, you must make sure the list of transactions stored by DTC is empty. This is referred to as *draining the transaction logs*. If you do not drain the logs, the information in the logs (the transaction state information for unresolved transactions) will be lost during the migration. Unresolved transactions include **Active**, **In Doubt**, and **Cannot Notify** transactions.

To drain DTC transaction logs of unresolved transactions

1. Stop the application that creates transactions on the clustered instance of DTC that is being migrated.
2. On a node of the cluster that you are migrating from, click **Start**, point to **Administrative Tools**, and then click **Component Services**. (In Windows Server 2012 R2, open **Component Services** directly from the Start screen.)
3. Expand **Component Services**, expand **Computers**, expand **My Computer**, expand **Distributed Transaction Coordinator**, and then expand **Clustered DTCs**.
4. Expand the clustered instance of DTC that you are migrating, and then click **Transaction List**.
5. View the transaction list to see if it is empty. If there are transactions listed, then either wait for them to be completed or right-click each transaction, click **Resolve**, and then select **Forget**, **Commit**, or **Abort**.

   For information about the effect of each of these options, see Transaction State Resolution After System Failure.

For additional information, see View Transaction Information.

Clustered File Server and Scale-out File Server migrations

Several methods are available for migrating a scale-out file server or traditional clustered file server to Windows Server 2012 R2. For all methods, there are trade-offs among required downtime, migration duration, resource usage, and required hardware. The best method for your

environment depends on hardware and resources you have available, the volume of data to be moved, the number of clustered file servers that are affected, and service requirements.

Choosing the best migration method for your file server

When you plan your clustered file server migration, consider these methods:

- Virtual machine storage migration
- Copy Cluster Roles Wizard - Migrate to a new multi-node cluster
- Copy Cluster Roles Wizard – In-place migration
- Migrate storage pools

Note
For a fuller discussion of storage upgrade options as an integral part of upgrading your private cloud infrastructure, view the presentation Upgrading Your Private Cloud with Windows Server 2012 R2, presented at TechEd 2013.

Virtual machine storage migration

Introduced in Windows Server 2012, virtual machine storage migration enables you to the virtual hard disks used by one clustered file server to another clustered file server while the virtual machine remains running. This is known as *storage migration*. After you migrate storage for each virtual machine, you migrate the virtual machines to the new Windows Server 2012 R2 failover cluster. For more information, see Virtual Machine Storage Migration Overview.

This method is useful for moving to new storage if you have the resources available to maintain required service levels on all of the virtual machines during migration.

Migration method: Virtual machine storage migration

Advantages	Disadvantages
Live-migrate storage without any downtime for the virtual machines.	The process moves lots of data over the network, using lots of resources. If you are migrating a large number of virtual machines, and don't have the network capacity to gracefully handle the large loads, this can have a large impact on performance. You must move to new storage.

Copy Cluster Roles Wizard - Migrate to a new multi-node cluster

With this method, you set up a new Windows Server 2012 R2 failover cluster, migrate the File Server role to the new cluster, and then take the file server offline while you redirect storage to the new cluster. The wizard does not move data; if you migrate to new storage, the wizard updates the storage settings for the role, but you must move the data and files manually during the migration. For step-by-step instructions, see Migrate Between Two Multi-Node Clusters: Migration to Windows Server 2012 R2.

Use this method if you have too much data to move over the network without unacceptable impact on the performance of your clustered file servers.

Migration method: Copy Cluster Roles Wizard – Migrate to a new multi-node cluster

Advantages	Disadvantages
This method is much faster than storage migration. In a large enterprise with hundreds of clustered file servers, the migration can take hours rather than days.	Downtime is required. You must take the File Server roles offline on the old cluster while you redirect the storage to the new cluster, However, this method is faster than moving VHDs over the network, and you can schedule the downtime for a maintenance window, when you will experience a limited interruption in service but will not risk degrading service for running virtual machines over long periods. Additional hardware is required to create the new failover cluster.

Copy Cluster Roles Wizard – In-place migration

If you do not have the hardware available to create a new multi-node Windows Server 2012 R2 failover cluster before you migrate the cluster roles, you can perform an in-place migration. In an in-place migration, you use hardware from an existing cluster to create the new cluster, evicting one node to use as the first node in the new cluster.

For a two-node cluster, you would evict one node, perform a clean installation of Windows Server 2012 R2 on that node, create a new single-node failover cluster with that node, and then migrate the File Server role from the old cluster to the new cluster. At that point, you must take the File Server roles offline on the old cluster while you redirect storage to the new cluster. When the migration is complete, you then destroy the old cluster, install Windows Server 2012 R2 on the other cluster node, and add that node to the new cluster. For step-by-step instructions, see In-Place Migration for a Two-Node Cluster: Migration to Windows Server 2012 R2.

Migration method: Copy Cluster Roles Wizard – In-place migration

Advantages	Disadvantages
No new hardware required. Data is not migrated over the network.	Downtime is required: you must take the File Server roles offline on the old cluster before you can redirect storage to the new cluster and then bring the roles and the storage online on the new cluster. While migrating a two-node cluster in place, you take on the added risk of losing high availability for your file servers from the time

Advantages	Disadvantages
	when you remove the first node from the old cluster until you add the second node to the new cluster. Service can be degraded on the nodes that remain online during the migration, particularly if you are migrating large numbers of clustered file servers.

Storage pool migration

If you are migrating from a Windows Server 2012 failover cluster that uses storage pools, you can minimize the impact of migration by migrating one storage pool at a time, from the old cluster to the new cluster. With storage pools, instead of managing each disk individually, you add physical disks to one or more pools and then create virtual disks from the available capacity. You then create volumes on the virtual disks, as if they were physical disks. When you run low on the available capacity in the pool, add physical disks to the pool to create bigger pools with more capacity for more virtual disks.

Storage Spaces uses commodity drives that are attached via Serial-Attached SCSI (SAS), Serial ATA (SATA), or USB. When it is time to change from the old cluster using the storage to the new cluster using the storage, you might need to change the cabling. If you are reusing hardware (that is, you are performing an in-place migration), when you evict a node from the old cluster, you need to disconnect that server's connection to the disks. When it is time to change the storage from the old cluster to the new cluster, disconnect the storage from the old cluster before you connect the storage to the new cluster, so that only one cluster is connected to the disks at one time. When you connect the storage to the new cluster, the Storage Spaces and associated storage pools becomes available to the new cluster so that the migration can complete.

For more information about using Storage Spaces and storage pools, see Storage Spaces Overview, What's New in Storage Spaces in Windows Server 2012 R2, and Deploy Clustered Storage Spaces. For a video presentation from TechEd 2013 that demonstrates Storage Spaces basics and new features, see Storage Spaces: What's New in Windows Server 2012 R2.

Migration method: Storage pool migration

Advantages	Disadvantages
No downtime is required. High availability is maintained throughout migration.	Data moves over the network. A four-node cluster is required to enable you to maintain two nodes on both the old and new clusters during migration.

Additional tasks for file server migration using the Copy Cluster Roles Wizard

If you choose to use the Copy Cluster Roles Wizard to migrate your file server, be aware of the following requirements:

- If you plan to migrate to new storage, keep in mind that if the migrated files and folder inherit permissions from their parents, during migration it is the inheritance setting that is migrated, not the inherited permissions. Therefore it is important to make sure that the parent folders on the source server and the destination server have the same permissions to maintain the permissions on migrated data that has inherited permissions. After the file server migration, it's important to verify the folder permissions after the migration. Sometimes folder permissions reset to Read-only during a file server migration.
- You do not need to migrate the quorum resource. When you run the Create a Cluster Wizard in Windows Server 2012 R2 or win8_server_2 , the cluster software automatically chooses the quorum configuration that provides the highest availability for your new failover cluster, and it dynamically updates the quorum configuration if you add or evict nodes. You can change the quorum configuration on the new cluster if necessary for your specific environment. However, Dynamic Quorum is not in effect on a Windows Server 2008 R2 failover cluster. If you evict a node to perform an in-place migration, you will need to update the quorum configuration.

Clustered FSRM migrations

To migrate the File Server Resource Manager (FSRM) classification, storage reporting, and file management task configuration on a clustered file server running Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2 to a failover cluster running Windows Server 2012 R2, you must export the configuration from one FSRM server node in the cluster and then import the configuration to another FSRM server. These steps must be performed locally on one node of the cluster. You then fail over the other nodes until this process is complete. For step-by-step instructions, see Migrate File and Storage Services to Windows Server 2012 R2.

Important
When you migrate the configuration, FSRM requires that you use the same drive letters on both the source and destination servers.

Clustered Message Queuing (MSMQ) migrations

When you migrate a clustered instance of Message Queuing (also known as MSMQ) to a cluster running Windows Server 2012 R2, it's important to take the following precautions to ensure that the data is preserved and you can bring the service online on the new cluster:

- Before you migrate, you should back up the data that is associated with clustered instances of Message Queuing. This ensures that you can restore service-specific Message Queuing data if it is accidentally deleted during migration. For more information about Message Queuing backup and restore, see Backing up and restoring messages.
- During the migration, it's important to make sure that the migration is complete before you delete either clustered instance of Message Queuing (old or new). Otherwise, service-specific

data for Message Queuing might be deleted from the shared storage, which prevents the remaining Message Queuing resource from coming online. After the migration is complete and you are ready to delete a clustered instance of Message Queuing (old or new), first remove the disk resource from that clustered instance and take the disk offline. Then delete the clustered instance of Message Queuing.

Other Server migrations involving resource types not built into failover clusters

Before you use the Copy Cluster Roles Wizard to migrate an application that uses a clustered resource type that is not built into failover clustering, be sure to add the resource type to the new cluster. You can then use the Copy Cluster Roles Wizard to migrate your clustered application. In this situation, the Copy Cluster Roles Wizard attempts a "best effort" migration.

To add a resource type to a failover cluster running Windows Server 2012 R2

1. Open Failover Cluster Manager from the Start screen of any node in the cluster running Windows Server 2012 R2.
2. If the cluster to which you want to migrate is not displayed, in the console tree, right-click **Failover Cluster Manager**, click **Connect to Cluster**, select the cluster that you want to migrate to, and then click **OK**.
3. In the console tree, right-click the cluster, and then click **Properties**.
4. Click the **Resource Types** tab, and then click **Add**.
5. Specify the following information for the resource type:
   - **Resource DLL path and file name**: The path and file name of the resource dynamic-link library (DLL) that the Cluster service should use when it communicates with your service or application.
   - **Resource type name**: The name that the Cluster service uses for the resource type. This name stays the same regardless of the regional and language options that are currently selected.
   - **Resource type display name**: The name that is displayed for the resource type. This name might vary when you make changes to regional and language options.

Migration of highly available virtual machines

You can use the Copy Cluster Roles Wizard to migrate highly available virtual machines created in Hyper-V from a Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2 failover cluster to a cluster running Windows Server 2012 R2. Using the wizard, you migrate the Virtual Machine clustered role, select highly available virtual machines to migrate, and update virtual network settings for the virtual machines on the new cluster.

Migrating HAVMs by using the Copy Cluster Roles Wizard has the advantage of not copying VHDs over the network, so migration completes fairly quickly and downtime is limited. However, the wizard cannot migrate virtual machines to new storage. Also, if you migrate one virtual machine on a Cluster Shared Volume (CSV) volume, all virtual machines on that volume are

migrated. And downtime is required: after you copy the Virtual Machine roles to the cluster, you must take the virtual machines on the old cluster offline, mask the storage to the old cluster, unmask the storage to the new cluster, then bring the storage online on the new cluster, and then start the virtual machines on the new cluster.

Caution

It is very important that you not turn on the migrated virtual machines on the new cluster before you take the virtual machines offline on the old cluster. Running a virtual machine on both clusters at the same time might corrupt the virtual machine.

For step-by-step instructions for migrating highly available virtual machines from a Windows Server 2012 failover cluster to a Windows Server 2012 R2 failover cluster by using the Copy Cluster Roles Wizard, see Copy Cluster Roles Wizard in Hyper-V Cluster Using Separate Scale-Out File Server Migration, or, if your virtual machines are stored on Cluster Shared Volume (CSV) volumes, see Hyper-V Cluster Using Cluster Shared Volumes (CSV) Migration. You can use the same procedures to migrate virtual machines from CSV volumes on a Windows Server 2008 R2 cluster to a Windows Server 2012 R2 cluster.

Alternate methods for migrating HAVMs to a Windows Server 2012 R2 failover cluster

Depending upon your environment and the service requirements for the migrated workloads, you should consider two alternate methods for migrating highly available virtual machines:

- **Cross version live migration** – Windows Server 2012 R2 introduces a new method for migrating a highly available virtual machine from a Windows Server 2012 cluster to a Windows Server 2012 R2 cluster. Using cross version live migration, you can migrate vritual machines to the new failover cluster without any downtime. If the virtual hard disks (VHDs) are stored on a Scale-out File Server share that is accessible to both clusters, you don't have to copy files over the network. However, depending on factors such as the amount of memory configured for the virtual machine, migration can be slow, and resource consumption during the live migrations can be high.
- **Export/Import method** - You also can migrate individual virtual machines by using the **Export** and **Import** actions in Hyper-V Manager (also available in Windows PowerShell). The Export/Import method lets you migrate virtual machines one at a time and control the method by which they the VHDs are copied to the new cluster. The virtual machine must be taken offline during the export and import, and you must re-enable Hyper-V replication on the virtual machine after migration.

For a comparison of migration methods for migrating HAVMs to a Windows Server 2012 R2 failover cluster, see Hyper-V: Migration Options.

Note

You must use the Copy Cluster Roles Wizard or the **Export** and **Import** actions to migrate from a Windows Server 2008 R2 cluster to a Windows Server 2012 R2 cluster. Cross version live migration is only available when you migrate from Windows Server 2012.

Additional tasks for using the Copy Cluster Roles Wizard to migrate HAVMs

When you migrate HAVMs by using the Copy Cluster Roles Wizard, a few extra steps are required:

- You must merge or discard all shadow copies before you migrate the volumes that are attached to the virtual machines. Before you begin working with shadow copies, you should back up volumes.
- After you migrate the virtual machines to the new cluster, install the latest Hyper-V integration services on the new virtual machines.
- After you migrate, The Copy Cluster Roles Wizard does not migrate the following settings. You will need to configure the settings on the new cluster after migration.
  - Hyper-V Replica settings

    Important
    If you using Hyper-V Replica with the workload that you are migrating, see the "Hyper-V Replica" section of Hyper-V: Migration Options for special considerations when migrating from Windows Server 2012 to Windows Server 2012 R2.
  - Volume Shadow-Copy Service (VSS) tasks
  - Cluster-Aware Updating (CAU) settings

Additional references

- Migrate Cluster Roles to Windows Server 2012 R2
- Windows Server Migration Forum
- Failover cluster basics:
  - What's New in Failover Clustering in Windows Server 2012 R2
  - Failover Clustering Overview
- Instructions for migrations that use the Copy Cluster Role Wizard:
  - Migrate Between Two Multi-Node Clusters: Migration to Windows Server 2012 R2
  - In-Place Migration for a Two-Node Cluster: Migration to Windows Server 2012 R2
  - Hyper-V: Hyper-V Cluster Migration (Migrating highly available virtual machines from Windows Server 2012 or Windows Server 2012 R2)
  - Cluster Migrations Involving New Storage: Mount Points
- Migrating individual roles and features:
  - Migrating Roles and Features in Windows Server
  - Migrate File and Storage Services to Windows Server 2012 R2
- High availability for Microsoft Exchange Server 2013:
  - Deploying High Availability and Site Resilience
  - Understanding Upgrade to Exchange 2010
  - Upgrade from Exchange 2010 to Exchange 2013
  - Exchange Server Supportability Matrix

- High availability for Microsoft SQL Server 2012:
  - Microsoft SQL Server AlwaysOn Solutions Guide for High Availability and Disaster Recovery (whitepaper)
  - SQL Server 2012 AlwaysOn: Multisite Failover Cluster Instance (whitepaper)
  - SQL Server 2012 Upgrade Technical Guide (whitepaper)
  - Upgrade a SQL Server Failover Cluster
  - Upgrade a SQL Server Failover Cluster Instance (Deployment)
  - Supported Version and Edition Upgrades

Migrate Between Two Multi-Node Clusters: Migration to Windows Server 2012 R2

This topic provides step-by-step instructions for migrating cluster roles from a multi-node failover cluster running Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2 to a multimode cluster running Windows Server 2012 R2. (Alternatively, you can perform an in-place migration using a single two-node cluster. For more information, see In-Place Migration for a Two-Node Cluster: Migration to Windows Server 2012 R2.) If you plan to migrate highly available Hyper-V virtual machines (by migrating the Virtual Machine cluster role), see Hyper-V: Hyper-V Cluster Migration for step-by-step instructions.

Important
Before you begin your migration, review Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012 R2 to confirm that the clustered service or application can be migrated by using the Copy Cluster Roles Wizard.

Overview of migration of cluster roles between two multi-node failover clusters

The procedures in this topic describe the following process for migrating cluster roles from an existing multi-node cluster to a new multi-node Windows Server 2012 R2 failover cluster.

1. Cluster roles: Prepare to migrate between two multi-node clusters
   - To prepare servers for the new cluster – Install the Windows Server 2012 R2 operating system, required server roles and features, and any services or applications that will run on the new failover cluster. Pre-test services and applications to make sure they are compatible with Windows Server 2012 R2. Verify that your storage is certified for use with Windows Server 2012 R2.
   - To prepare storage for the new cluster – Storage preparation differs slightly depending on whether you will use the same storage for the new cluster that the old cluster is using or you plan to migrate to new storage. If you migrate highly available virtual machines, some additional storage preparation is required.

- To create the new failover cluster and configure your firewall

2. Cluster roles: Migrate the cluster roles – Use the Copy Cluster Roles Wizard in Failover Cluster Manager to migrate the cluster roles to the new cluster.
3. Cluster roles: Post-migration tasks for a migration between two multi-node clusters – Before you can bring a cluster role online on the new cluster, take the role offline on the new server, take the storage offline on the old cluster, and bring the storage on the new cluster online on the new cluster online. If you migrated highly available virtual machines, install the latest integration services on the virtual machines.
4. Cluster roles: Verify the migration:
   - To verify that the migrated cluster roles are performing as expected on the new cluster - Verify that the workload is available on the new cluster, and that service is provided at the required service-level agreement (SLA). For virtual machines, verify the status of the virtual machines in Hyper-V Manager, and confirm that you can connect to the virtual machines by using Remote Desktop or Virtual Machine Connection.
   - To verify that the migrated cluster roles can fail over successfully
   - To troubleshoot issues with failover for the migrated cluster roles

Impact of a migration between two multi-node clusters

When you migrate a cluster role between two multi-node failover clusters, you can prepare the destination failover cluster, configure storage on the new cluster, and copy the cluster role to the new cluster while maintaining service availability on the old cluster. However, customers will experience a brief downtime after the cluster role is migrated and before you bring the role online on the new cluster.

If the new cluster will use the same storage that the old cluster is using, before you bring the role online on the new cluster, you must take the clustered role offline on the old cluster, mask the storage to the old cluster, unmask the storage to the new cluster, and then bring the volumes or disks online on the new cluster.

If you are migrating to new storage, before you can bring the role online on the new cluster, you must take the role offline on the old cluster, copy data and files for the clustered role to the new storage (the Copy Cluster Roles Wizard does not move data), and then bring the new storage online on the new cluster.

Warning

If you plan to use the Copy Cluster Roles Wizard to migrate a highly available Hyper-V virtual machine from a Windows Server 2008 R2 failover cluster to a Windows Server 2012 R2 failover cluster, be aware that live migration is not supported for that scenario. However, you can live migrate an HAVM from a cluster running Windows Server 2012 or Windows Server 2012 R2 to a Windows Server 2012 R2 failover cluster.

Access rights required to complete migration

To migrate a cluster role by using the Copy Cluster Roles Wizard, you must be a local administrator on the destination failover cluster and on the cluster or cluster node from which you are migrating.

Additional references

In-Place Migration for a Two-Node Cluster: Migration to Windows Server 2012 R2

Hyper-V: Migration Options

Hyper-V: Hyper-V Cluster Migration

Cluster Migrations Involving New Storage: Mount Points

Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012 R2

Windows Server Migration Forum

Clustering Forum for Windows Server 2012

Cluster roles: Prepare to migrate between two multi-node clusters

To prepare to migrate cluster roles to a new failover cluster, perform the following tasks:

- To prepare servers for the new cluster
- To prepare storage for the new cluster
- To create the new failover cluster and configure your firewall

To prepare servers for the new cluster

1. Perform a clean installation of Windows Server 2012 R2 on each server that you will add to the new failover cluster.
2. Install the Failover Clustering feature on each server.
3. If you plan to migrate highly available virtual machines, add the Hyper-V role to each server.
4. Install any needed services, applications, and server roles. For example, if you plan to migrate clustered Windows Internet Name Service (WINS) to the new cluster, install the WINS Server feature by using Server Manager.
5. If you are migrating a Generic Application, Generic Script, or Generic Service resource, confirm that any associated application is compatible with Windows Server 2012 R2. You also must confirm that any associated service exists in Windows Server 2012 R2 and has the same name that it had in the old cluster. Test the application or service (separately, not as part of a cluster) to confirm that it runs as expected.

To prepare storage for the new cluster

1. If you plan to migrate to existing storage, verify that your existing storage is certified for

use with Windows Server 2012 R2.

2. Make an appropriate number of LUNs or disks accessible to the servers, and do not make those LUNs or disks accessible to any other servers. If the new cluster will use old storage, for testing purposes, you can limit the number of LUNs or disks to one or two. If the new cluster will use new storage, make as many disks or LUNs accessible to the new server as you think the cluster will need.

   Note
   We recommend that you keep a small disk or LUN available (unused by clustered services and applications) throughout the life of the cluster, so that you can always run storage validation tests without taking your services and applications offline.

3. Confirm that the intended cluster disks are visible and are formatted appropriately:
   a. On one of the servers that you plan to include in the cluster, open **Computer Management** from the Start screen, and then click **Disk Management** in the console tree.
   b. In Disk Management, confirm that the intended cluster disks are visible.
   c. Check the format of any exposed volume or LUN. We recommend that you use NTFS for the format. (For a disk witness, you must use NTFS.)
4. To prepare to migrate a highly available virtual machine, you must merge or discard all shadow copies that have been created for the virtual machine:
   a. Back up the volumes that store the virtual machines.
   b. Merge or discard shadow copies for each virtual hard disk (VHD).
   c. If you are migrating virtual machines stored on a Cluster Shared Volume (CSV) volume, make sure that you want to migrate all of the virtual machines on any volume that you plan to migrate. If you migrate one virtual machine that is stored on Cluster Shared Volume (CVS) volume, the Copy Cluster Roles Wizard migrates all virtual machines on that volume. This restriction does not apply when you migrate a Scale-out File Server cluster, which does not use CSV volumes.
5. If you are using new storage, and your disk configuration uses mount points, review Cluster Migrations Involving New Storage: Mount Points to identify any additional steps that you need to perform.

To create the new failover cluster and configure your firewall

1. Create the new failover cluster. For information about how to create a Windows Server 2012 R2 failover cluster, see Create a Failover Cluster.
2. After you create the cluster, ensure that your firewall is configured appropriately. For example, if you are using Windows Firewall, and you will be sharing folders and files, use your preferred Windows Firewall interface to allow the exception for **Remote Volume Management**.

Cluster roles: Migrate the cluster roles

Use the following instructions to migrate clustered services and applications from your old cluster to your new cluster. The Copy Cluster Roles Wizard leaves most of the migrated resources offline so that you can perform additional steps before you bring them online.

Note

To migrate a cluster role by using the Copy Cluster Roles Wizard, you must be a local administrator on the destination failover cluster and on the cluster or cluster node from which you are migrating.

Before you copy cluster roles to a new failover cluster

- If you plan to use new storage with the migrated clustered roles, before you run the Copy Cluster Roles Wizard, ensure that the storage is available to the new cluster – that is, ensure that the volumes have been added to the new cluster and that the volumes are online. This enables the wizard to update storage settings during migration.

To copy cluster roles from an existing cluster to a new cluster

1. From the **Start** screen or from **Server Manager** (**Tools**), open **Failover Cluster Manager**.
2. In the console tree, if the cluster that you created is not displayed, right-click **Failover Cluster Manager**, click **Connect to Cluster**, and then select the cluster that you want to configure.
3. In the console tree, expand the cluster that you created to see the items underneath it.
4. If the clustered servers are connected to a network that is not to be used for cluster communications (for example, a network intended only for iSCSI), then under **Networks**, right-click that network, click **Properties**, and then click **Do not allow cluster network communication on this network**. Click **OK**.
5. In the console tree, select the cluster.
6. Under **Configure**, click **Copy Cluster Roles**.

   The Copy Cluster Roles Wizard opens.
7. Read the Welcome page, and then click **Next**.
8. Specify the name or IP address of the cluster or cluster node from which you want to migrate services and applications, and then click **Next**.
9. The **Select Roles** page lists the clustered roles that can be migrated from the old cluster. The list does not contain any role that is not eligible for migration. Click **View Report** to view details in the Failover Cluster Pre-Copy Report. Then select each cluster role that you want to copy to the new cluster, and click **Next**.

   Important

   We recommend that you read the report, which explains whether each resource is eligible for migration. (The wizard also provides a report after it finishes, which describes any additional steps that might be needed before you bring the

migrated resource groups online.)

If storage is available on the new cluster, the **Specify Storage for Migration** page appears, giving you the option to migrate to new storage. If storage is not available on the new cluster, the wizard retains existing storage settings and does not display the page.

Note

Not all clustered roles can be migrated to new storage. For example, the wizard cannot be used to migrate highly available virtual machines (the Virtual Machine role) to new storage. For an overview of options for migrating highly available virtual machines to a Windows Server 2012 R2 failover cluster and step-by-step instructions for each migration option, see Hyper-V: Hyper-V Cluster Migration.

10. If you want to use new storage for a service or application:
    a. On the **Specify Storage for Migration** page, select the cluster disk that you want to migrate to new storage, and then click **Select Storage**.
    b. In the **Select Storage for Resource Group** dialog box, under **Available Storage in New Cluster**, select the cluster disk that you want the service or application to use in the new cluster, and then click **OK**.
    c. Repeat these steps for each cluster disk that you want to migrate to new storage. Then click **Next**.

Important

The Copy Cluster Roles Wizard does not move existing data and folders to the new storage. You must copy the folders and data manually.

11. Follow the instructions in the wizard to perform the migration. From the **Summary** page, we recommend that you read the Failover Cluster Post-Copy Roles Report, which describes any additional steps that you might need to complete before you bring the roles online. For example, if you have not already installed needed applications on the new cluster node, you might need to install them.

After the wizard completes, most migrated resources will be offline. Leave them offline at this stage.

Caution

At no time should a virtual machine be running on both the old cluster and the new cluster. A virtual machine that runs on both the old cluster and the new cluster at the same time might become corrupted. You can run a virtual machine on the old cluster while you migrate it to a new cluster with no problems; the virtual machine on the new cluster is created in a Stopped state. However, to avoid corruption, it is important that you do not turn on the virtual machine on the new cluster until after you stop the virtual machine on the old cluster.

Cluster roles: Post-migration tasks for a migration between two multi-node clusters

To complete the transition to the new cluster running Windows Server 2012 R2, perform the following steps. After you complete the transition, verify that the migrated workloads are available and are performing at the expected service levels, and verify that the cluster roles can successfully fail over within the new cluster.

To complete the transition from the old cluster to the new cluster

1. Prepare for clients to experience downtime, probably briefly.
2. On the old cluster, take the role and resource that were copied to the new cluster offline.
3. Complete the transition for the storage:
   - If the new cluster will use old storage, follow your plan for making LUNs or disks inaccessible to the old cluster and accessible to the new cluster.
   - If the new cluster will use new storage, copy the appropriate folders and data to the storage. As needed for disk access on the old cluster, bring individual disk resources online on that cluster. (Keep other resources offline, to ensure that clients cannot change data on the disks in storage.) On the new cluster, use Disk Management to confirm that the appropriate LUNs or disks are visible to the new cluster and not visible to any other servers.
4. If the new cluster uses mount points, adjust the mount points as needed, and make each disk resource that uses a mount point dependent on the resource of the disk that hosts the mount point. For more information about mount points, see Cluster Migrations Involving New Storage: Mount Points.
5. Bring the services and resources that were copied to the new cluster online.
6. If you migrated virtual machines, install the latest integration services on each virtual machine.

Note

The Copy Cluster Roles Wizard does not migrate Volume Shadow Copy Service (VSS) tasks, Hyper-V Replica Broker settings, Task Scheduler tasks, and Cluster-Aware Updating (CAU) settings. If you were using any of these features on the old cluster, you will need to configure them on the new cluster.

Cluster roles: Verify the migration

After you complete the transition to the new failover cluster, verify that the migrated workloads are available and are performing at the expected service levels, and verify that the cluster roles can successfully fail over within the new cluster.

- To verify that the migrated cluster roles are performing as expected on the new cluster
- To verify that the migrated cluster roles can fail over successfully
- To troubleshoot issues with failover for the migrated cluster roles

To verify that the migrated cluster roles are performing as expected on the new cluster

1. Verify that you can access the workload that was migrated. For example, can you connect to a highly available file server after it is migrated? Can you see the data that the server stores?
2. Run the necessary application-specific tests to ensure that the new cluster can provide the same service levels for the migrated workload that was provided before the clustered role was migrated.
3. If you migrated virtual machines, verify the status of the virtual machines in Hyper-V Manager, and confirm that you can connect to the virtual machines by using Remote Desktop or Virtual Machine Connection.

To verify that the migrated cluster roles can fail over successfully

1. In the console tree of **Failover Cluster Manager**, click the failover cluster on which the role is running.
2. Expand **Roles**, and then click a migrated role that you want to test.
3. On the **Actions** pane, expand **Roles**, and then click the cluster role that you want to test.

   To perform a basic test of failover for the copied cluster role, on the **Actions** pane, click **Move**, and then either select a node to move the role to (**Select Node** option) or move the role to the best possible node. When prompted, confirm your choice.

   You can observe the status changes in the center pane of Failover Cluster Manager as the cluster role is moved.

If there are any issues with failover, use the following procedure to troubleshoot those issues.

To troubleshoot issues with failover for the migrated cluster roles

1. View events in Failover Cluster Manager. To do this, in the console tree, right-click **Cluster Events**, and then click **Query**. In the **Cluster Events Filter** dialog box, select the criteria for the events that you want to display, or to return to the default criteria, click the **Reset** button. Click **OK**. To sort events, click a heading, for example, **Level** or **Date and Time**.
2. Confirm that necessary services, applications, or server roles are installed on all nodes. Confirm that services or applications are compatible with Windows Server 2012 R2 and run as expected.
3. If you used old storage for the new cluster, use the **Validate Cluster** action in Failover Cluster Manager to rerun the Validate a Configuration Wizard and confirm the validation results for all LUNs or disks in the storage.
4. Review migrated resource settings and dependencies. If you are using new storage that includes disks that use mount points, see Cluster Migrations Involving New Storage: Mount Points.
5. If you migrated one or more Network Name resources with the Kerberos protocol enabled, confirm that the computer account for the failover cluster has **Full Control** permission for the computer accounts (computer objects) of your Kerberos protocol-

enabled Network Name resources. On a domain controller, open **Active Directory Users and Computers**, and then verify the permissions for the appropriate computer accounts (computer objects).

In-Place Migration for a Two-Node Cluster: Migration to Windows Server 2012 R2

This topic provides an overview and steps for upgrading an existing failover cluster to Windows Server 2012 R2 when you have only two servers - that is, for performing an *in-place migration*.

Important

Before you begin the migration, confirm that the cluster role that you want to migrate can be migrated by using the Copy Cluster Roles Wizard, as described in Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012 R2, and note any preparation or follow-up steps that are required for the role that is being migrated.

Note

For an alternative approach to failover cluster migration, see Migrate Between Two Multi-Node Clusters: Migration to Windows Server 2012 R2. If you plan to migrate highly available Hyper-V virtual machines (by migrating the Virtual Machine cluster role), see Hyper-V: Hyper-V Cluster Migration for step-by-step instructions that use the Copy Cluster Roles Wizard to migrate virtual machines.

Overview of an in-place migration for a two-node cluster

The procedures in this topic describe the following process for upgrading an existing failover cluster to Windows Server 2012 R2 when only two servers are available.

1. Create a new cluster from a node in the old cluster – You will evict a node from the old cluster; upgrade that server to Windows Server 2012 R2 and install roles, features, and any needed software; prepare storage for the new cluster; and then create the Windows Server 2012 R2 failover cluster.
2. Copy the cluster roles to the new cluster – Use the Copy Cluster Roles Wizard to copy the clustered roles and features from the old cluster to the new cluster.
3. Perform post-migration tasks – Make existing data and files available to the new cluster; if you migrated to new storage, you will need to copy the data and files to the new storage location. Then bring the new cluster online, and verify that the migrated cluster roles and resources are available and are performing as expected.
4. Add the second node to the new cluster – First, you will destroy the old cluster. Then you will prepare the remaining node for the new cluster as you did the first node. Perform a complete

set of cluster validation tests to validate the configurations of both nodes. Add the second node to the new cluster. Then configure quorum settings on the new cluster.

5. Verify failover for the migrated cluster roles – After you add the second node to the new cluster, you can verify that the migrated cluster roles fail over successfully, and you can troubleshoot any issues with failover.

Impact of the migration

When you perform an in-place migration on a two-node failover cluster, you can prepare the destination failover cluster, configure storage, and copy the cluster role to the new cluster while maintaining service availability. High availability is lost when you evict the first node from the old cluster to use for the new cluster, and it is not restored until you have repurposed the remaining cluster node and added it to the new cluster.

Customers will experience a brief downtime after the cluster role is migrated and before you bring the role online on the new cluster.

If the new cluster will use the same storage that the old cluster is using, before you bring the role online on the new cluster, you must take the clustered role offline on the old cluster, mask the storage to the old cluster, unmask the storage to the new cluster, and then bring the volumes or disks online on the new cluster.

If you are migrating to new storage, before you can bring the role online on the new cluster, you must take the role offline on the old cluster, copy data and files for the clustered role to the new storage (the Copy Cluster Roles Wizard does not move data), and then bring the new storage online on the new cluster.

Access rights required to complete migration

To migrate a clustered role by using the Copy Cluster Roles Wizard, you must be a local administrator on the destination failover cluster and on the cluster or cluster node from which you are migrating.

Additional references

Migrate Between Two Multi-Node Clusters: Migration to Windows Server 2012 R2

Hyper-V: Migration Options

Hyper-V: Hyper-V Cluster Migration

Cluster Migrations Involving New Storage: Mount Points

Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012 R2

Windows Server Migration forum

Clustering Forum for Windows Server 2012

Create a new cluster from a node in the old cluster

For this phase of the migration, allow one existing server to continue running Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2 and the Cluster service while you prepare to migrate the cluster roles.

In this phase, you will perform the following tasks:

1. To evict a node from the old cluster
2. To prepare the node for the new cluster
3. To prepare storage for the new cluster
4. To create the new failover cluster and configure your firewall

To evict a node from the old cluster

1. Before you evict a node from a failover cluster, take the following precautions:
   - For each clustered role that you plan to migrate, verify that there are no special requirements or procedures for removing or evicting a node from the cluster. You can evict a node from a clustered file server or a cluster with the Hyper-V role with no special preparation. However, you might need to uncluster some services or applications before you evict a node.
   - If you are migrating from Windows Server 2008 R2, you should migrate all roles, including cluster core roles, to the remaining node before you evict a node.
   - To prevent any loss of application data when the node is evicted, shut down all services and applications on the cluster before you evict the node.
2. From the Start screen of either node in the cluster, open **Failover Cluster Manager**.
3. In the console tree, expand the cluster, expand Nodes, and then click the node that you want to evict to select it.
4. Right-click the node, click **More Actions**, and then click **Evict**.

To prepare the node for the new cluster

1. Perform a clean installation of Windows Server 2012 R2 on the server that you removed from the old cluster.
2. Add the Failover Clustering feature to the server.
3. If you plan to migrate highly available virtual machines, add the Hyper-V role to the server.
4. Install any other needed services, applications, and server roles. For example, if you plan to migrate clustered Windows Internet Name Service (WINS) to the new cluster, install the WINS Server feature by using Server Manager.
5. If you are migrating a Generic Application, Generic Script, or Generic Service resource, confirm that any associated application is compatible with Windows Server 2012 R2. You also must confirm that any associated service exists in Windows Server 2012 R2 and has the same name that it had in the old cluster. Test the application or service (separately,

not as part of a cluster) to confirm that it runs as expected.

To prepare storage for the new cluster

1. If you plan to migrate to existing storage, verify that your existing storage is certified for use with Windows Server 2012 R2.
2. Make an appropriate number of LUNs or disks accessible to the servers, and do not make those LUNs or disks accessible to any other servers. If the new cluster will use old storage, for testing purposes, you can limit the number of LUNs or disks to one or two. If the new cluster will use new storage, make as many disks or LUNs accessible to the new server as you think the cluster will need.

   Note
   We recommend that you keep a small disk or LUN available (unused by clustered services and applications) throughout the life of the cluster, so that you can always run storage validation tests without taking your services and applications offline.
3. Confirm that the intended cluster disks are visible and are formatted appropriately:
   a. On one of the servers that you plan to include in the cluster, open **Computer Management** from the Start screen, and then click **Disk Management** in the console tree.
   b. In Disk Management, confirm that the intended cluster disks are visible.
   c. Check the format of any exposed volume or LUN. We recommend that you use NTFS for the format. (For a disk witness, you must use NTFS.)
4. To prepare to migrate a highly available virtual machine, you must merge or discard all shadow copies that have been created for the virtual machine:
   a. Back up the volumes that store the virtual machines.
   b. Merge or discard shadow copies for each virtual hard disk (VHD).
   c. If you are migrating virtual machines stored on a Cluster Shared Volume (CSV) volume, make sure that you want to migrate all of the virtual machines on any volume that you plan to migrate. If you migrate one virtual machine that is stored on Cluster Shared Volume (CVS) volume, the Copy Cluster Roles Wizard migrates all virtual machines on that volume. This restriction does not apply when you migrate a Scale-out File Server cluster, which does not use CSV volumes.
5. If you are using new storage and your disk configuration uses mount points, review Cluster Migrations Involving New Storage: Mount Points to identify any additional steps that you need to perform.

To create the new failover cluster and configure your firewall

1. Create the new failover cluster. For information about how to create a Windows Server 2012 R2 failover cluster, see Create a Failover Cluster.
2. After you create the cluster, ensure that your firewall is configured appropriately. For example, if you are using Windows Firewall, and you will be sharing folders and files, use your preferred Windows Firewall interface to allow the exception for **Remote Volume**

Management.

Copy the cluster roles to the new cluster

Use the following instructions to copy cluster roles from your old one-node cluster to your new one-node cluster. The Copy Cluster Roles Wizard leaves most of the migrated resources offline so that you can perform additional steps before you bring them online.

Before you copy cluster roles to a new failover cluster

- If you plan to use new storage with the migrated clustered roles, before you run the Copy Cluster Roles Wizard, ensure that the storage is available to the new cluster – that is, ensure that the volumes have been added to the new cluster and that the volumes are online. This enables the wizard to update storage settings during migration.

To copy cluster roles from an existing cluster to a new cluster

1. From the **Start** screen or from **Server Manager** (**Tools**), open **Failover Cluster Manager**.
2. In the console tree, if the cluster that you created is not displayed, right-click **Failover Cluster Manager**, click **Connect to Cluster**, and then select the cluster that you want to configure.
3. In the console tree, expand the cluster that you created to see the items underneath it.
4. If the clustered servers are connected to a network that is not to be used for cluster communications (for example, a network intended only for iSCSI), then under **Networks**, right-click that network, click **Properties**, and then click **Do not allow cluster network communication on this network**. Click **OK**.
5. In the console tree, select the cluster.
6. Under **Configure**, click **Copy Cluster Roles**.

   The Copy Cluster Roles Wizard opens.
7. Read the Welcome page, and then click **Next**.
8. Specify the name or IP address of the cluster or cluster node from which you want to migrate services and applications, and then click **Next**.
9. The **Select Roles** page lists the clustered roles that can be migrated from the old cluster. The list does not contain any role that is not eligible for migration. Click **View Report** to view details in the Failover Cluster Pre-Copy Report. Then select each cluster role that you want to copy to the new cluster, and click **Next**.

   Important
   We recommend that you read the report, which explains whether each resource is eligible for migration. (The wizard also provides a report after it finishes, which describes any additional steps that might be needed before you bring the migrated resource groups online.)

   If storage is available on the new cluster, the **Specify Storage for Migration** page

appears, giving you the option to migrate to new storage. If storage is not available on the new cluster, the wizard retains existing storage settings and does not display the page.

Note
Not all clustered roles can be migrated to new storage. For example, the wizard cannot be used to migrate highly available virtual machines (the Virtual Machine role) to new storage. For an overview of options for migrating highly available virtual machines to a Windows Server 2012 R2 failover cluster and step-by-step instructions for each migration option, see Hyper-V: Hyper-V Cluster Migration.

10. If you want to use new storage for a service or application:
    a. On the **Specify Storage for Migration** page, select the cluster disk that you want to migrate to new storage, and then click **Select Storage**.
    b. In the **Select Storage for Resource Group** dialog box, under **Available Storage in New Cluster**, select the cluster disk that you want the service or application to use in the new cluster, and then click **OK**.
    c. Repeat these steps for each cluster disk that you want to migrate to new storage. Then click **Next**.

Important
The Copy Cluster Roles Wizard does not move existing data and folders to the new storage. You must copy the folders and data manually.

11. Follow the instructions in the wizard to perform the migration. From the **Summary** page, we recommend that you read the Failover Cluster Post-Copy Roles Report, which describes any additional steps that you might need to complete before you bring the roles online. For example, if you have not already installed needed applications on the new cluster node, you might need to install them.

After the wizard completes, most migrated resources will be offline. Leave them offline at this stage.

Caution
At no time should a virtual machine be running on both the old cluster and the new cluster. A virtual machine that runs on both the old cluster and the new cluster at the same time might become corrupted. You can run a virtual machine on the old cluster while you migrate it to a new cluster with no problems; the virtual machine on the new cluster is created in a Stopped state. However, to avoid corruption, it is important that you do not turn on the virtual machine on the new cluster until after you stop the virtual machine on the old cluster.

Perform post-migration tasks

During this phase of migration, you will perform the following tasks:

1. To make existing data available to the new cluster and bring the cluster online
2. To verify that the migrated cluster roles are performing as expected on the new cluster

To make existing data available to the new cluster and bring the cluster online

1. Prepare for clients to experience downtime, probably briefly.
2. On the old cluster, take the roles and resources that were copied to the new cluster offline.
3. Complete the transition of storage to the new cluster:
   - If the new cluster will use old storage, follow your plan for making LUNs or disks inaccessible to the old cluster and accessible to the new cluster.
   - If the new cluster will use new storage, copy the appropriate folders and data to the storage. As needed for disk access on the old cluster, bring individual disk resources online on that cluster. (Keep other resources offline to ensure that clients cannot change data on the disks in storage.) Then, on the new cluster node, use Disk Management to confirm that the appropriate LUNs or disks are visible to the new cluster and not visible to any other servers.
4. If the new cluster uses mount points, adjust the mount points as needed, and make each disk resource that uses a mount point dependent on the resource of the disk that hosts the mount point. For more information about mount points, see Cluster Migrations Involving New Storage: Mount Points.
5. Bring the cluster roles and resources that were copied to the new cluster online.
6. If you migrated virtual machines, install the latest integration services on each virtual machine.

Note

The Copy Cluster Roles Wizard does not migrate Volume Shadow Copy Service (VSS) tasks, Hyper-V Replica Broker settings, Task Scheduler tasks, and Cluster-Aware Updating (CAU) settings. If you were using any of these features on the old cluster, you will need to configure them on the new cluster.

To verify that the migrated cluster roles are performing as expected on the new cluster

1. Verify that you can access the workload that was migrated. For example, can you connect to a highly available file server after it is migrated? Can you see the data that the server stores?
2. Run the necessary application-specific tests to ensure that the new cluster can provide the same service levels for the migrated workload that was provided before the clustered role was migrated.
3. If you migrated virtual machines, verify the status of the virtual machines in Hyper-V Manager, and confirm that you can connect to the virtual machines by using Remote Desktop or Virtual Machine Connection.

Add the second node to the new cluster

Use the following procedures to prepare the second node and then add it to the new cluster. As part of this process, you will run validation tests that include both servers.

1. To delete the copied cluster roles and destroy the old cluster
2. To prepare the second node for the new cluster
3. To perform a full validation of both cluster nodes
4. To add the node to the cluster
5. To verify storage for the new cluster
6. To configure quorum settings for the new cluster

To delete the copied cluster roles and destroy the old cluster

1. From the Start screen, open **Failover Cluster Manager**.
2. Remove cluster roles that were copied to the new cluster:
   a. Expand the old cluster in the console tree, and then expand **Roles**.
   b. To delete a role, right-click the role, and click **Delete**.
3. To destroy the cluster, right-click the cluster in the console tree, click **More Actions**, and then click **Destroy Cluster**.

To prepare the second node for the new cluster

1. Perform a clean installation of Windows Server 2012 R2.
2. Add the Failover Clustering feature in the same way that you added it to the other server.
3. If the new cluster hosts virtual machines, add the Hyper-V role to the server.
4. Connect the newly installed server to the same networks and storage that the existing failover cluster node is connected to.
5. Install any other needed server roles, services, and applications.
6. Identify the disks or LUNs that are exposed to the new one-node failover cluster, and expose them to the newly installed server also.

   We recommend that you keep a small disk or LUN accessible to both nodes, and unused by clustered services and applications, throughout the life of the cluster. With this LUN, you can always run storage validation tests without taking your services and applications offline.

To perform a full validation of both cluster nodes

1. On either server running Windows Server 2012 R2, open **Failover Cluster Manager** from the Start screen.
2. In the console tree, confirm that **Failover Cluster Manager** is selected, and then, in the center pane, under **Management**, click **Validate Cluster**.
3. Follow the instructions in the Validate a Configuration Wizard, but this time, be sure to specify both servers (not just the existing cluster node) and specify that you want to run

all tests. Then, run the tests. Because two nodes are now being tested, a more complete set of tests runs, which takes longer than testing one node.

Important

If any cluster role is using a disk when you start the wizard, the wizard asks whether to take that cluster role offline for testing. If you choose to take a cluster role offline, the role remains offline until the tests are complete.

4. On the **Summary** page, which appears after the tests run, review the test results:
   - Click **View Report** and view the full set of test results in the Failover Cluster Validation Report.

     Notes

     To view the results of the tests after you close the wizard, open the report on the following path:

     <*SystemRoot*>\Cluster\Reports\Validation Report <*date and time*>.mht

     where <*SystemRoot*> is the folder in which the operating system is installed (for example, C:\Windows\).
   - To view Help topics to help you interpret the results, click **More about cluster validation tests**.
5. As necessary, make changes in the configuration, and then rerun the tests.

Note

For more information about failover cluster validation tests, see Validate Hardware for a Failover Cluster.

To add the node to the cluster

1. If the new cluster is not displayed, in the console tree, right-click **Failover Cluster Manager**, click **Connect to Cluster**, and then select the new cluster.
2. Select the new cluster in the console tree. Then, on the **Actions** pane, click **Add Node**.
3. Follow the instructions in the wizard to specify the server that you want to add to the cluster. On the **Summary** page, click **View Report** to review the tasks that the wizard performed.

   After the wizard closes, you can view the report in the <*SystemRoot*>\Cluster\Reports\ folder.

   Note

   After you close the wizard, in the center pane, you might see a warning about "Node Majority." You will correct this issue when you configure quorum settings for the new cluster.

To verify storage for the new cluster

1. In the console tree of Failover Cluster Manager, select the new cluster.

2. Expand **Storage**. Then check to see if all the disks that you want to make available to the new cluster are shown, either in one of the clustered services or applications or in **Available Storage**.
3. In most cases, you need at least one disk in **Available Storage** for your next task (specifying a witness disk). If you need to add a disk, on the **Actions** pane, click **Add Disk**, and follow the steps in the wizard.

   Before you can add a disk to storage, the disk must be accessible from both nodes in the cluster. To be used for a witness disk, a disk can be a relatively small, but must be at least 512 MB.

To configure quorum settings for the new cluster

1. In the console tree of Failover Cluster Manager, right-click the new cluster, click **More Actions**, and then click **Configure Cluster Quorum Settings**.
2. Follow the instructions in the wizard to select the most appropriate quorum setting for your needs. In most cases, this is the Node Majority quorum configuration, which requires that you specify an appropriate disk (from **Available Storage**) for the witness disk. For more information about quorum settings in Windows Server 2012 R2, see Configure and Manage the Quorum in a Windows Server 2012 Failover Cluster.

Verify failover for the migrated cluster roles

Earlier, after you copied the cluster roles to the new single-node cluster, you verified that workloads were accessible on the new cluster and that the services and applications performed as expected. Now that you have a multi-node cluster, and have configured quorum settings, you can verify failover for the migrated cluster roles.

To verify that the migrated cluster roles can fail over successfully

1. In the console tree of **Failover Cluster Manager**, click the failover cluster on which the role is running.
2. Expand **Roles**, and then click a migrated role that you want to test.
3. On the **Actions** pane, expand **Roles**, and then click the cluster role that you want to test.

   To perform a basic test of failover for the copied cluster role, on the **Actions** pane, click **Move**, and then either select a node to move the role to (**Select Node** option) or move the role to the best possible node. When prompted, confirm your choice.

   You can observe the status changes in the center pane of Failover Cluster Manager as the cluster role is moved.

If there are any issues with failover, use the following procedure to troubleshoot those issues.

To troubleshoot issues with failover for the migrated cluster roles

1. View events in Failover Cluster Manager. To do this, in the console tree, right-click **Cluster Events**, and then click **Query**. In the **Cluster Events Filter** dialog box, select the

criteria for the events that you want to display, or to return to the default criteria, click the **Reset** button. Click **OK**. To sort events, click a heading, for example, **Level** or **Date and Time**.

2. Confirm that necessary services, applications, or server roles are installed on all nodes. Confirm that services or applications are compatible with Windows Server 2012 R2 and run as expected.
3. If you used old storage for the new cluster, use the **Validate Cluster** action in Failover Cluster Manager to rerun the Validate a Configuration Wizard and confirm the validation results for all LUNs or disks in the storage.
4. Review migrated resource settings and dependencies. If you are using new storage that includes disks that use mount points, see Cluster Migrations Involving New Storage: Mount Points.
5. If you migrated one or more Network Name resources with the Kerberos protocol enabled, confirm that the computer account for the failover cluster has **Full Control** permission for the computer accounts (computer objects) of your Kerberos protocol-enabled Network Name resources. On a domain controller, open **Active Directory Users and Computers**, and then verify the permissions for the appropriate computer accounts (computer objects).

Cluster Migrations Involving New Storage: Mount Points

This topic describes considerations for configuring mount points during a migration to a failover cluster running Windows Server 2012 R2 or Windows Server 2012 when the destination cluster will use new storage after the migration.

Caution

If you want to use new storage, you must copy or move the data or folders (including shared folder settings) during a migration. The wizard for migrating clustered resources does not copy data from one shared storage location to another.

The Migrate a Cluster Wizard does not migrate mount point information (that is, information about hard disk drives that do not use drive letters, but are mounted instead in a folder on another hard disk drive). However, the wizard can migrate Physical Disk Resource settings to and from disks that use mount points. The wizard also does not configure the necessary dependency between the resources for mounted disks and the resource for a host disk (the disk on which the other disks are mounted). You must configure those dependencies after the wizard completes.

When you work with new storage for your cluster migration, you have some flexibility in the order in which you complete the tasks. You must create the mount points, run the Migrate a Cluster Wizard, copy the data to the new storage, and confirm the disk letters and mount points for the

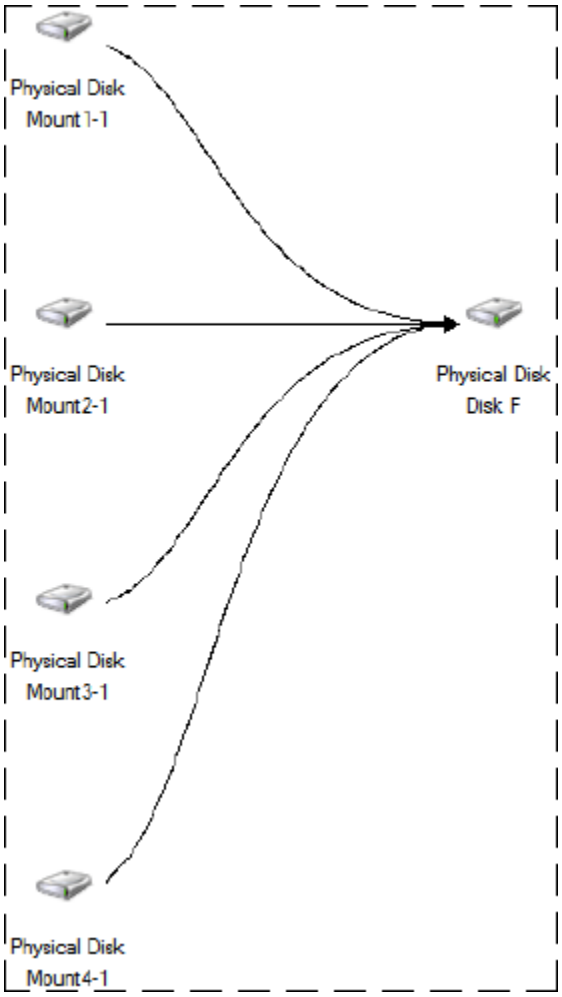
new storage. After completing those tasks, configure the disk resource dependencies in Failover Cluster Manager.

A useful way to keep track of disks in the new storage is to give them labels that indicate your intended mount point configuration. For example, in the new storage, when you are mounting a new disk in a folder called **Mount1-1** on another disk, you can also label the mounted disk as **Mount1-1**. (This assumes that the label **Mount1-1** is not already in use in the old storage.) When you run the Migrate a Cluster Wizard, and you need to specify that disk for a particular migrated resource, you can select the disk labeled **Mount1-1** from the list. After the wizard completes, you can return to Failover Cluster Manager to configure the disk resource for **Mount1-1** so that it is dependent on the appropriate resource - for example, the resource for disk **F**. Similarly, you would configure the disk resources for all other disks mounted on disk F so that they depended on the disk resource for disk F.

After you run the wizard and fully configure the mounted disk, your last task is to configure the disk dependencies in Failover Cluster Manager. For each disk resource for a mounted hard disk drive, open the Properties sheet and, on the **Dependencies** tab, specify a dependency on the disk resource for the host drive (where the mounted drives reside). This ensures that the Cluster service brings the host drive online first, followed by the drives that are dependent on it.

After you configure the dependencies, you can view a dependency report. To view a dependency report, click the service or application in Failover Cluster Manager, and then, under **Actions**, click **Show Dependency Report**. The following illustration shows four mount points that are configured with the correct dependencies on the disk on which they are mounted:

Four mount points with dependencies configured



Additional references

Migrate Cluster Roles to Windows Server 2012 R2

Migrating Clustered Services and Applications to Windows Server 2012

Additional References

- Overview of failover clusters:
  - What's New in Failover Clustering in Windows Server 2012 R2
  - Failover Clustering Overview
  - Failover Clustering Hardware Requirements and Storage Options
  - Validate Hardware for a Failover Cluster
- Community resources:

    - Windows Server Migration forum
    - Clustering Forum for Windows Server 2012
- Deploying failover clusters:
    - Create a Failover Cluster
    - Deploy an Active Directory-Detached Cluster
    - Deploy a Hyper-V Cluster
- Cluster configuration:
    - Configure and Manage the Quorum in a Windows Server 2012 Failover Cluster
    - Use Cluster Shared Volumes in a Failover Cluster
- Migrating highly available virtual machines:
    - Migrate Hyper-V to Windows Server 2012 R2 from Windows Server 2012
    - Hyper-V: Migration Options
    - Hyper-V: Hyper-V Cluster Migration

Migrate Network Policy Server to Windows Server 2012 R2

This document provides guidance for migrating the Network Policy Server (NPS) or Internet Authentication Server (IAS) role service from an x86-based or x64-based server running Windows Server 2003, Windows Server® 2008, Windows Server® 2008 R2, or Windows Server® 2012 to a new Windows Server® 2012 server.

About this guide

Note

Your detailed feedback is very important, and helps us to make Windows Server Migration Guides as reliable, complete, and easy to use as possible. Please take a moment to rate this topic by clicking the stars in the upper-right corner of the page (1=poor, 5=excellent), and then add comments that support your rating. Describe what you liked, did not like, or want to see in future versions of the topic. To submit additional suggestions about how to improve Migration guides or utilities, post on the Windows Server Migration forum.

NPS migration documentation and tools ease the migration of NPS role service settings and data from an existing server to a destination server that is running Windows Server 2012. By using the tools that are described in this guide, you can simplify the IAS/NPS migration process, reduce migration time, increase the accuracy of the IAS/NPS migration process, and help to eliminate possible conflicts that might otherwise occur during the migration process.

Target audience

This guide is intended for the following IT professionals:

- IT architects responsible for computer management and security throughout an organization.
- IT operations engineers who are responsible for the day-to-day management and troubleshooting of networks, servers, client computers, operating systems, or applications.
- IT operations managers who are accountable for network and server management.

What this guide does not provide

This guide does not provide detailed steps to migrate the configuration of other services that might be running on the source server.

Guidance is not provided for scenarios in which the new operating system is installed on existing server hardware by using the upgrade option during setup.

Supported migration scenarios

This guide provides the instructions for migrating an existing server that is running NPS or IAS to a server that is running Windows Server 2012. This guide does not contain instructions for Network Policy Server migration when the source server is running multiple roles. If your server is running multiple roles, it is recommended that you design a custom migration procedure specific to your server environment, based on the information provided in other role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server.

Caution

If your source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other roles that are running on the source server to fail.

Supported operating systems

The following table displays the minimum operating system requirements that are supported by this guide.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 SP2	Windows Server 2012 R2	x64-based
x86- or x64-based	Windows Server 2003 R2	Windows Server 2012 R2	x64-based
x86- or x64-based	Windows Server® 2008	Windows Server 2012 R2	x64-based

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x64-based	Windows Server 2008 R2	Windows Server 2012 R2	x64-based
x64-based	Windows Server 2012	Windows Server 2012 R2	x64-based
x64-based	Windows Server 2012 R2	Windows Server 2012 R2	x64-based

- The NPS role service is not available in Server Core editions. Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported as either source or destination servers. Windows Server Foundation edition is not available for Windows Server 2003.
- Migration from a source server to a destination server that is running an operating system with a different installed language is not supported. For example, migration of server roles from a computer that is running Windows Server 2008 with a system language of French to a computer that is running Windows Server 2012 R2 with a system language of German is not supported. The system language is the language of the localized installation package that was used to set up the Windows operating system.
- Both x86-based and x64-based migrations are supported for Windows Server 2003 and Windows Server 2008. All editions of Windows Server 2012 R2 are x64-based.

Supported NPS role configurations

Migration of the following NPS settings are supported by this guide:

1. **Policies**. Migration of NPS policy configuration, including connection request policies, network policies, and health policies is supported by using this guide.
2. **Authentication methods**. All supported authentication method settings can be migrated using this guide. For more information about authentication methods, see NPS Authentication Methods (http://go.microsoft.com/fwlink/?LinkId=169629).
3. **System Health Validators (SHVs)**. Migration of SHV configuration settings implemented using Microsoft published SDK are supported.
4. **NPS templates**. Template settings are migrated using NPS UI export and import functionality. You cannot migrate template settings using the command line.
5. **RADIUS clients and remote RADIUS servers**. RADIUS clients and remote RADIUS server configuration settings, including shared secrets can be migrated using this guide.
6. **SQL accounting**. The configuration of SQL parameters, including connection, description, accounting, authentication, periodic accounting status, periodic authentication status, and max sessions settings can be migrated using this guide. It is recommended to manually configure SQL connection string settings. For more information, see Configure SQL Server Logging in NPS (http://go.microsoft.com/fwlink/?LinkId=169631).

IP address and host name configuration

This guide supports the following scenarios:

1. The destination server is configured with the same host name or IP address as source server.
2. The destination server is configured with a different host name or IP address than the source server.

Migration scenarios that are not supported

The following migration scenarios are not covered in this document:

- **Upgrade**. Guidance is not provided for scenarios in which the new operating system is installed on existing server hardware by using the **Upgrade** option during setup.
- **Extension DLLs**. This guide does not support migration of extension DLL registry key settings. For more information about extension DLL registry key migration, see Setting Up the Extension DLLs (http://go.microsoft.com/fwlink/?LinkId=169632).
- **Non-Microsoft authentication methods**. The migration of settings for non-Microsoft authentication methods is not supported. To migrate these settings, refer to your vendor documentation.
- **Non-Microsoft SHVs**. The migration of settings for non-Microsoft SHVs is supported only if the SHV is developed using guidance from the NAP SHA/SHV SDK. To migrate these settings, refer to your vendor documentation.

Overview of migration process for this role

Network Policy Server (NPS) is the Microsoft implementation of a Remote Authentication Dial-in User Service (RADIUS) server and proxy in Windows Server 2012 R2. NPS is the replacement for Internet Authentication Service (IAS) in Windows Server 2003.

The current topic provides an overview of the NPS migration process. The NPS migration guide also includes the following major sections:

- Prepare to Migrate
- Migrating the NPS Server
- Verifying the NPS Server Migration
- Post-Migration Tasks
- Appendix A - Data Collection Worksheet

The pre-migration process involves establishing a storage location for migration data, collection of information that will be used to perform the server migration, and operating system installation on the destination server. The NPS migration process includes using the **iasmigreader** tool if the source server is running Windows Server 2003. If the source server is running Windows Server 2008 or Windows Server 2008 R2, the Network Shell (netsh) utility is used to obtain NPS settings. When migrating a source server running Windows Server 2012 or Windows Server 2012 R2, you can use netsh or Windows PowerShell®. Procedures are then performed on the destination server to install the required roles and migrate NPS settings. Verification procedures

include testing the destination server to ensure it works correctly. Post-migration procedures include retiring or repurposing the source server.

Impact of migration

In its recommended configuration, the destination server has the same host name and IP address as the source server. In this scenario, the source server will be unavailable to process network access requests for the duration of the migration process (estimated 1-2 hours).

This guide also includes procedures for migration of the NPS server configuration from the source server to a destination server with a different host name or IP address. This allows the source and destination NPS servers to run simultaneously until all testing and verification is complete, and reduces service disruption. If you change the name or IP address of the server running NPS, RADIUS clients must also be updated with the new NPS server name and IP address.

Impact of migration on the source server

- When deploying the destination server with the same host name and IP address as the source server, the source server must be decommissioned and taken offline prior to renaming the destination server from *tempNPS* to the host name of the source server.
- When deploying the destination server with a different host name and IP address, there is no impact to the source server.

Impact of migration on other computers in the enterprise

- When deploying the destination server with the same host name and IP address, network access requests cannot be evaluated by NPS while the source server is offline and before the destination server brought online with the same name and IP address. During this time, client computers requesting access to the network cannot authenticate and are denied network access.
- When deploying the destination server with a different host name and IP address, RADIUS client settings for all network access servers that are configured to use the source server must be updated.

Permissions required to complete migration

The following permissions are required on the source server and the destination server:

- Membership in the **Administrators** group, or the equivalent, is the minimum required to install and configure server running NPS.
- Membership in the SQL database rights are required for SQL settings migration.
- If the destination server is a domain member, membership in the **Domain Admins** group, or the equivalent, is the minimum required to authorize the NPS server.

Estimated duration

The work required to migrate NPS settings from the source to destination server, including testing, can require 1 to 2 hours. Additional time may be required for migration of non-Microsoft authentication methods, SHVs or extension DLLs.

Prepare to Migrate

Migration of Network Policy Server (NPS) includes the following tasks:

- Choose a migration file storage location
- Prepare your source server
- Prepare your destination server

Complete the steps or procedures in these sections to prepare your environment for migration.

If the server running NPS will be joined to a domain, membership in the **Domain Admins** group, or equivalent, is the minimum required to complete this procedure. If the server running NPS is not domain joined, membership in the **Administrators** group, or equivalent, is required. Review details about using the appropriate accounts and group memberships at Local and Domain Default Groups (http://go.microsoft.com/fwlink/?LinkId=83477).

Choose a migration file storage location

First, choose a location where migration files will be kept.

To choose a storage location

1. Select a file storage location where migration files will be kept. The storage location can be a network share that is accessible by both the source and destination server, or portable media that can be transferred from one server to another.

Prepare your source server

Follow these steps to prepare an x86-based or x64-based server running Windows Server 2003, Windows Server® 2008, Windows Server® 2008 R2, Windows Server® 2012, or Windows Server 2012 R2 for NPS migration.

To prepare the source server

1. Determine the domain, server name, IP address, and passwords on the source server.
2. If the source server is domain joined, determine the group membership of the source server in Active Directory Domain Services (AD DS), including security group and OU membership. This can be done using the Active Directory Users and Computers console (dsa.msc) or Server Manager on a domain controller.

Prepare your destination server

Follow these steps to prepare an x64-based destination server running Windows Server 2012 R2 for NPS migration.

Scenario 1: Prepare the destination server using the same host name and IP address

1. Install Windows Server 2012 R2 on the destination server.
2. If the source server host name is used by RADIUS clients or remote RADIUS server groups, name the destination server with a temporary server name, for example: *TempNPS*.
3. If the source server IP address is used by RADIUS clients or remote RADIUS server groups, assign a different temporary static IP address to the destination server.
4. If the source server is domain joined, add the destination server to the domain of the source server. Configure AD DS group membership settings on the destination server that are identical to the source server, including security group and OU membership.
5. Install the NPS role service using the steps provided in Install Network Policy Server (NPS) (http://go.microsoft.com/fwlink/?LinkId=169633).
6. If the source server has non-Microsoft authentication methods installed, then install same authentication methods on the destination server using your vendor documentation before importing the source server configuration.
7. If the source server has extension DLLs installed, install the same extension DLLs on the destination server before importing the source server configuration. For more information, see Setting Up the Extension DLLs (http://go.microsoft.com/fwlink/?LinkId=169632).
8. If the source server has non-Microsoft SHVs installed, then install same SHVs on the destination server using your vendor documentation before importing the source server configuration.

Scenario 2: Prepare the destination server using a different host name and IP address

1. Follow the same steps as provided for scenario 1, replacing the temporary server name with the new destination server host name, and assigning a permanent static IP address.

The destination server is now prepared for migration.

Migrating the NPS Server

This topic contains steps and procedures for migrating the Network Policy Server (NPS) role service from a legacy source server to a new x64-based destination server running Windows Server 2012 R2.

This topic includes sample Windows PowerShell cmdlets that you can use to automate some of the procedures described. For more information, see Using Cmdlets.

Known issues

If you previously created conditional attributes for your remote access policy using **Called Station ID** and **Calling Station ID**, the comparison of these attributes in Windows Server 2012 R2 uses a regular expression instead of matching the exact string. For a description of these attributes, see Remote Access Policy Conditions in the **IAS Authorization** section.

Exporting settings from the source server

Use the following procedures to export the NPS settings from your x86-based or x64-based source server prior to migrating to an x64-based server running Windows Server 2012. Follow the steps in the appropriate section based on the version of Windows Server that is running on the source server:

- Exporting settings from Windows Server 2003
- Exporting settings from Windows Server 2008
- Exporting settings from Windows Server 2008 R2
- Exporting settings from Windows Server 2012 or Windows Server 2012 R2

Warning

When you use the following procedures to export configuration settings, apply appropriate precautions when moving these files from the source server to destination servers. NPS server configurations are not encrypted in the exported XML file, and contain shared secrets for RADIUS clients and members of remote RADIUS server groups. Therefore, sending these files over a network connection might pose a security risk. You can add the file to an encrypted, password protected archive file before moving the file to provide greater security. In addition, store the file in a secure location to prevent access by unauthorized users.

Exporting settings from Windows Server 2003

Configuration settings for Internet Authentication Service (IAS) in Windows Server 2003 are stored in **.MDB** files. Configuration settings for Network Policy Server (NPS) in Windows Server 2012 are stored in **.XML** files. **Iasmigreader.exe** is a command-line tool that exports the configuration settings of IAS on a computer running Windows Server 2003 to a text file. You can obtain the **iasmigreader.exe** command line migration tool for migrating Windows Server 2003 IAS settings to Windows Server 2012 from the following locations:

1. Windows Server 2012 installation media provides a copy of the migration tool in the **\sources\dlmanifests\microsoft-windows-iasserver-migplugin** directory.
2. The migration tool is available in the **%windir%\syswow64** directory on a server running Windows Server 2012.

To export settings from a source server running Windows Server 2003

1. Copy **iasmigreader.exe** to the source server into a directory configured in the **%path%**

environment variable.

Tip

To review the source server's **%path%** configuration, type **echo %path%** at a command prompt and press Enter.

2. At an elevated command prompt, type **iasmigreader.exe**, and then press Enter. The migration tool will automatically export settings to a text file.

Important

Configuration changes made to IAS will take at least one minute to be available for export.

3. IAS settings are stored in the file **ias.txt** located in the **%windir%\system32\ias** directory on the source server. If you are running a 64-bit version of Windows Server 2003, the **ias.txt** file is located in the **%windir%\syswow64\ias** directory.
4. You must manually copy SQL log configuration settings on the source server to a file (example: sql.txt).

   To record these settings:

   a. At an elevated command prompt, type **ias.msc**, and then press Enter.
   b. In the IAS console tree, click **Remote Access Logging**, right-click **SQL Server**, and then click **Properties**.
   c. Record the configuration settings on the **Settings** tab, and then click **Configure**.
   d. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.
5. Copy the ias.txt and sql.txt files to the migration store file location.

Warning

Store the ias.txt and sql.txt files in a secure location. These files contain shared secret information and SQL connection strings.

Important

When you migrate the configuration settings of the IAS role service that is running on a 32-bit or a 64-bit Windows Server 2003–based source server to the NPS role service that is running on a Windows Server 2012 R2–based destination server, the import procedure seems to complete successfully. However, the Extensible Authentication Protocol (EAP) method is misconfigured. This occurs because the migration tool generates a faulty parameter that is stored in the configuration text file (ias.txt). For more information about this issue and for a workaround, see The EAP method is configured incorrectly during the migration process from Windows Server 2003 32-bit or a 64-bit to Windows Server 2008 R2 (http://go.microsoft.com/fwlink/?LinkID=181982).

Exporting settings from Windows Server 2008

Configuration settings for NPS in Windows Server 2008 are stored in **.XML** files that can be directly imported to the destination server. The Network Shell (NetSh) command line utility can be used to export and import these settings. You can also use the Windows interface to import and export these settings.

Warning

You cannot use the Windows interface or a command line to export or import detailed SQL configuration settings. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.

To export settings from a source server running Windows Server 2008 using a command line

1. On the source NPS server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps export filename="path\file.xml" exportPSK=YES
   ```

   Replace *path* with the directory location where you want to save the source server configuration file, and replace *file* with the name of the .XML file that you want to save.
2. Confirm that a message appears indicating that the export to file was successful.
3. On the source server, type the following command and then press Enter:

   ```
   netsh nps show sqllog > path\sql.txt
   ```

   Replace *path* with the directory location where you want to save the source server SQL configuration file, and replace *sql* with the name of the .TXT file that you want to save. This file contains the basic configuration for SQL logging that is found on the **Settings** tab in SQL logging properties. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.
4. Copy the **file.xml** and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server running Windows Server 2008 using the Windows interface

1. On the source server, open Server Manager.
2. In the Server Manager console tree, open **Roles\Network Policy and Access Services\NPS**.
3. Right click **NPS**, and then click **Export Configuration**.
4. In the dialog box that appears, select the check box next to **I am aware that I am exporting all shared secrets**, and then click **OK**.
5. Next to **File name**, type **file.xml**, navigate to the migration store file location, and then click **Save**.
6. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

To record these settings:

a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.

b. Record the configuration settings on the **Settings** tab, and then click **Configure**.

c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.

7. Copy the ias.txt and sql.txt files to the migration store file location.

Exporting settings from Windows Server 2008 R2

Configuration settings for NPS in Windows Server 2008 R2 are stored in **.XML** files that can be directly imported to the destination server. The Network Shell (NetSh) command line utility can be used to export and import these settings. You can also use the Windows interface to import and export settings.

Warning

You cannot use the Windows interface or a command line to export or import detailed SQL configuration settings. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.

Important

The netsh utility does not support migration of template configuration settings. To migrate these settings, you must use the Windows interface.

To export settings from a source server running Windows Server 2008 R2 using a command line

1. On the source NPS server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps export filename="path\file.xml" exportPSK=YES
   ```

   Replace *path* with the directory location where you want to save the source server configuration file, and replace *file* with the name of the .XML file that you want to save.

2. Confirm that a message appears indicating that the export to file was successful.

3. On the source server, type the following command and then press Enter:

   ```
   netsh nps show sqllog > path\sql.txt
   ```

   Replace *path* with the directory location where you want to save the source server SQL configuration file, and replace *sql* with the name of the .TXT file that you want to save. This file contains the basic configuration for SQL logging that is found on the **Settings** tab in SQL logging properties. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.

4. Copy the **file.xml** and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server running Windows Server 2008 R2 using the Windows interface

1. On the source server, open Server Manager.
2. In the Server Manager console tree, open **Roles\Network Policy and Access Services\NPS**.
3. Right click **NPS**, and then click **Export Configuration**.
4. In the dialog box that appears, select the check box next to **I am aware that I am exporting all shared secrets**, and then click **OK**.
5. Next to **File name**, type **file.xml**, navigate to the migration store file location, and then click **Save**.
6. In the console tree, right-click **Templates Management** and then click **Export Templates to a file**.
7. Next to **File name**, type **iastemplates.xml**, navigate to the migration store file location, and then click **Save**.
8. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

   To record these settings:

   a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.

   b. Record the configuration settings on the **Settings** tab, and then click **Configure**.

   c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.
9. Copy the **file.xml**, **iastemplates.xml**, and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

Exporting settings from Windows Server 2012 or Windows Server 2012 R2

Configuration settings for NPS in Windows Server 2012 R2 are stored in **.XML** files that can be directly imported to the destination server. You can use the following methods to export and import these settings:

1. The Network Shell (NetSh) command line utility
2. The Windows interface
3. Windows PowerShell cmdlets

Warning

You cannot use Windows PowerShell, the Windows interface or a command line to export or import detailed SQL configuration settings. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.

Important

The netsh utility and Windows PowerShell do not support migration of template configuration settings. To migrate these settings, you must use the Windows interface.

To export settings from a source server using Windows PowerShell

1. On the source server, create a new folder for your settings (for example: C:\ConfigSettings).
2. Export your configuration settings to an .xml file in that folder, by following these steps.
   a. On the **Start** screen, type **PowerShell**, and then click Enter.
   b. To switch to the NPS context enter the following Windows PowerShell command and then press Enter:

      Import-Module NPS

   c. To export the configuration file to an .xml file, enter the following Windows PowerShell command, using the -path parameter to identify the name of the .xml file to be created and the folder into which it should be placed:

      Export-NpsConfiguration [-Path] <String>

      Tip

      For example:

      Export-NpsConfiguration –Path C:\ConfigSettings -Path nps01.xml

      Caution

      The exported file contains unencrypted shared secrets for RADIUS clients and members of remote RADIUS server groups. Because of this, you should ensure that the file is stored in a secure location to prevent malicious users from accessing the file.
3. Confirm that no errors were reported by Windows PowerShell.
4. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

   To record these settings:

   a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.
   b. Record the configuration settings on the **Settings** tab, and then click **Configure**.
   c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text

logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.

5. Copy the **file.xml**, **iastemplates.xml**, and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server using the Netsh utility

1. On the source NPS server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps export filename="path\file.xml" exportPSK=YES
   ```

   Replace *path* with the directory location where you want to save the source server configuration file, and replace *file* with the name of the .XML file that you want to save.
2. Confirm that a message appears indicating that the export to file was successful.
3. On the source server, type the following command and then press Enter:

   ```
   netsh nps show sqllog > path\sql.txt
   ```

   Replace *path* with the directory location where you want to save the source server SQL configuration file, and replace *sql* with the name of the .TXT file that you want to save. This file contains the basic configuration for SQL logging that is found on the **Settings** tab in SQL logging properties. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.
4. Copy the **file.xml** and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server using the Windows interface

1. On the source server, open Server Manager.
2. In the Server Manager console tree, click **ALL SERVERS**, then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
3. Right click the root node **NPS**, and then click **Export Configuration**.
4. In the dialog box that appears, select the check box next to **I am aware that I am exporting all shared secrets**, and then click **OK**.
5. Next to **File name**, type **file.xml**, navigate to the migration store file location, and then click **Save**.
6. In the console tree, right-click **Templates Management** and then click **Export Templates to a file**.
7. Next to **File name**, type **iastemplates.xml**, navigate to the migration store file location, and then click **Save**.
8. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

   To record these settings:

   a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.

b. Record the configuration settings on the **Settings** tab, and then click **Configure**.

c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see Appendix A - Data Collection Worksheet.

9. Copy the **file.xml**, **iastemplates.xml**, and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

Importing settings to the destination server

Use the following procedures to import the NPS settings from your x86-based or x64-based source server to an x64-based destination server running Windows Server 2012 R2.

- Importing settings from Windows Server 2003
- Importing settings from Windows Server 2008 or Windows Server 2008 R2
- Importing settings from Windows Server 2012 or Windows Server 2012 R2

Importing settings from Windows Server 2003

The configuration file **ias.txt** that was exported from the source server is in a format that can be imported to a destination server running Windows Server 2012 or Windows Server 2012 R2. If SQL accounting settings were saved, these settings are recorded manually in the **sql.txt** file.

Important

When you migrate the configuration settings of the IAS role service that is running on a 32-bit or a 64-bit Windows Server 2003–based source server to the NPS role service that is running on a Windows Server 2012 R2–based destination server, the import procedure seems to complete successfully. However, the Extensible Authentication Protocol (EAP) method is misconfigured. This occurs because the migration tool generates a faulty parameter that is stored in the configuration text file (ias.txt). For more information about this issue and for a workaround, see The EAP method is configured incorrectly during the migration process from Windows Server 2003 32-bit or a 64-bit to Windows Server 2008 R2 (http://go.microsoft.com/fwlink/?LinkID=181982).

To import settings from a source server running Windows Server 2003

1. Copy the configuration file **ias.txt** that was exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command.
2. On the destination server, use either netsh or Windows PowerShell to import the configuration.
   - To use netsh, do the following:

a. Open an elevated command prompt, type the following command and then press Enter:

```
netsh nps import filename="path\ias.txt"
```

Replace *path* with the directory where the **ias.txt** file is located. Verify that a message appears indicating that the import process was successful.

Tip

If the configuration file is located on a network share, provide full path to the file. For example: **netsh nps import filename = "\\fileserver1\Data\ias.txt"**.

- To use Windows PowerShell, do the following:

a. On the **Start** screen, type **PowerShell**, and then click Enter.

b. Switch to the NPS context, enter the following Windows PowerShell command:

Import-Module NPS

c. To import the configuration, enter the following:

Import-NpsConfiguration [-Path] <String>

Replace *String* with the directory where the **ias.txt** file is located. Verify that a message appears indicating that the import process was successful.

Tip

For example:

Import-NpsConfiguration –Path c:\temp\ias.txt

3. If required, configure SQL accounting. To configure SQL accounting:
   a. In the Server Manager console tree, click **ALL SERVERS**, then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
   b. Click **Accounting** and then click **Change SQL Server Logging Properties**.
   c. Manually enter SQL settings from the **sql.txt** file that you created.

Importing settings from Windows Server 2008 or Windows Server 2008 R2

The configuration file **file.xml** that was exported from the source server is in a format that can be imported to a destination server running Windows Server 2012. SQL accounting settings are saved in the **sql.txt** file.

Note

For source servers running Windows Server 2008 R2: If you saved a templates configuration file, **iastemplates.xml**, you must use the Windows interface to import these settings.

To import settings from a source server running Windows Server 2008 or Windows Server 2008 R2

1. Copy the configuration files **file.xml** and **sql.txt** that were exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command.
2. On the destination server, use either netsh or Windows PowerShell to import the configuration.
   - To use netsh, do the following:

     a. Open an elevated command prompt, type the following command and then press Enter:

     ```
     netsh nps import filename="path\file.xml"
     ```

     Replace *path* with the directory where the **file.xml** file is located. Verify that a message appears indicating that the import process was successful.

     Tip

     If the configuration file is located on a network share, provide full path to the file. For example: **netsh nps import filename = "\\fileserver1\Data\file.xml"**.

   - To use Windows PowerShell, do the following:

     a. On the **Start** screen, type **PowerShell**, and then click Enter.

     b. Switch to the NPS context, enter the following Windows PowerShell command:

     Import-Module NPS

c. To import the configuration, enter the following:

Import-NpsConfiguration [-Path] <String>

Replace <*String*> with the directory where the **file.xml** file is located.

Tip

For example:

Import-NpsConfiguration –Path c:\temp\file.xml

d. Confirm that no errors were reported by Windows PowerShell.

3. If required, configure SQL accounting. To configure SQL accounting:
   a. In the Server Manager console tree, click **ALL SERVERS**, then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
   b. Click **Accounting** and then click **Change SQL Server Logging Properties**.
   c. Manually enter SQL settings from the **sql.txt** file.

Importing settings from Windows Server 2012 or Windows Server 2012 R2

The configuration file **file.xml** that was exported from the source server is in a format that can be imported to a destination server running Windows Server 2012 or Windows Server 2012 R2. SQL accounting settings are saved in the **sql.txt** file. If you saved a templates configuration file, **iastemplates.xml**, you must use the Windows interface to import these settings.

To import settings from a source server

1. Copy the configuration files **file.xml** and **sql.txt** that were exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command.
2. On the destination server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps import filename="path\file.xml"
   ```

   Replace *path* with the directory where the **file.xml** file is located. Verify that a message appears indicating that the import process was successful.

   Tip

   If the configuration file is located on a network share, provide full path to the file. For example: **netsh nps import filename = "\\fileserver1\Data\file.xml"**.

   The following Windows PowerShell command performs the same function:

   ```
   Import-NpsConfiguration –Path c:\temp\file.xml
   ```

3. If required, configure SQL accounting. To configure SQL accounting:

a. In the Server Manager console tree, click **ALL SERVERS**, then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
b. Click **Accounting** and then click **Change SQL Server Logging Properties**.
c. Manually enter SQL settings from the **sql.txt** file.

Using the NPS console to migrate NPS settings

You can also use the Windows interface on the destination server to import configuration settings.

To import settings from a source server using the Windows interface

1. Copy the configuration files **file.xml**, **iastemplates.xml**, and **sql.txt** that were exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command. If you have custom settings that were recorded using the Appendix A - Data Collection Worksheet, these must be configured manually on the destination server.
2. On the destination server, open Server Manager.
3. In the Server Manager console tree, click **ALL SERVERS**, and then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
4. To import template configuration settings, follow steps 5 to 13. If you do not have template settings, skip to step 7.
5. In the console tree, right-click **Templates Management** and then click **Import Templates from a file**.
6. Select the template configuration file **iastemplates.xml** that you copied from the source server and then click **Open**.
7. In the console tree, right-click **NPS** and then click **Import Configuration**.
8. Select the configuration file **file.xml** or **ias.txt** that you copied from the source server and then click **Open**.
9. Verify that a message appears indicating the import was successful.
10. Configure SQL accounting if required using the **sql.txt** file and the data collection worksheet. To configure SQL accounting, follow steps 11 to 13.
11. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties** in the details pane.
12. Modify the properties on the **Settings** tab if required, and then click **Configure** to enter detailed settings.
13. Using information recorded in the **sql.txt** file, enter the required settings on the **Connection** and **Advanced** tabs, and then click **OK**.

Verifying the NPS Server Migration

After the migration of your Network Policy Server (NPS) server is complete, you can perform some tasks to verify that the migration was successful.

Verifying NPS Migration

To verify the functionality of NPS on the destination server, confirm that the service is running, that the correct configuration was migrated, and that client computers can authenticate successfully.

To verify NPS migration

1. To verify that the NPS service is running on the destination server, type the following command at an elevated command prompt on the destination server and then press ENTER.

   ```
   sc query ias
   ```

   In the command output, verify that **RUNNING** is displayed next to **STATE**.
2. To verify that the source NPS configuration has been migrated to the destination server, type the following command at an elevated command prompt on the destination server and then press ENTER:

   ```
   netsh nps show config
   ```

   Verify that the destination server is not using default NPS settings. For example, default settings display a single policy under **Connection request policy configuration** with the name **Use Windows authentication for all users**.
3. To verify that the NPS console on the destination server displays the correct settings, type the following command at an elevated command prompt on the destination server and then press ENTER:

   ```
   nps.msc
   ```

   a. The NPS console will open. In the console tree, click **Accounting**, click **Change SQL Server Logging Properties**, click **Configure**, and verify that the correct settings are displayed on the **Connection** and **Advanced** tabs.

   b. In the NPS console tree, click **Policies** and then click **Connection Request Policies**, **Network Policies**, and **Health Policies**. For each type of policy, verify that the correct policies are displayed.

   c. In the NPS console tree, click **RADIUS Clients and Servers** and then click **RADIUS Clients and Remote RADIUS Server Groups**. Verify that the correct RADIUS clients and remote RADIUS server groups are displayed.

   d. In the NPS console tree, click **Network Access Protection**, and then click **System Health Validators** and **Remediation Server Groups**. Verify that the correct Network Access Protection (NAP) related settings are displayed.

e. In the NPS console tree, click **Templates Management**. If the source server was running Windows Server 2008 R2, verify that the correct templates settings are displayed.

f. In the NPS console tree, right-click **NPS**, click **Properties**, and then click the **Ports** tab. Verify that the correct **Authentication** and **Accounting** ports are displayed.

4. To verify the configuration of authentication methods, you must manually review settings in connection request policy and network policy. Certificate based EAP methods require that the proper certificate is chosen, and might require that you provision a computer certificate on the destination server.

Verifying authentication methods

a. If you use certificate based EAP methods, your destination server might already be provisioned with a suitable certificate through autoenrollment. You might also be required to manually enroll the destination server with a computer certificate. For an overview of certificate requirements for network authentication, see Network access authentication and certificates (http://go.microsoft.com/fwlink/?LinkId=169625).

b. To view certificates associated with EAP methods, click **Start**, click **Run**, type **nps.msc**, and press ENTER.

c. In the NPS console tree, open **Policies** and then open the type of policy you are using to perform authentication. For example, if the option to **Override network policy authentication settings** is enabled on the **Settings** tab in a connection request policy, then authentication is performed in connection request policy. Otherwise, authentication is performed in network policy. Authentication can be configured in both types of policies.

d. For connection request policy, double-click the policy name and then click the **Settings** tab. For network policy, double-click the policy name and then click the **Constraints** tab.

e. Click **Authentication Methods**, and then under **EAP Types** click the name of the certificate-based authentication method. For example: **Microsoft: Protected EAP (PEAP)** or Microsoft: Smart Card or other certificate.

f. Click **Edit**, verify that the correct certificate is chosen next to **Certificate issued** or **Certificate issued to**, and then click **OK**.

Note
Client computers using certificate based authentication methods must trust the certification path for this certificate.

5. To verify that client computers can authenticate using the destination server, attempt to connect to the network using client VPN connection, an 802.1X connection, or another connection that requires successful RADIUS authentication for network access.

Verifying client connections

a. To verify that client computers are successfully connecting to the network, click **Start**, click **Run**, type **eventvwr.msc**, and then press ENTER.
b. In the event viewer console tree, open **Custom Views\Server Roles\Network Policy and Access Services**.
c. In the details pane, verify under **Event ID** that event number 6272 is displayed.
d. Events 6273 or 6274 indicate that client authentication attempts are unsuccessful.
e. If no events are displayed, client connection requests are unable to reach the destination server, or the server is not logging authentication attempts.

Post-Migration Tasks

After all migration steps are complete and you have verified the migration of the Network Policy Server (NPS) role service, perform the following post-migration tasks.

Post migration tasks

After verifying NPS configuration is working on the destination server, the following steps need to be performed:

To decommission a source server using the same host and IP address

1. Remove the source server from your Active Directory domain.
2. Shut down the source server.
3. Rename the destination server from *tempNPS* to the name of the source server and configure the same static IP address as that used by the source server.
4. Perform verification steps in Verifying the NPS Server Migration with the updated host name and IP address configured on the destination server.

To decommission a source server using a different host and IP address

1. NPS server name/ IP address should be updated on Remote RADIUS servers and RADIUS clients. It requires manual update of the configurations on RADIUS clients and Network Access Servers (NAS). Please refer to your RADIUS client configuration guide for more information.
2. Perform verification steps in Verifying the NPS Server Migration.
3. When the destination server has been configured, tested, and verified, then the NPS role on the source server may be retired.

Restoring the role in the event of migration failure

If the destination server is deployed simultaneously with the source server using a different host name and IP address, then the migration can be reversed by changing RADIUS clients, remote RADIUS server groups, and network access device settings to use the source NPS server name and IP address. If the destination server is replacing the source server using the same host name and IP address, then the destination server will need to be renamed, the IP address must be updated, and the destination server must be removed from the domain to reverse the migration and bring the source server back online.

Appendix A - Data Collection Worksheet

Migration data collection worksheet

You can use this migration data collection worksheet to collect data about your source server and help ensure that the destination server functions properly after the migration.

NPS data worksheet

#	Source server essential settings	Setting values
1	**Server name** At a command prompt, type the following command, and then press ENTER. `ipconfig /all` The host name of a server is the first part of the fully qualified domain name (FQDN). The FQDN is the full computer name, including both the host name and the primary DNS suffix, separated by dots (.). For example, the FQDN of a computer named *host* with a primary DNS suffix of *example.microsoft.com* is *host.example.microsoft.com*.	Computer host name: _____ FQDN: _____
2	**Authentication, authorization, and accounting (AAA) roles** Determine what types of network access requests are validated using the RADIUS protocol on the source	Check all that apply (√) ☐ Network Access Protection (NAP) ☐ RADIUS server for dial-Up or VPN connections ☐ RADIUS server for 802.1X wireless

#	Source server essential settings	Setting values
	server.	or wired connections
3	**Text logging** Record the path and settings used for text logging. By default, local file accounting logs are stored in **%windir%\system32\LogFiles**.	Local file logging directory: _____ Format: _____ Create a new log file: _____
4	**SQL settings** Manually record any customized SQL data link properties.	Application Name: _____ Auto Translate: _____ Connect Timeout: _____ Current Language: _____ Data Source: _____ Extended Properties: _____ General Timeout: _____ Initial Catalog: _____ Initial File Name: _____ Integrated Security: _____ Locale Identifier: _____ Network Address: _____ Network Library: _____ Packet Size: _____ Password: _____ Persistent Security Info: _____

#	Source server essential settings	Setting values
		Replication server name connect option: _____ Tag with column collation when possible: _____ Use Encryption for Data: _____ Use Procedure for Prepare: _____ User ID: _____ Workstation ID: _____

Migrate Roles and Features to Windows Server 2012

Migration documentation and tools ease the process of migrating server roles, features, operating system settings, and data from an existing server that is running Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012 to a computer that is running Windows Server 2012. By using migration guides linked to on this page (and where appropriate, Windows Server Migration Tools) to migrate roles, role services, and features, you can simplify deployment of new servers (including those that are running the Server Core installation option of Windows Server 2012, and virtual servers), reduce migration downtime, increase accuracy of the migration process, and help eliminate conflicts that could otherwise occur during the migration process.

In this section

- Install, Use, and Remove Windows Server Migration Tools
- Migrate Active Directory Federation Services Role Services to Windows Server 2012
- Migrate File and Storage Services to Windows Server 2012
- Migrate Health Registration Authority to Windows Server 2012
- Migrate Hyper-V to Windows Server 2012 from Windows 2008 R2
- Migrate IP Configuration to Windows Server 2012
- Migrate Network Policy Server to Windows Server 2012
- Migrate Print and Document Services to Windows Server 2012
- Migrate Remote Access to Windows Server 2012
- Migrate Windows Server Update Services to Windows Server 2012

- Migrating Clustered Services and Applications to Windows Server 2012

See Also

Migrating Roles and Features to Windows Server

Install, Use, and Remove Windows Server Migration Tools

Windows Server Migration Tools Installation, Access, and Removal describes how to locate, install, use, and remove Windows Server Migration Tools. Administrators can use Windows Server Migration Tools to migrate server roles, features, operating system settings, and other data and shares to computers that are running Windows Server® 2012 R2 or Windows Server® 2012.

This topic supports migrations in which the migration destination servers are running Windows Server 2012 R2 or Windows Server 2012. For information about how to prepare to use Windows Server Migration Tools for migrations to servers that are running Windows Server 2008 R2, see Windows Server Migration Tools Installation, Access, and Removal.

Windows Server Migration Tools installation and preparation can be divided into the following stages.

1. Installing Windows Server Migration Tools on destination servers that run Windows Server 2012 R2 or Windows Server 2012.
2. Creating deployment folders on migration destination servers, for copying to source servers.
3. Copying deployment folders from destination servers to source servers.
4. Registering Windows Server Migration Tools on source servers.

In this guide

Supported operating systems

The following table indicates the Windows Server operating systems that Windows Server Migration Tools supports.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 with Service Pack 2	Windows Server 2012 R2 or Windows Server 2012, both full and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2003 R2	Windows Server 2012 R2 or Windows Server 2012, both full and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2008, full installation option	Windows Server 2012 R2 or Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2012 R2 or Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2008 R2	Windows Server 2012 R2 or Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Windows Server 2012	Windows Server 2012 R2 or Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2012	Windows Server 2012 R2 or Windows Server 2012, both full	x64-based

Source server processor	Source server operating system	Destination server operating system	Destination server processor
		and Server Core installation options	
x64-based	Windows Server 2012 R2	Windows Server 2012 R2, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2012 R2	Windows Server 2012 R2, both full and Server Core installation options	x64-based

The versions of operating systems shown in the previous table are the oldest combinations of operating systems and service packs that are supported. If available, newer service packs are supported.

Migrations between physical operating systems and virtual operating systems are supported. Migrations that use Windows Server Migration Tools to migrate to Windows Server 2012 or Windows Server 2012 R2 support cross-subnet migrations.

Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shares from a computer that is running Windows Server 2008 in the French system UI language to a computer that is running Windows Server 2012 in the German system UI language.

Note

The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Both x86- and x64-based migrations are supported for Windows Server 2003 and Windows Server 2008. All editions of Windows Server 2012 R2, Windows Server 2012, and Windows Server 2008 R2 are x64-based.

Roles that are running on the Server Core installation option of Windows Server 2008 cannot be migrated, because the Microsoft .NET Framework is not available in the Server Core installation option of Windows Server 2008.

Permission requirements

At minimum, you must be a member of the **Administrators** group on both source and destination servers to install, remove, or set up Windows Server Migration Tools.

Prepare for installation

Follow the steps in this section if you are registering Windows Server Migration Tools on migration source servers that are running Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012, and if the source server is running an older release of Windows Server than the migration destination server. For example, if the source server is running Windows Server 2012, but the destination server is running Windows Server 2012 R2. Otherwise, see Install Windows Server Migration Tools.

Note

All commands in this guide are case-insensitive unless specifically noted.

Windows Server 2012 source server

Complete the following tasks to prepare a source server that is running Windows Server 2012 for migration in which the destination server is running Windows Server 2012 R2.

- Verify that the source server has sufficient disk space (at least 23 MB) to store the Windows Server Migration Tools deployment folder.

Windows Server 2008 R2 source server

Complete the following tasks to prepare a source server that is running Windows Server 2008 R2 for Windows Server Migration Tools.

- Verify that the source server has sufficient disk space (at least 23 MB) to store the Windows Server Migration Tools deployment folder.

Windows Server 2008 source server

Complete the following tasks to prepare a source server that is running Windows Server 2008 for Windows Server Migration Tools.

- Verify that the source server has sufficient disk space (at least 23 MB) to store the Windows Server Migration Tools deployment folder.
- Install Windows PowerShell by using Server Manager or by running the Server Manager command prompt tool, **ServerManagerCmd.exe**. For more information about how to add features to the server by using **ServerManagerCmd.exe**, see Overview of Server Manager Commands in the Windows Server 2008 Server Manager Help.

Windows Server 2003 or Windows Server 2003 R2 source server

Complete the following tasks to prepare a source server that is running Windows Server 2003 or Windows Server 2003 R2 for Windows Server Migration Tools.

- Verify that the source server has sufficient disk space (at least 25 MB) to store the Windows Server Migration Tools deployment folder.
- Download and install Microsoft .NET Framework 2.0. Microsoft .NET Framework 2.0 is available for download from the Microsoft Web site.

- Download and install Windows PowerShell 1.0, or a later version. Windows PowerShell 1.0 is available for download from the Microsoft Web site.

Note
Windows PowerShell 2.0 and 3.0 are available in a graphically-oriented version, Windows PowerShell ISE. For more information about Windows PowerShell ISE, see Windows PowerShell 3.0 Integrated Scripting Environment (ISE).

Other computers in your enterprise

Because you might have to restart the server after you install Windows Server Migration Tools, notify users in advance that they might experience downtime while the server operating system loads. To minimize downtime, and reduce its effect on users in your enterprise, install Windows Server Migration Tools during off-peak hours.

Install Windows Server Migration Tools

This section describes how to install Windows Server Migration Tools on both source and destination servers. If both source and destination computers are running the same operating system on which Windows Server Migration Tools is available for installation (if both servers are running Windows Server 2012 R2, or both servers are running Windows Server 2012), install Windows Server Migration Tools on both computers by following installation steps in either Full installation option of Windows Server 2012 R2 or Windows Server 2012 or Server Core installation option of Windows Server 2012 R2 or Windows Server 2012.

If you plan to migrate roles, features, or other data from computers that are running older releases of Windows Server than your destination server—that is, Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003—you must complete the following additional tasks after you install Windows Server Migration Tools on destination servers.

1. Create a Windows Server Migration Tools deployment folder on destination servers. For more information, see Creating a deployment folder on destination computers.
2. Register Windows Server Migration Tools on source computers that are running older releases of Windows Server than your destination server; that is, Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003. For more information, see Registering Windows Server Migration Tools on source computers.

For more detailed information, see Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003 source computers.

Full installation option of Windows Server 2012 R2 or Windows Server 2012

To install Windows Server Migration Tools

1. Do one of the following to open a Windows PowerShell session with elevated user rights.

   Note
   If you are installing Windows Server Migration Tools from a remote server, you do not need to run Windows PowerShell with elevated user rights.

   - On the Windows desktop, right-click **Windows PowerShell** on the taskbar, and then click **Run as Administrator**.
   - On the Windows **Start** screen, right-click the Windows PowerShell tile, and then on the app bar, click **Run as Administrator**.
2. Type the following, and then press **Enter**. If you are installing the feature on the local server, omit the `ComputerName` parameter.

   ```
   Install-WindowsFeature Migration –ComputerName
   <computer_name>
   ```

   Note
   You can also install Windows Server Migration Tools on a full installation of Windows Server 2012 R2 or Windows Server 2012 by using the Add Roles and Features Wizard in Server Manager. For more information about how to use the Add Roles and Features Wizard, see Install or uninstall roles, role services, or features.

Server Core installation option of Windows Server 2012 R2 or Windows Server 2012

Windows PowerShell is installed by default on the Server Core installation option of Windows Server 2012 R2 and Windows Server 2012. By default, programs on the Server Core installation option run as Administrator, so there is no need to start Windows PowerShell with elevated user rights.

To install Windows Server Migration Tools on a Server Core installation of Windows Server 2012

1. Open a Windows PowerShell session by typing the following in the current command prompt session, and then press **Enter**.

   ```
   powershell.exe
   ```
2. In the Windows PowerShell session, install Windows Server Migration Tools by using the Windows PowerShell **Install-WindowsFeature** cmdlet for Server Manager. In the Windows PowerShell session, type the following, and then press **Enter**. Omit the `ComputerName` parameter if you are installing Windows Server Migration Tools on the local server.

   ```
   Install-WindowsFeature Migration –ComputerName
   <computer_name>
   ```

Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003 source computers

Complete the following two tasks to install Windows Server Migration Tools.

1. Create deployment folders for source computers by running the **smigdeploy.exe** tool (included with Windows Server Migration Tools) on your destination server. For more information, see Creating a deployment folder on destination computers.
2. Register Windows Server Migration Tools on source computers that are running Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003 by using **SmigDeploy.exe**. For more information, see Registering Windows Server Migration Tools on source computers.

Creating a deployment folder on destination computers

This procedure describes how to create the deployment folder on your destination server that is running Windows Server Migration Tools. After you create the deployment folder, copy it to the local drive of a migration source server that is running an older release of Windows Server; that is, Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, or Windows Server 2003.

To create a deployment folder on destination computers

1. If you have not already installed Windows Server Migration Tools on the destination server, see Install Windows Server Migration Tools in this topic.
2. Open a Command Prompt window with elevated user rights. On the Server Core installation option of Windows Server 2012 R2 or Windows Server 2012, an elevated command prompt is already opened by default. On the full installation option, type **cmd** on the **Start** screen, right-click the **Command Prompt** tile, and then click **Run as administrator**.
3. At the command prompt, change to the directory in which the **smigdeploy.exe** tool is stored. Type the following, and then press **Enter**.

   ```
   cd %Windir%\System32\ServerMigrationTools\
   ```

4. Do one of the following to create a Windows Server Migration Tools deployment folder.
   - To create a folder to copy to an x64-based computer that is running Windows Server 2012, where Windows Server 2012 R2 is running on the destination server, type the following, in which *deployment folder path* represents the path of the deployment folder on the source computer, and then press **Enter**.

     ```
     SmigDeploy.exe /package /architecture amd64 /os WS12 /path
     <deployment folder path>
     ```

   - To create a folder to copy to an x64-based computer that is running Windows Server 2008 R2, type the following, in which *deployment folder path* represents the path of the deployment folder on the source computer, and then press **Enter**.

     ```
     SmigDeploy.exe /package /architecture amd64 /os WS08R2 /path
     <deployment folder path>
     ```

- To create a folder to copy to an x64-based source computer that is running Windows Server 2008, type the following, in which *deployment folder path* represents the path of the deployment folder on the source computer, and then press **Enter**.

  ```
  SmigDeploy.exe /package /architecture amd64 /os WS08 /path <deployment folder path>
  ```

- To create a folder to copy to an x64-based source computer that is running Windows Server 2003, type the following, in which *deployment folder path* represents the path of the deployment folder on the source computer, and then press **Enter**.

  ```
  SmigDeploy.exe /package /architecture amd64 /os WS03 /path <deployment folder path>
  ```

- To create a folder to copy to an x86-based source computer that is running Windows Server 2008, type the following, in which *deployment folder path* represents the path of the deployment folder on the source computer, and then press **Enter**.

  ```
  SmigDeploy.exe /package /architecture X86 /os WS08 /path <deployment folder path>
  ```

- To create a folder to copy to an x86-based source computer that is running Windows Server 2003, type the following, in which *deployment folder path* represents the path of the deployment folder on the source computer, and then press **Enter**.

  ```
  SmigDeploy.exe /package /architecture X86 /os WS03 /path <deployment folder path>
  ```

Note

Each of these commands creates a deployment folder named in the format SMT_<*Operating System*>_<*Architecture*> and stores it in the specified deployment folder path.

You can also specify a network path as the path for the deployment folder. Verify that you have access to the network path before you create the deployment folder.

For more information about **SmigDeploy.exe**, at a command prompt, type **SmigDeploy.exe /?**, and then press **Enter**.

Registering Windows Server Migration Tools on source computers

Before you can run the Windows Server Migration Tools snap-in for the first time on a source server that is running an older release of Windows Server than your destination server, it must be registered with Windows PowerShell. Use **SmigDeploy.exe** to register the Windows Server Migration Tools snap-in on a migration source server that is running an older release of Windows Server than your destination server (that is, Windows Server 2012, Windows Server 2008 R2, Windows Server 2008 or Windows Server 2003).

Before you start the procedure in this section, verify the following.

- Microsoft .NET Framework 2.0 is installed on computers that are running Windows Server 2003.

- Windows PowerShell 1.0 or a later version is installed on source computers that are running either Windows Server 2008 or Windows Server 2003. (Windows PowerShell is already installed on computers that are running Windows Server 2008 R2 and Windows Server 2012.)

To register Windows Server Migration Tools

1. Copy the Windows Server Migration Tools deployment folder that was created by using the procedure in Creating a deployment folder on destination computers to a local drive on the source computer that is running an older release of Windows Server than your destination server. Be sure that the operating system architecture of the deployment folder matches that of the source computer to which you are copying the folder.

   For example, the **SMT_WS08_amd64** folder should only be copied to the local drive of an AMD64 source computer that is running Windows Server 2008.
2. On the source computer, open a Command Prompt window.
   - On computers that are running Windows Server 2003 or the Server Core installation option of Windows Server 2008 R2, you do not have to run a Command Prompt window with elevated user rights. Click **Start**, click **Run**, type **cmd**, and then click **OK**.
   - On computers that are running the full installation options of Windows Server 2012, Windows Server 2008 R2 or Windows Server 2008, you must open a Command Prompt window with elevated user rights. To do this, right-click the shortcut for Command Prompt, and then click **Run as Administrator**.
3. At the command prompt, change to the directory to which you copied the Windows Server Migration Tools deployment folder in step 1.

   Note

   You can register and run Windows Server Migration Tools cmdlets from a removable drive, CD-ROM, or DVD-ROM. However, to increase the reliability of registering the cmdlets, we recommend that you copy the deployment folder to a local drive of the source computer. You cannot register or run Windows Server Migration Tools cmdlets from a network location.
4. In the deployment folder directory, type the following command to register Windows Server Migration Tools cmdlets, and then press **Enter**.

   ```
   .\Smigdeploy.exe
   ```

Note

When registration is finished, a status message is displayed that indicates that the registration finished successfully, and a Windows PowerShell session opens. You can run Windows Server Migration Tools cmdlets in this Windows PowerShell session. If you close the Windows PowerShell session, see Windows Server 2003 or Windows Server 2008 source computers for information about how to access and use Windows Server Migration Tools cmdlets.

Use Windows Server Migration Tools

This section describes how to run Windows Server Migration Tools cmdlets.

Full installation option of Windows Server 2012 R2

Start Windows PowerShell and run Windows Server Migration Tools cmdlets by using either of the following procedures. These can apply to either source or destination servers.

To run Windows Server Migration Tools from the Start screen

- To open a Windows Server Migration Tools custom Windows PowerShell session, right-click the **Windows Server Migration Tools** tile, and then on the app bar, click **Run as administrator**.

To run Windows Server Migration Tools in a new Windows PowerShell session

1. Do one of the following to open a Windows PowerShell session with elevated user rights.
   - On the Windows desktop, right-click **Windows PowerShell** on the taskbar, and then click **Run as Administrator**.
   - On the Windows **Start** screen, right-click the Windows PowerShell tile, and then on the app bar, click **Run as Administrator**.
2. Load Windows Server Migration Tools into your Windows PowerShell session. To load Windows Server Migration Tools, type the following, and then press **Enter**.

```
Add-PSSnapin Microsoft.Windows.ServerManager.Migration
```

Server Core installation option of Windows Server 2012 R2

This procedure applies to either source or destination servers.

To run Windows Server Migration Tools cmdlets

1. Type **powershell** into a command prompt, and then press **Enter**.
2. Load Windows Server Migration Tools into your Windows PowerShell session. To load Windows Server Migration Tools, type the following, and then press **Enter**.

```
Add-PSSnapin Microsoft.Windows.ServerManager.Migration
```

Full installation option of Windows Server 2012

Start Windows PowerShell and run Windows Server Migration Tools cmdlets by using either of the following procedures. These can apply to either source or destination servers.

To run Windows Server Migration Tools from the Start screen

- To open a Windows Server Migration Tools custom Windows PowerShell session, right-click the **Windows Server Migration Tools** tile, and then on the app bar, click **Run as administrator**.

To run Windows Server Migration Tools in a new Windows PowerShell session

1. Do one of the following to open a Windows PowerShell session with elevated user rights.
   - On the Windows desktop, right-click **Windows PowerShell** on the taskbar, and then click **Run as Administrator**.
   - On the Windows **Start** screen, right-click the Windows PowerShell tile, and then on the app bar, click **Run as Administrator**.
2. Load Windows Server Migration Tools into your Windows PowerShell session. To load Windows Server Migration Tools, type the following, and then press **Enter**.

```
Add-PSSnapin Microsoft.Windows.ServerManager.Migration
```

Server Core installation option of Windows Server 2012

This procedure applies to either source or destination servers.

To run Windows Server Migration Tools cmdlets

1. Type **powershell** into a command prompt, and then press **Enter**.
2. Load Windows Server Migration Tools into your Windows PowerShell session. To load Windows Server Migration Tools, type the following, and then press **Enter**.

```
Add-PSSnapin Microsoft.Windows.ServerManager.Migration
```

Source computer running full installation option of Windows Server 2008 R2

If you close the Windows PowerShell session that is opened automatically when **SmigDeploy.exe** finishes registering the Windows Server Migration Tools cmdlets, you can run Windows Server Migration Tools cmdlets by using any of the following procedures.

To run Windows Server Migration Tools from the Start menu

- To open a Windows Server Migration Tools custom Windows PowerShell session, click **Start**, point to **Administrative Tools**, open the **Windows Server Migration Tools** folder, right-click **Windows Server Migration Tools**, and then click **Run as administrator**.

To run Windows Server Migration Tools in a new Windows PowerShell session

1. Open a Windows PowerShell session with elevated user rights. To do this, click **Start**, click **All Programs**, click **Accessories**, click **Windows PowerShell**, right-click the Windows PowerShell shortcut, and then click **Run as administrator**.
2. Load Windows Server Migration Tools into your Windows PowerShell session. To load Windows Server Migration Tools, type the following, and then press **Enter**.

```
Add-PSSnapin Microsoft.Windows.ServerManager.Migration
```

Source computer running Server Core installation option of Windows Server 2008 R2

Start Windows PowerShell and use Windows Server Migration Tools cmdlets by using any of the following procedures.

To open Windows PowerShell together with Windows Server Migration Tools

- At a command prompt on a computer that is running the Server Core installation option of Windows Server 2008 R2, type the following, and then press **Enter**.

```
powershell.exe -PSConsoleFile
%SystemRoot%\system32\ServerMigrationTools\ServerMigration.ps
c1
```

To open Windows PowerShell and load Windows Server Migration Tools separately

1. At a command prompt, type the following, and then press **Enter**.

```
powershell
```

2. Load Windows Server Migration Tools into the Windows PowerShell session. To load Windows Server Migration Tools, type the following, and then press **Enter**.

```
Add-PSSnapin Microsoft.Windows.ServerManager.Migration
```

Windows Server 2003 or Windows Server 2008 source computers

If you close the Windows PowerShell session that is opened automatically when **SmigDeploy.exe** finishes registering the Windows Server Migration Tools cmdlets, you can run Windows Server Migration Tools cmdlets by using any of the following procedures.

To open Windows Server Migration Tools from the Start menu

- Do one of the following:
  - On computers that are running Windows Server 2003, click **Start**, point to **Administrative Tools**, open the **Windows Server Migration Tools** folder, and then

click **Windows Server Migration Tools**.

- On computers that are running Windows Server 2008, click **Start**, point to **Administrative Tools**, open the **Windows Server Migration Tools** folder, right-click **Windows Server Migration Tools**, and then click **Run as administrator**.

To open Windows PowerShell and load Windows Server Migration Tools separately

1. Do one of the following:
   - On computers that are running Windows Server 2003, open a Windows PowerShell session by clicking **Start**, clicking **All Programs**, opening the **Windows PowerShell** folder, and clicking the **Windows PowerShell** shortcut.
   - On computers that are running Windows Server 2008, open a Windows PowerShell session with elevated user rights. To do this, click **Start**, click **All Programs**, open the **Windows PowerShell** folder, right-click the **Windows PowerShell** shortcut, and then click **Run as administrator**.
2. In the Windows PowerShell session, type the following to load the Windows Server Migration Tools snap-in, and then press **Enter**.

```
Add-PSSnapin Microsoft.Windows.ServerManager.Migration
```

To open Windows PowerShell together with Windows Server Migration Tools from a Command Prompt window

1. Do one of the following.
   - On computers that are running Windows Server 2003, open a Command Prompt window by clicking **Start**, clicking **Run**, typing **cmd**, and then pressing **Enter**.
   - On computers that are running Windows Server 2008, open a Command Prompt window with elevated user rights. To do this, click **Start**, click **All Programs**, click **Accessories**, right-click the **Command Prompt** shortcut, and then click **Run as administrator**.
2. At the command prompt, change directories to the location of the Windows Server Migration Tools deployment folder.
3. In the deployment directory, type the following to open a Windows PowerShell session with preloaded Windows Server Migration Tools cmdlets, and then press **Enter**.

```
PowerShell.exe -PSConsoleFile ServerMigration.psc1
```

Additional resources and next steps for using Windows Server Migration Tools

For more information about Windows Server Migration Tools and Windows PowerShell, see the following resources.

- For detailed, step-by-step information about how to migrate specific roles or data, see Migrate Roles and Features to Windows Server.
- In a Windows PowerShell session, type the following, and then press **Enter** to view detailed information about how to use a specific Windows Server Migration Tools cmdlet.

```
Get-Help <cmdlet_name> -full
```

- See the Windows PowerShell page on the Microsoft Web site.

Remove Windows Server Migration Tools

Follow the procedures in this section to remove Windows Server Migration Tools from computers.

Full installation option of Windows Server 2012 R2 or Windows Server 2012

You can use either Server Manager deployment cmdlets, or the Add Roles and Features Wizard in Server Manager to remove Windows Server Migration Tools. If Windows Server 2012 was a source computer for a migration to a server running Windows Server 2012 R2, unregister Windows Server Migration Tools on the source computer instead of uninstalling Windows Server Migration Tools. For more information, see Source computers running full and Server Core installation options of Windows Server 2012.

To uninstall Windows Server Migration Tools from the full installation option

1. Do one of the following to open a Windows PowerShell session with elevated user rights.

   Note
   If you are uninstalling Windows Server Migration Tools from a remote server, you do not need to run Windows PowerShell with elevated user rights.

   - On the Windows desktop, right-click **Windows PowerShell** on the taskbar, and then click **Run as Administrator**.
   - On the Windows **Start** screen, right-click the Windows PowerShell tile, and then on the app bar, click **Run as Administrator**.
2. Type the following, and then press **Enter**. If you are uninstalling the feature from the local server, omit the `ComputerName` parameter.

   ```
   Uninstall-WindowsFeature Migration –ComputerName
   <computer_name>
   ```

   Note
   You can also uninstall Windows Server Migration Tools from a full installation of Windows Server 2012 R2 or Windows Server 2012 by using the Add Roles and Features Wizard in Server Manager. For more information about how to use the Add Roles and Features Wizard, see Install or uninstall roles, role services, or features.

Server Core installation option of Windows Server 2012 R2 or Windows Server 2012

Windows PowerShell is installed by default on the Server Core installation option of Windows Server 2012 R2 or Windows Server 2012. By default, programs on the Server Core installation option run as Administrator, so there is no need to start Windows PowerShell with elevated user rights.

To uninstall Windows Server Migration Tools from the Server Core installation option

1. Open a Windows PowerShell session by typing the following in the current command prompt session, and then press **Enter**.

   ```
   powershell.exe
   ```

2. In the Windows PowerShell session, uninstall Windows Server Migration Tools by using the Windows PowerShell **Uninstall-WindowsFeature** cmdlet for Server Manager. In the Windows PowerShell session, type the following, and then press **Enter**. Omit the `ComputerName` parameter if you are uninstalling Windows Server Migration Tools from the local server.

   ```
   Uninstall-WindowsFeature Migration –ComputerName
   <computer_name>
   ```

Source computers running full and Server Core installation options of Windows Server 2012

To remove Windows Server Migration Tools from a source computer that is running Windows Server 2012, and on which Windows Server Migration Tools was registered for migrating to a destination server running Windows Server 2012 R2, you must first reverse the registration of Windows Server Migration Tools cmdlets, and then remove the deployment folder.

To remove Windows Server Migration Tools from Windows Server 2012

1. Do one of the following.
   - On computers that are running the full installation option of Windows Server 2012, open a Command Prompt window with elevated user rights. To do this, on the **Start** screen, type **cmd**. Right-click the **Command Prompt** tile, and then click **Run as Administrator**.
   - On computers that are running the Server Core installation option of Windows Server 2012, select the Command Prompt window to bring it in focus. You do not need to open a command prompt with elevated user rights on Server Core installations.
2. Change directories to the location of the Windows Server Migration Tools deployment folder.
3. Type the following to reverse the registration of Windows Server Migration Tools cmdlets, and then press **Enter**.

   ```
   SmigDeploy.exe /unregister
   ```

4. When **SmigDeploy.exe** has finished, delete the Windows Server Migration Tools deployment folder and its contents.

Source computers running full and Server Core installation options of Windows Server 2008 R2

To remove Windows Server Migration Tools, you must first reverse the registration of Windows Server Migration Tools cmdlets, and then remove the deployment folder.

To remove Windows Server Migration Tools from Windows Server 2008 R2

1. Do one of the following.
   - On computers that are running the full installation option of Windows Server 2008 R2, open a Command Prompt window with elevated user rights. To do this, click **Start**, click **All Programs**, click **Accessories**, right-click **Command Prompt**, and then click **Run as administrator**.
   - On computers that are running the Server Core installation option of Windows Server 2008 R2, select the Command Prompt window to bring it in focus. You do not need to open a command prompt with elevated user rights on Server Core installations.
2. Change directories to the location of the Windows Server Migration Tools deployment folder.
3. Type the following to reverse the registration of Windows Server Migration Tools cmdlets, and then press **Enter**.

   ```
   SmigDeploy.exe /unregister
   ```
4. When **SmigDeploy.exe** has finished, delete the Windows Server Migration Tools deployment folder and its contents.

Windows Server 2003 or Windows Server 2008 source computers

To remove Windows Server Migration Tools, you must first reverse the registration of Windows Server Migration Tools cmdlets, and then remove the deployment folder.

To remove Windows Server Migration Tools from Windows Server 2003 or Windows Server 2008

1. Do one of the following.
   - On computers that are running Windows Server 2003, open a Command Prompt window by clicking **Start**, clicking **Run**, typing **cmd**, and then pressing **Enter**.
   - On computers that are running Windows Server 2008, open a Command Prompt window with elevated user rights. To do this, click **Start**, click **All Programs**, click **Accessories**, right-click **Command Prompt**, and then click **Run as administrator**.
2. At a command prompt, change directories to the location of the Windows Server

Migration Tools deployment folder.

3. Type the following to reverse the registration of Windows Server Migration Tools cmdlets, and then press **Enter**.

```
SmigDeploy.exe /unregister
```

4. When **SmigDeploy.exe** has finished, delete the Windows Server Migration Tools deployment folder and its contents.

See Also

Windows Server Migration Portal

Windows PowerShell

Install or uninstall roles, role services, or features

Adding Server Roles and Features

Migrate Active Directory Federation Services Role Services to Windows Server 2012

About this guide

This guide provides instructions to migrate the following role services to Active Directory Federation Services (AD FS) that is installed with Windows Server 2012:

- AD FS 1.1 Windows token-based agent and AD FS 1.1 claims-aware agent installed with Windows Server 2008 or Windows Server 2008 R2
- AD FS 2.0 federation server and AD FS 2.0 federation server proxy installed on Windows Server 2008 or Windows Server 2008 R2

Target audience

- IT architects who are responsible for computer management and security throughout an organization
- IT operations engineers who are responsible for the day-to-day management and troubleshooting of networks, servers, client computers, operating systems, or applications
- IT operations managers who are accountable for network and server management

Supported migration scenarios

The migration instructions in this guide consist of the following tasks:

- Exporting the AD FS 2.0 configuration data from your server that is running Windows Server 2008 or Windows Server 2008 R2

- Performing an in-place upgrade of the operating system of this server from Windows Server 2008 or Windows Server 2008 R2 to Windows Server 2012
- Recreating the original AD FS configuration and restoring the remaining AD FS service settings on this server, which is now running the AD FS server role that is installed with Windows Server 2012.

This guide does not include instructions to migrate a server that is running multiple roles. If your server is running multiple roles, we recommend that you design a custom migration process specific to your server environment, based on the information provided in other role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server.

Supported operating systems

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 with Service Pack 2	Windows Server 2012 or Windows Server 2008 R2 (Server Core and full installation options)	x64-based
x86- or x64-based	Windows Server 2003 R2		
x86- or x64-based	Windows Server 2008, both full and Server Core installation options		
x64-based	Windows Server 2008 R2		
x64-based	Server Core installation option of Windows Server 2008 R2		
x64-based	Server Core and full installation options of Windows Server 2012		

Notes

- The versions of operating systems that are listed in the preceding table are the oldest combinations of operating systems and service packs that are supported.
- The Foundation, Standard, Enterprise, and Datacenter editions of the Windows Server operating system are supported as the source or the destination server.
- Migrations between physical operating systems and virtual operating systems are supported.

Supported AD FS role services and features

The following table describes the migration scenarios of the AD FS role services and their respective settings that are described in this guide.

From	To AD FS installed with Windows Server 2012
AD FS 1.0 federation server installed with Windows Server 2003 R2	Migration is not supported
AD FS 1.0 federation server proxy installed with Windows Server 2003 R2	Migration is not supported
AD FS 1.0 Windows token-based agent installed with Windows Server 2003 R2	Migration is not supported
AD FS 1.0 claims-aware agent installed with Windows Server 2003 R2)	Migration is not supported
AD FS 1.1 federation server installed with Windows Server 2008 or Windows Server 2008 R2	Migration is not supported
AD FS 1.1 federation server proxy installed with Windows Server 2008 or Windows Server 2008 R2	Migration is not supported
AD FS 1.1 Windows token-based agent installed with Windows Server 2008 or Windows Server 2008 R2	Migration on the same server is supported, but the migrated AD FS Windows token-based agent will function only with an AD FS 1.1 federation service installed with Windows Server 2008 or Windows Server 2008 R2. For more information, see: • Migrate the AD FS 1.1 Web Agents • Interoperating with AD FS 1.x
AD FS 1.1 claims-aware agent installed with Windows Server 2008 or Windows Server 2008 R2)	Migration on the same server is supported. The migrated AD FS 1.1 claims-aware web agent will function with the following: • AD FS 1.1 federation service installed with Windows Server 2008 or Windows Server 2008 R2 • AD FS 2.0 federation service installed on Windows Server 2008 or Windows Server 2008 R2 • AD FS federation service installed with Windows Server 2012

From	To AD FS installed with Windows Server 2012
	For more information, see: • Migrate the AD FS 1.1 Web Agents • Interoperating with AD FS 1.x
AD FS 2.0 federation server installed on Windows Server 2008 or Windows Server 2008 R2	Migration on the same server is supported. For more information, see: • Prepare to Migrate the AD FS 2.0 Federation Server • Migrate the AD FS 2.0 Federation Server
AD FS 2.0 federation server proxy installed on Windows Server 2008 or Windows Server 2008 R2	Migration on the same server is supported. For more information see: • Prepare to Migrate the AD FS 2.0 Federation Server Proxy • Migrate the AD FS 2.0 Federation Server Proxy

See Also

Prepare to Migrate the AD FS 2.0 Federation Server

Prepare to Migrate the AD FS 2.0 Federation Server Proxy

Migrate the AD FS 2.0 Federation Server

Migrate the AD FS 2.0 Federation Server Proxy

Migrate the AD FS 1.1 Web Agents

Prepare to Migrate the AD FS 2.0 Federation Server

This topic includes the following information:

- Prepare to migrate a stand-alone AD FS federation server or a single-node AD FS farm
- Prepare to migrate a WID farm
- Prepare to migrate a SQL Server farm

Prepare to migrate a stand-alone AD FS federation server or a single-node AD FS farm

To prepare to migrate (same server migration) a stand-alone AD FS 2.0 federation server or a single-node AD FS farm to Windows Server 2012, you must export and back up the AD FS configuration data from this server.

To export the AD FS configuration data, perform the following tasks:

- Step 1: Export service settings
- Step 2: - Export claims provider trusts
- Step 3: - Export relying party trusts
- Step 4: - Back up custom attribute stores
- Step 5: Back up webpage customizations

Step 1: Export service settings

To export service settings, perform the following procedure:

To export service settings

1. Record the certificate subject name and thumbprint value of the SSL certificate used by the federation service. To find the SSL certificate, open the Internet Information Services (IIS) management console, Select **Default Web Site** in the left pane, click **Bindings…** in the **Action** pane, find and select the https binding, click **Edit**, and then click **View**.

   Notes

   Optionally, you can also export the SSL certificate used by the federation service and its private key to a .pfx file. For more information, see Export the Private Key Portion of a Server Authentication Certificate.

   Exporting the SSL certificate is optional because this certificate is stored in the local computer Personal certificates store and is preserved in the operating system upgrade.

2. Record the configuration of the AD FS Service communications, token-decrypting and token-signing certificates. To view all the certificates that are used, open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to create a list of all certificates in use in a file `PSH:>Get-ADFSCertificate | Out-File ".\certificates.txt"`

   Notes

   Optionally, you can also export any token-signing, token-encryption, or service-communications certificates and keys that are not internally generated, in addition to all self-signed certificates. You can view all the certificates that are in use on your server by using Windows PowerShell. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows

PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell`. Then run the following command to view all certificates that are in use on your server `PSH:>Get-ADFSCertificate`. The output of this command includes StoreLocation and StoreName values that specify the store location of each certificate. You can then use the guidance in Export the Private Key Portion of a Server Authentication Certificate to export each certificate and its private key to a .pfx file.

Exporting these certificates is optional because all external certificates are preserved during the operating system upgrade.

3. Export AD FS 2.0 federation service properties, such as the federation service name, federation service display name, and federation server identifier to a file.

   To export federation service properties, open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to export federation service properties: `PSH:> Get-ADFSProperties | Out-File ".\properties.txt"`.

   The output file will contain the following important configuration values:

Federation Service Property name as reported by Get-ADFSProperties	Federation Service Property name in AD FS management console
HostName	Federation Service name
Identifier	Federation Service identifier
DisplayName	Federation Service display name

4. Back up the application configuration file. Among other settings, this file contains the policy database connection string.

   To back up the application configuration file, you must manually copy the `%programfiles%\Active Directory Federation Services 2.0\Microsoft.IdentityServer.Servicehost.exe.config` file to a secure location on a backup server.

   Notes

   Make note of the database connection string in this file, located immediately after "policystore connectionstring="). If the connection string specifies a SQL Server database, the value is needed when restoring the original AD FS configuration on the federation server.

   The following is an example of a WID connection string: `"Data Source=\\.\pipe\mssql$microsoft##ssee\sql\query;Initial Catalog=AdfsConfiguration;Integrated Security=True"`. The following is an example of a SQL Server connection string: `"Data Source=databasehostname;Integrated Security=True"`.

5. Record the identity of the AD FS 2.0 federation service account and the password of this

account.

To find the identity value, examine the **Log On As** column of **AD FS 2.0 Windows Service** in the **Services** console and manually record this value.

Note

For a stand-alone federation service, the built-in NETWORK SERVICE account is used. In this case, you do not need to have a password.

6. Export the list of enabled AD FS endpoints to a file.

   To do this, open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to export the list of enabled AD FS endpoints to a file: `PSH:> Get-ADFSEndpoint | Out-File ".\endpoints.txt"`.

7. Export any custom claim descriptions to a file.

   To do this, open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to export any custom claim descriptions to a file: `Get-ADFSClaimDescription | Out-File ".\claimtypes.txt"`.

Step 2: - Export claims provider trusts

To export the claims provider trusts, perform the following procedure:

To export claims provider trusts

1. You can use Windows PowerShell to export all claims provider trusts. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to export all claims provider trusts: `PSH:>Get-ADFSClaimsProviderTrust | Out-File ".\cptrusts.txt"`.

Step 3: - Export relying party trusts

To export relying party trusts, perform the following procedure:

To export relying party trusts

1. To export all relying party trusts, open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to export all relying party trusts:`PSH:>Get-ADFSRelyingPartyTrust | Out-File ".\rptrusts.txt"`.

Step 4: - Back up custom attribute stores

You can find information about custom attribute stores in use by AD FS by using Windows PowerShell. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to find information about the custom attribute stores: `PSH:>Get-ADFSAttributeStore`. The steps to upgrade or migrate custom attribute stores vary.

Step 5: Back up webpage customizations

To back up any webpage customizations, copy the AD FS webpages and the **web.config** file from the directory that is mapped to the virtual path **"/adfs/ls"** in IIS. By default, it is in the **%systemdrive%\inetpub\adfs\ls** directory.

Prepare to migrate a WID farm

To prepare to migrate AD FS 2.0 federation servers that belong to a Windows Internal Database (WID) farm to Windows Server 2012, you must export and back up the AD FS configuration data from these servers.

To export the AD FS configuration data, perform the following tasks:

- Step 1: - Export service settings
- Step 2: Back up custom attribute stores
- Step 3: Back up webpage customizations

Step 1: - Export service settings

To export service settings, perform the following procedure:

To export service settings

1. Record the certificate subject name and thumbprint value of the SSL certificate used by the federation service. To find the SSL certificate, open the Internet Information Services (IIS) management console, select **Default Web Site** in the left pane, click **Bindings…** in the **Action** pane, find and select the https binding, click **Edit**, then click **View**.

   Notes

   Optionally, you can also export the SSL certificate and its private key to a .pfx file. For more information, see Export the Private Key Portion of a Server Authentication Certificate.

   This step is optional because this certificate is stored in the local computer Personal certificates store and will be preserved in the operating system upgrade.

2. Export any token-signing, token-encryption, or service-communications certificates and keys that are not internally generated, in addition to self-signed certificates.

   You can view all the certificates that are in use on your server by using Windows

PowerShell. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to view all certificates that are in use on your server `PSH:>Get-ADFSCertificate`. The output of this command includes StoreLocation and StoreName values that specify the store location of each certificate. You can then use the guidance in Export the Private Key Portion of a Server Authentication Certificate to export each certificate and its private key to a .pfx file.

Note
This step is optional, because all external certificates are preserved during the operating system upgrade.

3. Record the identity of the AD FS 2.0 federation service account and the password of this account.

   To find the identity value, examine the **Log On As** column of **AD FS 2.0 Windows Service** in the **Services** console and manually record the value.

Step 2: Back up custom attribute stores

You can find information about custom attribute stores in use by AD FS by using Windows PowerShell. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to find information about the custom attribute stores: `PSH:>Get-ADFSAttributeStore`. The steps to upgrade or migrate custom attribute stores vary.

Step 3: Back up webpage customizations

To back up any webpage customizations, copy the AD FS webpages and the **web.config** file from the directory that is mapped to the virtual path **"/adfs/ls"** in IIS. By default, it is in the **%systemdrive%\inetpub\adfs\ls** directory.

Prepare to migrate a SQL Server farm

To prepare to migrate AD FS 2.0 federation servers that belong to a SQL Server farm to Windows Server 2012, you must export and back up the AD FS configuration data from these servers.

To export the AD FS configuration data, perform the following tasks:

- Step 1: Export service settings
- Step 2: Back up custom attribute stores
- Step 3: Back up webpage customizations

Step 1: Export service settings

To export service settings, perform the following procedure:

To export service settings

1. Record the certificate subject name and thumbprint value of the SSL certificate used by the federation service. To find the SSL certificate, open the Internet Information Services (IIS) management console, select **Default Web Site** in the left pane, click **Bindings…** in the **Action** pane, find and select the https binding, click **Edit**, and then click **View**.

   Notes

   Optionally, you can also export the SSL) certificate and its private key to a .pfx file. For more information, see Export the Private Key Portion of a Server Authentication Certificate.

   This step is optional because this certificate is stored in the local computer Personal certificates store and will be preserved in the operating system upgrade.

2. Export any other token-signing, token-encryption, or service-communications certificates and keys that are not internally generated by AD FS.

   You can view all certificates that are in use by AD FS on your server by using Windows PowerShell. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to view all certificates that are in use on your server `PSH:>Get-ADFSCertificate`. The output of this command includes StoreLocation and StoreName values that specify the store location of each certificate.

   Note

   Optionally, you can then use the guidance in Export the Private Key Portion of a Server Authentication Certificate to export each certificate and its private key to a .pfx file. This step is optional, because all external certificates are preserved during the operating system upgrade.

3. Back up the application configuration file. Among other settings, this file contains the policy database connection string.

   To back up the application configuration file, you must manually copy the `%programfiles%\Active Directory Federation Services 2.0\Microsoft.IdentityServer.Servicehost.exe.config` file to a secure location on a backup server.

   Note

   Record the SQL Server connection string after “policystore connectionstring=” in the following file: `%programfiles%\Active Directory Federation Services 2.0\Microsoft.IdentityServer.Servicehost.exe.config`. You need this string when you restore the original AD FS configuration on the federation server.

4. Record the identity of the AD FS 2.0 federation service account and the password of this account.

To find the identity value, examine the **Log On As** column of **AD FS 2.0 Windows Service** in the **Services** console and manually record the value.

Step 2: Back up custom attribute stores

You can find information about custom attribute stores in use by AD FS by using Windows PowerShell. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to find information about the custom attribute stores: `PSH:>Get-ADFSAttributeStore`. The steps to upgrade or migrate custom attribute stores vary.

Step 3: Back up webpage customizations

To back up any webpage customizations, copy the AD FS webpages and the **web.config** file from the directory that is mapped to the virtual path **"/adfs/ls"** in IIS. By default, it is in the **%systemdrive%\inetpub\adfs\ls** directory.

See Also

Migrate Active Directory Federation Services Role Services to Windows Server 2012

Prepare to Migrate the AD FS 2.0 Federation Server Proxy

To prepare to migrate an AD FS 2.0 federation server proxy to Windows Server 2012, you must export and back up the AD FS configuration data from this server proxy. The steps in this topic apply to a scenario with one proxy federation server or multiple proxy federation servers.

To export the AD FS configuration data, perform the following tasks:

- Step 1: Export proxy service settings
- Step 2: Back up webpage customizations

Step 1: Export proxy service settings

To export federation server proxy service settings, perform the following procedure:

To export proxy service settings

1. Export the Secure Sockets Layer (SSL) certificate and its private key to a .pfx file. For more information, see Export the Private Key Portion of a Server Authentication

Certificate.

Note
This step is optional because this certificate is preserved during the operating system upgrade.

2. Export AD FS 2.0 federation proxy properties to a file. You can do that by using Windows PowerShell.

   Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell".` Then run the following command to export federation proxy properties to a file: `PSH:> Get-ADFSProxyProperties | out-file ".\proxyproperties.txt".`

3. Ensure you know the credentials of an account that is either an administrator of the AD FS federation server or the service account under which the AD FS federation service runs. This information is required for the proxy trust setup.

Completing this step results in gathering the following information that is required to configure your AD FS federation server proxy:

- AD FS federation service name
- Name of the domain account that is required for the proxy trust setup
- The address and the port of the HTTP proxy (if there is an HTTP proxy between the AD FS federation server proxy and the AD FS federation servers)

Step 2: Back up webpage customizations

To back up webpage customizations, copy the AD FS proxy webpages and the **web.config** file from the directory that is mapped to the virtual path **"/adfs/ls"** in IIS. By default, it is in the **%systemdrive%\inetpub\adfs\ls** directory.

See Also

Migrate Active Directory Federation Services Role Services to Windows Server 2012

Migrate the AD FS 2.0 Federation Server

This topic provides instructions for the following migration scenarios:

- Migrate a stand-alone AD FS federation server or a single-node AD FS farm
- Migrate a WID farm
- Migrate a SQL Server farm

Migrate a stand-alone AD FS federation server or a single-node AD FS farm

To migrate a stand-alone AD FS federation server or a single-node AD FS farm to Windows Server 2012, perform the following procedure:

1. Review and perform the procedures in the “Prepare to migrate a stand-alone AD FS federation server or a single-node AD FS farm” section of Prepare to Migrate the AD FS 2.0 Federation Server.
2. Perform an in-place upgrade of the operating system on your server from Windows Server 2008 R2 or Windows Server 2008 to Windows Server 2012. For more information, see Installing Windows Server 2012.

   Important
   As the result of the operating system upgrade, the AD FS configuration on this server is lost and the AD FS 2.0 server role is removed. The Windows Server 2012 AD FS server role is installed instead, but it is not configured. You must manually create the original AD FS configuration and restore the remaining AD FS settings to complete the federation server migration.
3. Create the original AD FS configuration. You can create the original AD FS configuration by using either of the following methods:
   - Use the **AD FS Federation Server Configuration Wizard** to create a new federation server. For more information, see Create the First Federation Server in a Federation Server Farm.

     As you go through the wizard, use the information you gathered while preparing to migrate your AD FS federation server as follows:

Federation Server Configuration Wizard input option	Use the following value
SSL Certificate on the **Specify the Federation Service Name** page	Select the SSL certificate whose subject name and thumbprint you recorded while preparing for the AD FS federation server migration.
Service account and **Password** on the **Specify a Service Account** page	Enter the service account information that you recorded while preparing for the AD FS federation server migration. **Note** If you select stand-alone federation server on the second page of the

	wizard, NETWORK SERVICE is used automatically as the service account.

Important
You can employ this method only if you are using Windows Internal Database (WID) to store the AD FS configuration database for your stand-alone federation server or a single-node AD FS farm.

If you are using SQL Server to store the AD FS configuration database for your single-node AD FS farm, you must use Windows PowerShell to create the original AD FS configuration on your federation server.

- Use Windows PowerShell

  Important
  You must use Windows PowerShell if you are using SQL Server to store the AD FS configuration database for your stand-alone federation server or a single-node AD FS farm.

  The following is an example of how to use Windows PowerShell to create the original AD FS configuration on a federation server in a single-node SQL Server farm. Open the Windows PowerShell module and run the following command: `$fscredential = Get-Credential`. Enter the name and the password of the service account that you recorded while preparing your SQL server farm for migration. Then run the following command: `C:\PS> Add-AdfsFarmNode -ServiceAccountCredential $fscredential -SQLConnectionString "Data Source=<Data Source>;Integrated Security=True"` where `Data Source` is the data source value in the policy store connection string value in the following file: `%programfiles%\Active Directory Federation Services 2.0\Microsoft.IdentityServer.Servicehost.exe.config`.

4. Restore the remaining AD FS service settings and trust relationships. This is a manual step during which you can use the files that you exported and the values that you collected while preparing for the AD FS migration. For detailed instructions, see Restoring the Remaining AD FS Farm Configuration.

   Note
   This step is only required if you are migrating a stand-alone federation server or a single node WID farm. If the federation server uses a SQL Server database as the configuration store, the service settings and trust relationships are preserved in the database.

5. Update your AD FS webpages. This is a manual step. If you backed up your customized AD FS webpages while preparing for the migration, use your backup data to overwrite the default AD FS webpages that were created by default in the

%systemdrive%\inetpub\adfs\ls directory as a result of the AD FS configuration on Windows Server 2012.

6. Restore any remaining AD FS customizations, such as custom attribute stores.

Migrate a WID farm

To migrate a Windows Internal Database (WID) farm to Windows Server 2012, perform the following procedure:

1. For every node (server) in the WID farm, review and perform the procedures in the "Prepare to migrate a WID farm" section of Prepare to Migrate the AD FS 2.0 Federation Server.
2. Remove any non-primary nodes from the load balancer.
3. Upgrade of the operating system on this server from Windows Server 2008 R2 or Windows Server 2008 to Windows Server 2012. For more information, see Installing Windows Server 2012.

   
   Important
   As the result of the operating system upgrade, the AD FS configuration on this server is lost and the AD FS 2.0 server role is removed. The Windows Server 2012 AD FS server role is installed instead, but it is not configured. You must create the original AD FS configuration and restore the remaining AD FS settings to complete the federation server migration.
4. Create the original AD FS configuration on this server.

   You can create the original AD FS configuration by using the **AD FS Federation Server Configuration Wizard** to add a federation server to a WID farm. For more information, see Add a Federation Server to a Federation Server Farm.

   
   Notes
   - When you reach the **Specify the Primary Federation Server and a Service Account** page in the **AD FS Federation Server Configuration Wizard**, enter the name of the primary federation server of the WID farm and be sure to enter the service account information that you recorded while preparing for the AD FS migration. For more information, see the "Prepare to migrate a WID farm" section in Prepare to Migrate the AD FS 2.0 Federation Server.
   - When you reach the **Specify the Federation Service Name** page, be sure to select the same SSL certificate you recorded in the "Prepare to migrate a WID farm" section in Prepare to Migrate the AD FS 2.0 Federation Server.

5. Update your AD FS webpages on this server. If you backed up your customized AD FS webpages while preparing for the migration, you need to use your backup data to overwrite the default AD FS webpages that were created by default in the

%systemdrive%\inetpub\adfs\ls directory as a result of the AD FS configuration on Windows Server 2012.

6. Add the server that you just upgraded to Windows Server 2012 to the load balancer.
7. Repeat steps 1 through 6 for the remaining secondary servers in your WID farm.
8. Promote one of the upgraded secondary servers to be the primary server in your WID farm. To do this, open Windows PowerShell and run the following command: `PSH:> Set-AdfsSyncProperties –Role PrimaryComputer`.
9. Remove the original primary server of your WID farm from the load balancer.
10. Demote the original primary server in your WID farm to be a secondary server by using Windows PowerShell. Open Windows PowerShell and run the following command to add the AD FS cmdlets to your Windows PowerShell session: `PSH:>add-pssnapin "Microsoft.adfs.powershell"`. Then run the following command to demote the original primary server to be a secondary server: `PSH:> Set-AdfsSyncProperties – Role SecondaryComputer –PrimaryComputerName <FQDN of the Primary Federation Server>`.
11. Upgrade of the operating system on this last node (server) in your WID farm from Windows Server 2008 R2 or Windows Server 2008 to Windows Server 2012. For more information, see Installing Windows Server 2012.

    Important
    As the result of upgrading the operating system, the AD FS configuration on this server is lost and the AD FS 2.0 server role is removed. The Windows Server 2012 AD FS server role is installed instead, but it is not configured. You must manually create the original AD FS configuration and restore the remaining AD FS settings to complete the federation server migration.

12. Create the original AD FS configuration on this last node (server) in your WID farm.

    You can create the original AD FS configuration by using the **AD FS Federation Server Configuration Wizard** to add a federation server to a WID farm. For more information, see Add a Federation Server to a Federation Server Farm.

    Notes
    - When you reach the **Specify the Primary Federation server and a Service Account** page in the **AD FS Federation Server Configuration Wizard**, enter the service account information that you recorded while preparing for the AD FS migration. For more information, see the “Prepare to migrate a WID farm” section in Prepare to Migrate the AD FS 2.0 Federation Server.
    - When you reach the **Specify the Federation Service Name** page, be sure to select the same SSL certificate you recorded in the “Prepare to migrate a WID farm” section in Prepare to Migrate the AD FS 2.0 Federation Server.

13. Update your AD FS webpages on this last server in your WID farm. If you backed up your customized AD FS webpages while preparing for the migration, use your backup data to overwrite the default AD FS webpages that were created by default in the **%systemdrive%\inetpub\adfs\ls** directory as a result of the AD FS configuration on Windows Server 2012.
14. Add this last server of your WID farm that you just upgraded to Windows Server 2012 to

the load balancer.

15. Restore any remaining AD FS customizations, such as custom attribute stores.

Migrate a SQL Server farm

To migrate a SQL Server farm to Windows Server 2012, perform the following procedure:

1. For each server in your SQL Server farm, review and perform the procedures in the “Prepare to migrate a SQL Server farm” section of Prepare to Migrate the AD FS 2.0 Federation Server.
2. Remove any server in your SQL Server farm from the load balancer.
3. Upgrade the operating system on this server in your SQL Server farm from Windows Server 2008 R2 or Windows Server 2008 to Windows Server 2012. For more information, see Installing Windows Server 2012.

   Important
   As the result of the operating system upgrade, the AD FS configuration on this server is lost and the AD FS 2.0 server role is removed. The Windows Server 2012 AD FS server role is installed instead, but it is not configured. You must manually create the original AD FS configuration and restore the remaining AD FS settings to complete the federation server migration.
4. Create the original AD FS configuration on this server in your SQL Server farm by using AD FS Windows PowerShell cmdlets to add a server to an existing farm.

   Important
   You must use Windows PowerShell to create the original AD FS configuration if you are using SQL Server to store your AD FS configuration database.

   a. Open Windows PowerShell and run the following command: `$fscredential = Get-Credential`.
   b. Enter the name and the password of the service account that you recorded while preparing your SQL Server farm for migration.
   c. Run the following command: `Add-AdfsFarmNode -ServiceAccountCredential $fscredential -SQLConnectionString "Data Source=<Data Source>;Integrated Security=True"`, where `Data Source` is the data source value in the policy store connection string value in the following file: `%programfiles%\Active Directory Federation Services 2.0\Microsoft.IdentityServer.Servicehost.exe.config`.
5. Add the server that you just upgraded to Windows Server 2012 to the load balancer.
6. Repeat steps 2 through 6 for the remaining nodes in your SQL Server farm.
7. When all of the servers in your SQL Server farm are upgraded to Windows Server 2012, restore any remaining AD FS customizations, such as custom attribute stores.

Restoring the Remaining AD FS Farm Configuration

- Restore the following AD FS service settings to a single node WID farm or stand-alone federation service as follows:
  - In the AD FS management console, select **Service** and click **Edit Federation Service…**. Verify the federation service settings by checking each of the values against the values you exported into the properties.txt file while preparing for the migration:

Federation Service Property name as reported by Get-ADFSProperties	Federation Service Property name in AD FS Management console
DisplayName	Federation Service display name
HostName	Federation Service name
Identifier	Federation Service identifier

  - In the AD FS management console, select **Certificates**. Verify the service communications, token-decrypting, and token-signing certificates by checking each against the values you exported into the certificates.txt file while preparing for the migration.

    To change the token-decrypting or token-signing certificates from the default self-signed certificates to external certificates, you must first disable the automatic certificate rollover feature that is enabled by default. To do this, you can use the following Windows PowerShell command: `PSH: Set-ADFSProperties –AutoCertificateRollover $false`.
  - In the AD FS Management console, select **Endpoints**. Check the enabled AD FS endpoints against the list of enabled AD FS endpoints that you exported to a file while preparing for the AD FS migration.
  - In the AD FS Management console, select **Claim Descriptions**. Check the list of AD FS claim descriptions against the list of claim descriptions that you exported to a file while preparing for the AD FS migration. Add any custom claim descriptions included in your file but not included in the default list in AD FS. Note that Claim identifier in the management console maps to the ClaimType in the file. For more information on adding claim descriptions, see Add a Claim Description. For more information, see the “Step 1 - Export Service Settings” section in Prepare to Migrate the AD FS 2.0 Federation Server.
- In the AD FS Management console, select **Claims Provider Trusts**. You must recreate each Claims Provider trust manually using the **Add Claims Provider Trust Wizard**. Use the list of claims provider trusts that you exported and recorded while preparing for the AD FS migration. You can disregard the claims provider trust with Identifier “AD AUTHORITY” in the file because this is the “Active Directory” claims provider trust that is part of the default AD FS configuration. However, check for any custom claim rules you may have added to the Active Directory trust prior to the migration. For more information on creating claims provider trusts, see Create a Claims Provider Trust Using Federation Metadata or Create a Claims Provider Trust Manually.

- In the AD FS Management console, **select Relying Party Trusts**. You must recreate each Relying Party trust manually using the **Add Relying Party Trust Wizard**. Use the list of relying party trusts that you exported and recorded while preparing for the AD FS migration. For more information on creating relying party trusts, see Create a Relying Party Trust Using Federation Metadata or Create a Relying Party Trust Manually.

See Also

Migrate Active Directory Federation Services Role Services to Windows Server 2012

Migrate the AD FS 2.0 Federation Server Proxy

To migrate an AD FS 2.0 federation server proxy to Windows Server 2012, perform the following procedure:

1. For every federation server proxy that you plan to migrate to Windows Server 2012, review and perform the procedures in Prepare to Migrate the AD FS 2.0 Federation Server Proxy.
2. Remove a federation server proxy from the load balancer.
3. Perform an in-place upgrade of the operating system on this server from Windows Server 2008 R2 or Windows Server 2008 to Windows Server 2012. For more information, see Installing Windows Server 2012.

   
   Important
   As the result of the operating system upgrade, the AD FS proxy configuration on this server is lost and the AD FS 2.0 server role is removed. The Windows Server 2012 AD FS server role is installed instead, but it is not configured. You must manually create the original AD FS proxy configuration and restore the remaining AD FS proxy settings to complete the federation server proxy migration.
4. Create the original AD FS proxy configuration by using the **AD FS Federation Server Proxy Configuration Wizard**. For more information, see Configure a Computer for the Federation Server Proxy Role. As you execute the wizard, use the information you gathered in Prepare to Migrate the AD FS 2.0 Federation Server Proxy as follows:

Federation Server Proxy Wizard input option	Use the following value

Federation Service Name	Enter the BaseHostName value from proxyproperties.txt file
Use an HTTP proxy server when sending requests to this Federation Service check box	Check this box if your proxyproperties.txt file contains a value for the ForwardProxyUrl property
HTTP proxy server address	Enter the ForwardProxyUrl value from proxyproperties.txt file
Credential prompt	Enter the credentials of an account that is either an administrator of the AD FS federation server or the service account under which the AD FS federation service runs.

5. Update your AD FS webpages on this server. If you backed up your customized AD FS proxy webpages while preparing your federation server proxy for the migration, use your backup data to overwrite the default AD FS webpages that were created by default in the **%systemdrive%\inetpub\adfs\ls** directory as a result of the AD FS proxy configuration in Windows Server 2012.
6. Add this server back to the load balancer.
7. If you have other AD FS 2.0 federation server proxies to migrate, repeat steps 2 through 6 for the remaining federation server proxy computers.

See Also

Migrate Active Directory Federation Services Role Services to Windows Server 2012

Migrate the AD FS 1.1 Web Agents

To migrate the AD FS 1.1 Windows token-based agent or the AD FS 1.1 claims-aware agent that is installed with Windows Server 2008 R2 or Windows Server 2008 to Windows Server 2012, perform an in-place upgrade of the operating system of the computer that hosts either agent to Windows Server 2012. For more information, see Installing Windows Server 2012. No further configuration is required.

Important

The migrated AD FS 1.1 Windows token-based agent functions only with an AD FS 1.1 federation service that is installed with Windows Server 2008 R2 or Windows Server 2008. For more information, see Interoperating with AD FS 1.x.

The migrated AD FS 1.1 claims-aware web agent functions with the following:

- AD FS 1.1 federation service installed with Windows Server 2008 R2 or Windows Server 2008
- AD FS 2.0 federation service installed on Windows Server 2008 R2 or Windows Server 2008
- AD FS federation service installed with Windows Server 2012

  For more information, see Interoperating with AD FS 1.x.

See Also

Migrate Active Directory Federation Services Role Services to Windows Server 2012

Migrate File and Storage Services to Windows Server 2012

The File and Storage Services Migration Guide provides step-by-step instructions for how to migrate the File and Storage Services role, including data, shared folders, and operating system settings from a source server to a destination server that is running Windows Server 2012.

About this guide

Note

Your detailed feedback is very important, and helps us to make Windows Server Migration Guides as reliable, complete, and easy to use as possible. Please take a moment to rate this topic, and then add comments that support your rating. Click **Rate this topic** at the top of the page, and describe what you liked, did not like, or want to see in future versions of the topic. To submit additional suggestions about how to improve Migration guides or utilities, post on the Windows Server Migration forum.

Migration documentation and tools ease the migration of server role settings and data from an existing server to a destination server that is running Windows Server 2012. By using the tools that are described in this guide, you can simplify the migration process, reduce migration time, increase the accuracy of the migration process, and help to eliminate possible conflicts that might otherwise occur during the migration process. For more information about installing and using the migration tools on both source and destination servers, see Install, Use, and Remove Windows Server Migration Tools.

Specifically, this guide includes information about migrating the following:

- Information about the server's identity
- Local users and groups
- Data and shared folders
- Shadow Copies of Shared Folders
- Data Deduplication

- DFS Namespaces
- DFS Replication
- File Server Resource Manager (FSRM)
- Group Policy settings that are specific to server message block (SMB)
- Group Policy settings for Offline Files (also known as client-side caching or CSC)
- ISCSI Software Target

> **Note**
> ISCSI Software Target was previously an optional Windows Server and Windows Storage Server component download. Due to the amount of content, all iSCSI-specific migration information is located in File and Storage Services: Appendix C: Migrate iSCSI Software Target.

Target audience

This document is intended for information technology (IT) professionals and knowledge workers who are responsible for operating and deploying file servers in a managed environment.

What this guide does not provide

This guide does not provide information or support for the following migration scenarios:

- Clustering migration for clustered server configurations
- Migrating Roaming User Profiles (for additional information see **Upgrading or Migrating a Roaming User Profiles Environment to Windows 8.1 or Windows Server 2012 R2**).
- Upgrading roles on the same computer
- Migrating more than one server role
- Migrating data across subnets
- Migrating Network File System (NFS) shared folders
- Migrating file servers by using File Server Resource Manager
- Migrating encrypted files from Encrypting File System (EFS)
- Migrating file allocation tables (FAT) and FAT32 file systems
- Migrating hardware and software installation for storage resources

In addition to these unsupported scenarios, you should understand the following migration limitations:

- If the hard disk drive that contains your data is physically moved from the source server to the destination server, file and folder permissions for local users are not preserved.
- Reparse points, hard links, and mounted volumes are not migrated when data is copied, and they need to be migrated manually.
- To facilitate migrating file and shared folder permissions, you must migrate local users and groups as part of the migration procedure. However, not all user and group attributes are migrated.

For more information about the attributes of local users and groups that can be migrated, see the Local User and Group Migration Guide (http://go.microsoft.com/fwlink/?LinkId=258341) on the Microsoft Web site.

Supported migration scenarios

This guide provides instructions for migrating an existing server that is running File and Storage Services to a server that is running Windows Server 2012 R2 or Windows Server 2012. This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, it is recommended that you design a custom migration procedure for your server environment, based on the information that is provided in other server role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server 2012.

Caution

If your source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other server roles that are running on the source server to fail.

Supported migration scenarios include the following configurations or features:

- File server is joined to a domain
- File server is in a workgroup
- File server data and file shares are located in a storage area network (SAN) or other external storage location that preserves data and file share permissions (except data for local users and groups).
- File server data and file shares are located on the server disk (direct-attached storage) that is preserving data and files shares permissions.
- DFS Namespaces
- File Server Resource Manager
- Shadow Copies of Shared Folders

Important

The file migration portion of the Windows Server Migration Tools is designed for smaller data sets (under 100 GB of data). It copies files one at a time over HTTPS. For larger datasets, we recommend using the version of Robocopy.exe included with Windows Server 2012 R2 or Windows Server 2012.

Supported operating systems

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 with	Windows Server 2008 R2 or	x64-based

Source server processor	Source server operating system	Destination server operating system	Destination server processor
	Service Pack 2	Windows Server 2012, both full and Server Core installation options	
x86- or x64-based	Windows Server 2003 R2	Windows Server 2008 R2or Windows Server 2012, both full and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2008, full installation option	Windows Server 2008 R2or Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2008 R2or Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2008 R2	Windows Server 2008 R2or Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Server Core and full installation options of Windows Server 2012	Windows Server 2008 R2or Windows Server 2012, both full and Server Core installation options	x64-based

The versions of operating systems shown in the preceding table are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are supported.

Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported as either source or destination servers.

Migrations between physical operating systems and virtual operating systems are supported.

Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shares from a computer that is running Windows Server 2008 in the French system UI language to a computer that is running Windows Server 2012 in the German system UI language.

Note

The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Both x86- and x64-based migrations are supported for Windows Server 2003 and Windows Server 2008 R2. All editions of Windows Server 2008 R2 are x64-based.

File services migration overview

The following topics contain step-by-step information about how to migrate File and Storage Services from a computer that is running Windows Server 2003 or later to a computer that is running Windows Server 2012:

- File and Storage Services: Prepare to Migrate
- File and Storage Services: Migrate the File and Storage Services Role
- File and Storage Services: Verify the Migration
- File and Storage Services: Post-Migration Tasks

Impact of migration on other computers in the enterprise

The content in this section describes the impact to the computers in your enterprise during migration.

Impact of data migration by copying data and shared folders

- The performance of your source server can be affected during the data migration. This can result in slower access to files that are stored on the server.
- At the beginning of the second phase of the data migration, all open files are closed, which can lead to data loss.
- During the second phase of data migration, clients are unable to access the file server.

Impact of data migration by physically moving data drives

Clients cannot access the file server from the moment the storage device is disconnected from the source server until it is attached to the destination server.

Impact on DFS Namespaces

The DFS Namespaces are unavailable at several times during the migration process. You should plan your migration when you can take the namespace that is hosted on the source server offline.

Impact on DFS Replication

The impact of migration activity on other servers in the enterprise depends largely on the configuration of the replication topology. Typically, DFS Replication is configured in a hub and spoke replication topology with multiple branch office servers (spokes) replicating with a single hub server. Depending on which server in the replication topology is migrated, and how the data is migrated, the remaining servers in the enterprise can be affected. Client workstations that are accessing data that is contained in the replicated folder on the server can be affected during the migration process.

Client computers may be accessing data in the folder that is being replicated by using DFS Replication. The replicated folder is often exposed to client computers as an SMB shared folder.

For more information about the impact of the migration process on client computers, see Impact of data migration by copying data and shared folders earlier in this document.

Permissions required to complete migration

This section describes permissions that are required to perform the migration.

Permissions required for data and shared folder migration

For data and shared folder migration, local Administrator permissions are required on the source server and destination server.

Permissions required to complete migration on the destination server

This section describes permissions that are required to perform the migration on the destination server.

Permissions required to migrate DFS Namespaces

For a stand-alone namespace, the user must be a member of the local Administrators group on the destination server.

There are three permissions options for a domain-based namespace:

- Option 1: Membership in the Domain Admins group
- Option 2 (if there are more than one namespace server):
  - Permission to administer all namespaces that are hosted on the source server
  - Member of the local Administrators group on the destination server
- Option 3 (if there is a single namespace server):

- Permission to delete and create domain-based namespaces in the domain
- Member of the local Administrators group on the destination server

Permissions required to complete migration on the source server

This section describes permissions that are required to perform the migration on the source server.

Permissions required to migrate DFS Namespaces

For a stand-alone namespace, the user must have membership in the local Administrators group on the source server.

There are three permissions options for a domain-based namespace:

- Option 1: Membership in the Domain Admins group
- Option 2 (if there are more than one namespace servers):
  - Permission to administer the all namespaces that are hosted on the source server
  - Member of the local Administrators group on the source server
- Option 3 (if there is a single namespace server):
  - Permission to delete and create domain-based namespaces in the domain
  - Member of the local Administrators group on the destination server

Permissions required for DFS Replication

For DFS Replication, the user who starts the migration must be a member of the Domain Admins group or delegated permissions to the replication groups and replication members. This permission is required to remove the source server from replication groups to which it belongs. If the permissions to administer a replication group have been delegated to a particular user through the DFS Management snap-in, that user can use the DFS Management snap-in to perform tasks such as removing the source server from a replication group. The user must also be a member of the local Administrators group on the source server and the destination server.

See Also

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Prepare to Migrate

This guide provides you with instructions for migrating the File and Storage Services role to a server that is running Windows Server® 2008 R2.

Install migration tools

Windows Server Migration Tools in Windows Server® 2012 allows an administrator to migrate some server roles, features, operating system settings, shared folders, and other data from computers that are running certain editions of Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012 to computers that are running Windows Server 2012.

For complete installation, configuration, and removal instructions for Windows Server Migration Tools, see Install, Use, and Remove Windows Server Migration Tools.

Migration documentation and tools ease the process of migrating server role settings and data from an existing server that is running a Windows server operating system to another computer. For a complete list of supported operating systems, see Migrate File and Storage Services to Windows Server 2012.

By using these tools to migrate roles, you can simplify migration, reduce migration time, increase accuracy of the migration process, and help eliminate conflicts that could otherwise occur during the migration process.

Prepare for migration

The following list outlines the major steps for preparing to migrate the File and Storage Services role.

- Prepare the destination server
- Back up the source server
- Prepare the source server
- Prepare other computers in the enterprise

Important

Before you run the **Import-SmigServerSetting**, **Export-SmigServerSetting**, or **Get-SmigServerFeature** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with domain users or groups who are members of local groups on the source server.

Before you run the **Send-SmigServerData** or **Receive-SmigServerData** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with those domain users who have rights to files or shares that are being migrated.

Prepare the destination server

The following steps are necessary to prepare the destination server for migration.

Hardware requirements for the destination server

Verify that the data locations for the destination server have sufficient free space to migrate the data. Ensure that the destination server hard disk drives are the same size or larger than the source server hard disk drives.

Software requirements for the destination server

There are several software requirements that must be met to ensure a successful migration.

- Consult the migration matrix to determine if you can migrate the version of Windows Server that you are running on the source server to Windows Server 2012. For a complete list of supported operating systems, see Migrate File and Storage Services to Windows Server 2012.
- Before migration, install all critical updates and service packs on the source server that were released before Windows Server 2012. It is a recommended best practice that you install all current critical updates and service packs on the source server and the destination server.

Prepare for local user and group migration on the destination server

Verify that the destination server can resolve the names of domain users who are members of the local group during the import operation. If source server and destination server are in different domains, the destination server must be able to contact a global catalog server for the forest in which the source domain user accounts are located.

Prepare for File and Storage Services on destination server

1. Install Windows Server 2012 on the destination server.
2. Ensure that the time and date are set correctly on the destination server, and that they are in sync with the source server.
3. Determine the File Services role services that have been installed on the source server and then install the same File and Storage Services role services on the destination server.
4. Install Windows Server Migration Tools on the destination server.

   For more information about how to install Windows Server Migration Tools, see Install, Use, and Remove Windows Server Migration Tools.
5. Open UDP port 7000 and make sure that it is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to establish a data transfer connection.

   Note
   If you have changed the default behavior of Windows Firewall to block outbound traffic on computers that are running Windows Server 2012, you must explicitly allow outbound traffic on UDP port 7000.

6. Open TCP port 7000 and make sure that it is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to perform the data transfer.

   For more information about how to open UDP port 7000 and TCP port 7000, see File and Storage Services: Appendix A: Optional Procedures.

   For more information about how to determine if a port is in use, see the following article on the Microsoft Web site: How To Determine Which Program Uses or Blocks Specific Transmission Control Protocol Ports in Windows Server 2003 (http://go.microsoft.com/fwlink/?LinkId=149887).

7. Verify that the destination path has sufficient disk space to migrate the data. If NTFS or folder quota management (in File Server Resource Manager) is enabled on the destination server disk drive, verify that the quota limit allows for sufficient free disk space to migrate data. For more information about quota management in File Server Resource Manager, see one of the following.

   - Quota Management (http://go.microsoft.com/fwlink/?LinkId=154277) for Windows Server 2008 and Windows Server 2008 R2
   - Quota Management (http://go.microsoft.com/fwlink/?LinkId=154241) for Windows Server 2003 R2

   For more information about NTFS quota management, see one of the following.

   - Setting Disk Quotas (http://go.microsoft.com/fwlink/?LinkId=154243) for Windows Server 2008 and Windows Server 2008 R2
   - Enable disk quotas (http://go.microsoft.com/fwlink/?LinkId=154245) for Windows Server 2003 and Windows Server 2003 R2

Prepare File Server Resource Manager on destination server

If you are using File Classification Infrastructure plug-ins from a non-Microsoft vendor, you should register the non-Microsoft plug-ins on the destination server and refer to additional instructions for migration from the non-Microsoft plug-in vendor. You should register the plug-in after File Server Resource Manager (FSRM) has been installed and started on the destination server.

Use the same drive letters for the destination server volumes as for the source server. This is required, because FSRM migration requires the drive letter to remain the same.

Data and shared folder preparation on destination server

Do not allow users to access the destination server until migration is fully completed. This ensures data integrity and prevents failure when an open file on the destination server cannot be overwritten during migration.

Data integrity and security considerations on destination server

Server migration tools preserve file and folder permissions during data migration. When you are planning the migration, keep in mind that if the migrated files and folder inherit permissions from their parents, during migration it is the inheritance setting that is migrated, not the inherited permissions. Therefore it is important to make sure that the parent folders on the source server

and the destination server have the same permissions to maintain the permissions on migrated data that has inherited permissions.

For example:

1. Migrate folder c:\A\C from the source server to folder c:\B\D on the destination server.
2. Verify that on the source server, only Mary has access to folder c:\A and folder c:\A\C is specified to inherit permission from its parent.
3. Verify that on the destination server, only John has access to folder c:\B. After c:\A\C is migrated to c:\B\D, John will have access to folder D, but Mary will not.

If you use permissions inheritance for the migrated data, ensure that the parent folder for the migrated data on the destination server has the required permission set.

Prepare DFS Namespaces on destination server

The DFS Namespaces role service must be installed, and the DFS Namespace service must be running before migration. If the namespaces that you are migrating are domain-based, both source and destination servers must be in the same Active Directory® Domain Services (AD DS) domain. If the namespaces are stand-alone namespaces, AD DS membership does not matter.

Back up the source server

If DFS Namespaces are being migrated, back up the source server by using a full server backup or system state backup. If the DFS Namespaces are part of an AD DS domain, you need to back up the AD DS domain to save the Active Directory configuration information for DFS Namespaces.

For each domain-based DFS namespace, you should also back up the configuration information for the namespace. Repeat the following command for each namespace and save the output filename to a safe location:

```
DFSUtil.exe root export <\\<DomainName>\Namespace> <Filename>
```

Note

DFSUtil.exe is available on computers that are running Windows Server 2008, Windows Server 2008 R2, and Windows Server 2012. It is available to download for use on Windows Server 2003 and Windows Server 2003 R2 as part of the Windows Server 2003 Service Pack 1 32-bit Support Tools (http://go.microsoft.com/fwlink/?LinkId=147453).

Prepare the source server

The following sections describe how to prepare the source server for the migration.

Prepare all file services on source server

- Install Windows Server Migration Tools on the source server.

  For more information about how to install Windows Server Migration Tools, see Install, Use, and Remove Windows Server Migration Tools.

- Verify that the time and date are set correctly on the destination server and that they are synchronized with the source server.
- Open UDP port 7000 and make sure that is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to establish a data transfer connection.

  Note
  If you have changed the default behavior of Windows Firewall to block outbound traffic on computers that are running Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012, you must explicitly allow outbound traffic on UDP port 7000.
- Open TCP port 7000 and make sure that it is not in use by other applications. This port is used by **Send-SmigServerData** and **Receive-SmigServerData** to perform the data transfer.

For more information about how to open UDP port 7000 and TCP port 7000, see **File Services Migration: Appendix A: Optional Procedures**.

For more information about how to determine if a port is in use, see the following article on the Microsoft Web site: How To Determine Which Program Uses or Blocks Specific Transmission Control Protocol Ports in Windows Server 2003 (http://go.microsoft.com/fwlink/?LinkId=149887).

Data and shared folder preparation on the source server

To minimize downtime and reduce impact to users, plan your data migration to occur during off-peak hours. Use the `net share` command to list all shared folders on the source server.

You can use this list during the verification step to verify that all the required shared folders have migrated. Reparse points and hard links will not migrate when data is copied (versus a physical migration), and the reparse points need to be migrated manually. When you migrate hard links, a separate file is created on the destination server for each link. If your migration involves copying the data to the destination server, follow the instructions for how to detect the reparse points and hard links in File and Storage Services: Appendix A: Optional Procedures. Then you can remap and recreate them during migration, as instructed in the For copy data migration scenarios section.

Prepare DFS on the source server

DFS Namespaces role services must be installed, and DFS Namespace service must be running before migration.

For information about DFS Namespaces preparation, see Prepare DFS Namespaces on source server.

Prepare DFS Namespaces on source server

For domain-based namespaces with one namespace server, determine if you will add a temporary server to the namespace or if you will perform a manual inventory of the namespace permissions.

- Option 1 (recommended):

  Add a temporary server as a namespace server to each domain-based namespace on the source server when the source server is the only namespace server.
- Option 2:

  Inventory the permissions for managing each of the namespaces that are hosted on the source server when the source server is the only namespace server. This process can be completed by using the DFS Management MMC Snap-in.

Prepare other computers in the enterprise

Data and shared folder migration requires preparing other computers in the enterprise. Following are the steps that you should perform for copy data migration scenarios, and for physical data scenarios.

For copy data migration scenarios

- Notify the users that the server performance may be reduced during the first phase of data migration.
- Ask users to stop writing to the server before the second phase of data migration begins (to prevent possible data loss). We recommend that you prevent access to the file shares so that users don't ignore this advice. For example, you could temporarily set all file shares to be read-only by setting the share permissions to Everyone = Read Only.
- Notify users that they cannot access their files on the server when the second phase of the migration begins until the file server migration is fully completed.

For physical data migration scenarios

Notify the users that they cannot access the file server from the moment the storage is disconnected from the source server until the server migration is fully completed.

See Also

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Migrate the File and Storage Services Role

Migrate File Services

Perform the following tasks to migrate the File and Storage Services server role.

- Freeze administration configuration
- Export settings
- Migrate local users and groups to the destination server
- Migrate data
- Migrate the source server identity
- Configure DFS Replication on the destination server
- Import settings to the destination server

Freeze administration configuration

Administrators must stop all configuration changes to the File and Storage Services role services on the source server before starting migration. When the migration begins, you must not make any configuration changes to the source server other than those that are required for the migration (for example, no links can be added to a (DFS namespace after migration starts until the migration is verified successfully).

Install the Windows Server Migration Tools

Before you can use any of the following Windows PowerShell cmdlets for migration on the source server or destination server, ensure that the Windows Server Migration Tools is added. You can do this by using Server Manager or by using Windows PowerShell.

Do this step using Windows PowerShell

To install the Windows Server Migration Tools

1. Log on to the computer as a member of the local Administrators security group.
2. In Server Manager, click **Add roles and features**.
3. On the **Before you begin** page, click **Next**.
4. On the **Select installation type** page, select **the Role-base or feature-based installation** option, and then click **Next**.
5. On the **Select destination server** page, click **Next**.
6. On the **Select server roles** page, accept the default selections, and then click **Next**.
7. On the **Select features** page, click **Windows Server Migration Tools**, and then click **Next**.
8. On the **Confirm installation selections** page, click **Install**.

9. After the installation is complete, click **Close**.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Install-WindowsFeature Migration
```

The following is a list of Windows Server Migration Tools cmdlets:

- **Export-SmigServerSetting**
- **Import-SmigServerSetting**
- **Get-SmigServerFeature**
- **Send-SmigServerData**
- **Receive-SmigServerData**

For more information on how to work with the Windows Server Migration Tools see Install, Use, and Remove Windows Server Migration Tools.

Export settings

Export the following settings from the source server to the destination server. Settings include Server Message Block (SMB), Offline Files (also known as called client-side caching or CSC), DFS Namespaces, File Server Resource Manager (FSRM), and Shadow Copies of Shared Folders.

BranchCache for Network Files server key

The following procedure applies only if the source server is running Windows Server 2008 R2 or Windows Server 2012.

Notes

This procedure, which is used to migrate the seed value that is used by the BranchCache™ for Network Files component, enables data that was stored in branch office locations by using BranchCache to be used after the file server is migrated from the source server to the destination server.

For information about how to migrate a BranchCache host server, see the BranchCache Migration Guide (http://go.microsoft.com/fwlink/?LinkID=139091).

To migrate BranchCache for network files settings from the source server

1. In your Windows PowerShell session, collect data from the source server by running the **Export-SmigServerSetting** cmdlet as an member of the Administrators security group. This step runs the **Export-SmigServerSetting** cmdlet with all parameters from a single command line. The **Export-SmigServerSetting** cmdlet parameters can collect all source BranchCache feature data in a single file (Svrmig.mig), or you can run the **Export-SmigServerSetting** cmdlet multiple times by using one or more parameters to collect

and store data in multiple Svrmig.mig files.

For more information, see the section "Prepare for Migration" in File and Storage Services: Prepare to Migrate.

Review the following dependencies before you run the command.

- When you run the **Export-SmigServerSetting** cmdlet, you are prompted to provide a password to encrypt the migration store data. You must provide this same password to import data from the migration store.
- The *path* parameter can be to a folder that is empty or one that contains data. The actual data file in the folder (Svrmig.mig) is created by the **Export-SmigServerSetting** cmdlet. Therefore, the user does not have to specify a file name.
- If the path is not a shared location that the destination server can read, you must manually copy the migration store to the destination server or a location that the destination server can access.
- If a migration store location already exists and you want to rerun the **Export-SmigServerSetting** cmdlet, you must move the Svrmig.mig file from the migration store location and then store it elsewhere, or rename or delete the Svrmig.mig file first.

2. On the source server, type the following, and then press ENTER, where <storepath> is the path that will contain the Svrmig.mig file after this step is completed. An example of the path is \\fileserver\users\username\branchcachestore.

```
Export-SmigServerSetting -featureID BranchCache -Path
<storepath\BranchCache> -Verbose
```

Group or local policy specific to SMB and Offline Files

Most SMB and Offline Files settings are migrated as part of shared folder migration. The remaining settings that affect the server are set through group or local policies. This section describes how to inventory SMB and Offline Files settings that are controlled by Group Policy.

Server message block

Determine the policy settings that affect the SMB server. The SMB settings are controlled by Group Policy settings or local policy settings. If a Group Policy object (GPO) is applied, these policies override the local settings. First, determine if the settings are controlled by a GPO, and then determine local settings for anything that is not controlled by the GPO.

To determine if a GPO is applied to the source server

1. Open the Resultant Set of Policy snap-in. To open the Resultant Set of Policy snap-in, open a command prompt, type **rsop.msc**, and then press Enter.
2. In the snap-in tree pane, click **Computer Configuration**, click **Windows Settings**, click **Security Settings**, click **Local Policies**, and then click **Security Options**.
3. Note in the SMB data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets any Group Policy setting that affects the following

Microsoft® network server settings:

- Microsoft network server: Amount of idle time required before suspending session
- Microsoft network server: Attempt S4USelf to obtain claim information
- Microsoft network server: Digitally sign communications (always)
- Microsoft network server: Digitally sign communications (if client agrees)
- Microsoft network server: Disconnect clients when logon hours expire

On source servers that are running the Server Core installation option of the Windows Server 2008 R2 or Windows Server 2012 operating system, run the **gpresult** command to review Group Policy settings (for more information about **gpresult**, type **gpresult /?** at a command prompt.)

Notes

For any setting that is controlled by Group Policy, you must apply the same GPO to the destination server, or you can set the local policy of the destination server for the same behavior.

For any setting that is not controlled by Group Policy, use the following procedure to determine the local policy setting. Note the local policy setting in the SMB data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets.

To determine local policy settings

1. Open the Local Group Policy Editor. To open the Local Group Policy Editor, open a command prompt, type **gpedit.msc**, and then press Enter.
2. In the snap-in tree pane, click **Computer Configuration**, click **Windows Settings**, click **Security Settings**, click **Local Policies**, and then click **Security Options**.
3. Note the following settings for **Microsoft network server**:
   - Microsoft network server: Amount of idle time required before suspending a session
   - Microsoft network server: Attempt S4USelf to obtain claim information
   - Microsoft network server: Digitally sign communications (always)
   - Microsoft network server: Digitally sign communications (if client agrees)
   - Microsoft network server: Disconnect clients when logon hours expire

   On source servers that are running the Server Core installation, run the **secedit** command to export and review local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt.)

Offline Files

Note

This section only applies to source servers that are running Windows Server 2012 R2 Windows Server 2012, Windows Server 2008 R2, or. Previous operating system releases do not have Offline Files settings that affect the server.

Determine the policy settings that affect file shares on the server for which client computers use Offline Files. The Offline Files settings are controlled through Group Policy or local policy. If Group Policy is applied, then these policies override local settings. First, determine if the settings are controlled through Group Policy, then determine the local settings for anything that is not controlled by using Group Policy.

To determine if Group Policy is applied to the source server

1. Open the Resultant Set of Policy snap-in. To open the Resultant Set of Policy snap-in, open a command prompt, type **rsop.msc**, and then press Enter.
2. In the snap-in tree pane, click **Computer Configuration**, click **Administrative Templates**, click **Network**, and then click **Lanman Server**.

   Note
   If no policies are set, the preceding path won't exist. If the path does not exist, skip to the procedure To determine local policy settings. If the path exists and policies are found, go on to the next step.
3. Note in the BranchCache data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets any Group Policy settings that control the **Hash Publication for BranchCache** and **Hash Version support for BranchCache** settings.

   On source servers that are running the Server Core installation option, run the **gpresult** command to review Group Policy settings (for more information about **gpresult**, type **gpresult /?** at a command prompt).

For any setting controlled by Group Policy, have the same Group Policy apply to the destination server, or you can choose to set the local policy of the destination server to get the same behavior.

For any setting not controlled by Group Policy, use the following instructions to determine the local policy setting.

To determine local policy settings

1. Open the Local Group Policy Editor. To open the Local Group Policy Editor, open a command prompt, type **gpedit.msc**, and then press Enter.
2. In the snap-in tree pane, click **Computer Configuration**, click **Administrative Templates**, click **Network**, and then click **Lanman Server**.
3. Note in the BranchCache data collection worksheet in File and Storage Services: Appendix B: Migration Data Collection Worksheets the value of the **Hash Publication for BranchCache** and **Hash Version support for BranchCache** settings.

   On source servers that are running the Server Core installation option, run the **secedit** command to export and review local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt).

DFS Namespace configuration

Procedures in this section describe how to migrate DFS Namespaces from the source server to the destination server.

Before the migration of the namespace begins, you can inventory the namespaces that are hosted on the source server for tracking purposes. You can do this by using DFS Management or DFSUtil.exe.

The following procedure (To inventory DFS Namespaces by usingDFS Management) applies only to computers that are running at least the Windows Server 2003 R2 version of the Windows Server operating system. For computers that are running Windows Server 2003, you can perform a DFS Namespace inventory by using **DFSUtil.exe** as described in To inventory DFS Namespaces by using DFSutil.exe.

You can also perform a DFS Namespace inventory from a client computer that is running Windows Vista®, Windows® 7, or Windows® 8 by using DFS Management that is part of Remote Server Administration Tools.

- To download Remote Server Administration Tools for Windows Vista, see Microsoft Remote Server Administration Tools for Windows Vista (http://go.microsoft.com/fwlink/?LinkID=113192).
- To download Remote Server Administration Tools for Windows 7, see Remote Server Administration Tools for Windows 7 (http://go.microsoft.com/fwlink/?LinkID=131280).
- To download Remote Server Administration Tools for Windows® 8, see Remote Server Administration Tools for Windows 7 (http://go.microsoft.com/fwlink/?LinkID=131280).

To inventory DFS Namespaces by using DFS Management

1. Under DFS Management in the left pane, right-click **Namespaces**.
2. Select **Add Namespaces to Display**.
3. In the dialog box that is displayed, select **Server** from the Scope options.
4. Type the name of source server and click **Show Namespaces**.
5. Select all namespaces listed in the list box and click **OK**.
6. Right-click the first namespace listed in the left pane.
7. Select **Properties**.
8. On the **General** tab, check the **Type** field. The type of namespace that is hosted on the server is described here. Possible values are stand-alone, domain-based (Windows Server 2000 mode), and domain-based (Windows Server 2008 mode).
9. In the case of a domain-based namespace, click the **Namespace Servers** tab to identify the number of servers that host the namespace.
10. Repeat steps 7 through 10 for the remaining namespaces listed in the left pane.

To inventory DFS Namespaces by using DFSutil.exe

1. You can inventory your DFS Namespaces using DFSUtil.exe by using the command prompt. From a command prompt, type **DFSUtil.exe server SourceServer** where

SourceServer represents the name of the source server.

2. Identify the namespaces (DFS roots) listed for the source server.
3. Type the following command, list the namespace properties for the first namespace identified in step 2:

```
DFSUtil.exe root <\\SourceServer\Namespace>
```

4. Identify the namespace type; possible values are stand-alone root, domain root (domain-based namespace in Windows 2000 Server mode), domainV2 root (domain-based namespace in Windows 2008 mode).
5. Identify the DFS folders present in the namespace in each of the **Link Name** items displayed.
6. In the case of domain-based namespaces, identify all the namespace servers by typing the following command:

```
DFSUtil.exe root <\\Domain\Namespace>
```

7. Identify the namespace servers that host the namespace in each of the **Target** items displayed under **Root Name**.
8. Repeat steps 3 through 7 for the remaining namespaces on the source server.

Considerations for namespaces

Is the namespace stand-alone or domain-based? If the namespace is stand-alone, see the following section in this document:

Stand-alone namespaces.

If the namespace is domain-based, consider the number of namespace servers for each namespace. For more information, see the following sections in this document:

Domain-based namespaces with more than one namespace server

Domain-based namespaces with one namespace server

Stand-alone namespaces

Complete the following procedure to export a stand-alone namespace configuration.

To export the namespace configuration to an export file

1. On the destination server, open a Command Prompt window.
2. Type **DFSUtil.exe root export \\SourceServer\Namespace FileName** the following to export the standalone namespace to a file (where *FileName* represents the exported file), and then press ENTER.

Domain-based namespaces with more than one namespace server

For more than one namespace server, remove the namespace server from the namespace by using DFSUtil.exe.

To remove the namespace server from the namespace

1. On the destination server, open a Command Prompt window.

2. Type **DFSUtil.exe target remove <\\SourceServer\Namespace>**, and then press ENTER.

Domain-based namespaces with one namespace server

There are two options that you can use in this scenario: Export the namespace settings or add a temporary server to the namespace.

To export namespace settings

1. On the destination server, open a Command Prompt window.
2. Type **DFSUtil.exe root export \\Domain\Namespace FileName** where *FileName* represents the file containing settings for export, and then press ENTER.

   Note
   For each namespace, there must be a different file name to export settings.
3. Repeat step 2 for each namespace for which the source server is a namespace server.

You can use either of the following two options if a temporary server can be added to the namespace. This provides the ability to maintain the namespace online while the migration progresses. If this is not possible, follow the steps in To remove the namespace server from the namespace instead.

To add a temporary server to the namespace by using DFS Management

1. In the left pane, select the namespace to be migrated.
2. Click the **Namespace servers** tab.
3. Select **Add Namespace Server**.
4. In the **Namespace server** box, type the name of the temporary server, and then click **OK**.

The temporary server will be added to the namespace.

To add a temporary server to the namespace by using DFSUtil.exe

1. Create a shared folder for the namespace on the temporary server with the same permissions as on the source server.
2. On the destination server, open a Command Prompt window.
3. Type **DFSUtil.exe target add \\TemporaryServer\Namespace**and then press ENTER.

```
DFSUtil.exe target add <\\TemporaryServer\Namespace>
```

The temporary server will be added to the namespace.

After the namespace settings are exported or a temporary server is added to the namespace, the namespace source server can be removed from the namespace as described in To remove the namespace server from the namespace.

Inventory advanced registry keys

This section describes the process for determining if there are any settings that have been applied to the DFS Namespace component on the source server. These settings are stored in the registry and set or viewed through the dfsutil.exe tool. To inventory these settings, run the following commands from the destination server:

```
DFSUtil.exe server registry DfsDnsConfig <SourceServer>

DFSUtil.exe server registry LdapTimeoutValue <SourceServer>

DFSUtil.exe server registry SyncInterval <SourceServer>
```

Note the setting for any registry modification. Registry keys that have not been modified display a value similar to the following:

```
<KeyName> does not exist in the Registry.
```

DFS Replication configuration

To migrate DFS Replication settings, use the following Microsoft Enterprise Support blog series:Replacing DFSR Member Hardware or OS.

File Server Resource Manager configuration on the source server

When you migrate File Server Resource Manager, remember to use the same drive letters for the destination server volumes as for the source server. This is required because the File Server Resource Manager migration requires that the drive letter remains the same.

1. Stop the File Server Resource Manager and File Server Storage Reports Manager services. You can stop these services in Windows PowerShell by using the following command: **Stop-Service –Name “srmsvc”, “srmreports”**.
2. Export the File Server Resource Manager configuration. You can export the File Server Resource Manager configuration in Windows PowerShell by using the following command: **Export-SmigServerSetting -FeatureID FS-Resource-Manager -Path <storepath\FSRM> -Verbose**.
3. For each volume, get the configuration files by running the following commands in the Windows PowerShell session.
   a. Stop the file screen driver. Type **fltmc detach datascrn <VolumeLetter>:**, and then press ENTER.
   b. Stop the quota driver. Type **fltmc detach quota <VolumeLetter>:**, and then press ENTER.
   c. Grant Read permissions to the Administrator account for the "<*VolumeLetter*>:\System Volume information\SRM" folder and the following child objects:
      - takeown /F "<VolumeLetter>:\System Volume Information" /A /R /D Y
      - cacls "<VolumeLetter>:\System Volume Information" /T /E /G Administrators:F
      - attrib -S -H "<VolumeLetter>:\System Volume Information*" /S /D
   d. Copy the following files from the SRM folder to an external storage device:
      - Quota.xml

- Quota.md
- Datascrn.md
- DataScreenDatabase.xml

e. Start the file screen driver. Type **fltmc attach datascrn <VolumeLetter>:**, and then press ENTER.

f. Start the quota driver. Type **fltmc attach quota <VolumeLetter>:**, and then press ENTER.

4. Restart the File Server Resource Manager and File Server Storage Reports Manage services. Type **Start-Service -name "srmsvc","srmreports"**, and then press ENTER.

5. Configure scheduled reports.

   File Server Resource Manager reports and classification rule configurations are dependent on the drive letters and mount points. Any drives or mount points on the source server that are used by report or classification rule configurations must be available on the destination server or the configurations must be updated during import.

   To configure scheduled reports, follow step (a). However, if you are migrating from Windows Server 2003, follow step (b).

   - To configure scheduled reports on all servers except Windows Server 2003, run the following commands in a Windows PowerShell session on the source server that was opened with elevated user rights.
     - To get a list of all the task names associated with storage reports: `storrept r l`
     - For each task name listed run the following command on the source server: `schtasks /query /tn:"TASKNAME" /xml > "TASKNAME.xml"`
   - To configure scheduled reports when you migrate from Windows Server 2003:
     - On the source server, do the following:
       - Open File Server Resource Manager.
       - In storage report management, for each report task, note the report task, target, and schedule.
     - On the destination server, after you import the file server resource manager configuration, do the following:
       - Open File Server Resource Manager.
       - In **Storage Report Management**, for each report task, edit the report task properties.
       - On the **Schedule** tab, manually add the appropriate schedule for the report.

6. Configure scheduled file management tasks. This step applies only to source servers that are running Windows Server 2008 R2 or Windows Server 2012.

   a. To display a list of all task names associated with file management tasks, type the following command on the source server in a Windows PowerShell session opened with elevated user rights:

   ```
   (new-object -com
   Fsrm.FsrmFileManagementJobManager).EnumFileManagementJobs()
   ```

b. For each entry listed, locate the task element, and then type the following command:

```
Schtasks /query /tn:"TASK" /xml > "TASK.xml"
```

7. Export the classification schedule. This is only applicable to servers running Windows Server 2008 R2 or Windows Server 2012 that already have a classification schedule configured. From an elevated command prompt, type the following command:

```
Schtasks /query /tn:”FsrmAutoClassification{c94c42c4-08d5-473d-8b2d-98ea77d55acd}” /xml > “classification.xml”
```

Shadow Copies of Shared Folders

The following procedures describe how to migrate shadow copy settings.

To migrate shadow copy settings

1. Open Windows Explorer on the source server to view shadow copy storage locations and the creation schedule.

   Important
   This procedure applies to shadow copies for a server running the full installation option of Windows Server. If your source server is running the Server Core installation option of Windows Server, skip this procedure and follow the instructions in the following section: To migrate shadow copies in a Server Core installation.

2. For each volume on the source server, right-click the volume, select **Configure Shadow Copies**.

   On source servers that are running Windows Server 2003, right-click the volume, click **Properties**, and then click the **Shadow Copies** tab.

3. Click **Settings**, and note the location and size of the shadow copy storage.
4. Click **Schedule** and note the details for the snapshot creation task.

To migrate shadow copies in a Server Core installation

1. Log on to the computer that is running a Server Core installation remotely as follows:
   a. In Server Manager, click **Tools**, and then click **Computer Management**.
   b. In the **Computer Management** snap-in pane, right-click the top node, and then click **Connect to another computer**.
2. Type the computer name, and then click **OK**.
3. Expand **System Tools**, right-click **Shared Folders**, click the **All Tasks** tab, and then click **Configure Shadow Copies**.
4. For each volume on the source server, right-click the volume, select **Configure Shadow Copies**, click **Settings**, and note the location and size of the shadow copy storage.
5. Click **Schedule**, and then note details for the snapshot creation task.

Migrate local users and groups to the destination server

Before migrating data and shared folders, or completing your migration of the FSRM configuration, you must migrate local users and groups. Export local users and groups from the source server, and then import local users and groups to the destination server.

Important

If the source server is a domain member server, but the destination server is a domain controller, imported local users are elevated to domain users, and imported local groups become Domain Local groups on the destination server.

If the source server is a domain controller, but the destination server is not, Domain Local groups are migrated as local groups, and domain users are migrated as local users.

Export local users and groups from the source server

On the source server, export local users and groups to a migration store (as shown in the following example) in a Windows PowerShell session that has been opened with elevated user rights.

```
Export-SmigServerSetting -User All -Group -Path <storepath\UsersGroups> -Verbose
```

You can use one of the following values with the *-user* parameter:

- Enabled: Specify to export only enabled local users.
- Disabled: Specify to export only disabled local users.
- All: Specify to export enabled and disabled local users.

For more information about the attributes of local users and groups that can be migrated, see the Local User and Group Migration Guide (http://go.microsoft.com/fwlink/?LinkID=258341) on the Microsoft Web site.

You are prompted to provide a password to encrypt the migration store. Remember this password, because you must provide the same password to import from the migration store.

If the path is not a shared location that is accessible to the destination server, you must manually copy the contents of the migration store folder to the destination server or a location that is accessible to the destination server.

Import local users and groups to the destination server

On the destination server, import local users and groups from the migration store to which you exported the users and groups in Export local users and groups from the source server, as illustrated by the following example. Use a Windows PowerShell session that has been opened with elevated user rights.

```
Import-SmigServerSetting -User All -Group -Path <storepath\UsersGroups> -Verbose
```

You can use one of the following values with the *-user* parameter:

- Enabled: Specify to import only enabled local users.

- Disabled: Specify to import only disabled local users.
- All: Specify to import enabled and disabled local users.
- For the list of user attributes that are supported for migration, see the Local User and Group Migration Guide (http://go.microsoft.com/fwlink/?LinkID=258341).

You are prompted to provide the same password that you provided during export to decrypt the migration store.

Migrate data

To migrate data, you can copy file data or physically move it, for example, by attaching the storage drive from the source server to the destination server. If you copy the data, follow the copy data migration steps in the following section.

If you physically move the data, follow the steps described in the Physical data migration section later in this document.

Data copy migration

If you are planning a two-phase data copy migration as described in the previous section, note that if files have been deleted on the source server between the start of the first copy and the start of the final copy, copies of the deleted files might have already transferred to the destination server. So if a file is deleted between the two copy processes, the file might still be available on the destination server after the migration is complete. If this is unacceptable in your environment, perform data and shared folder migration in a single phase, and disconnect all users before starting migration.

Important
The file migration portion of the Windows Server Migration Tools is designed for smaller data sets (under 100GB of data). It copies files one at a time over HTTPS. For larger datasets, we recommend using the version of robocopy.exe included with Windows Server 2012 R2 or Windows Server 2012.

To copy data and shared folders and migrate all data without disconnecting users

1. Verify that the destination path has sufficient disk space to migrate the data. If NTFS or folder quota management is enabled on the destination server disk drive, verify that the NTFS or File Server Resource Manager quota limit allows for sufficient free disk space to migrate data. For more information about quota management in File Server Resource Manager, see one of the following.
   - Quota Management (http://go.microsoft.com/fwlink/?LinkId=154277) for Windows Server 2008, Windows Server 2008 R2, and Windows Server 2012
   - Quota Management (http://go.microsoft.com/fwlink/?LinkId=154241) for Windows Server 2003 R2

   For more information about NTFS quota management, see one of the following.
   - Setting Disk Quotas (http://go.microsoft.com/fwlink/?LinkId=154243) for Windows

Server 2008, Windows Server 2008 R2, and Windows Server 2012

- Enable disk quotas (http://go.microsoft.com/fwlink/?LinkId=154245) for Windows Server 2003 and Windows Server 2003 R2

2. Ensure that you have completed the migration of local users and groups.

   Send-SmigServerData and **Receive-SmigServerData** cmdlets must be run on the source and destination server within five minutes of each other. By default, **Send-SmigServerData** and **Receive-SmigServerData** time out if a connection cannot be established within 300 seconds (five minutes). This maximum connection time-out for the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets is stored in the following registry key, which is user-defined.

   Key: HKEY_LOCAL_MACHINE\Software\Microsoft\ServerMigration

   Value: MaxConnectionTime (REG_DWORD)

   Data: Between 1 and 3600 (represents connection time-out, in seconds)

   If a value larger than 3600 is specified, 3600 seconds (1 hour) is used as the maximum connection time-out.

   For information about how to create a Windows Registry key, see Add a Registry Key (http://go.microsoft.com/fwlink/?LinkId=147298) on the Microsoft Web site.

3. Use the following command to run the **Receive-SmigServerData** cmdlet on the destination server. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Receive-SmigServerData
   ```

   Note
   All output for the Send and Receive operations occurs on the source server only. The destination server will appear to be done before the operation has completed.

4. Use the following command to run the **Send-SmigServerData** cmdlet on the source server to migrate data and shared folders. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Send-SmigServerData -ComputerName <DestinationServer> -
   SourcePath d:\users -DestinationPath d:\shares\users -Recurse
   -Include All -Force
   ```

The destination data location does not have to be the same as the source location, and you can change it, if desired.

Notes

The Server service startup type must be set to Automatic on the destination server for shared folder migration to complete.

Data that is transferred is encrypted automatically. You are prompted to enter a password to encrypt the transferred data on the source server, and the same password to decrypt the received data on the destination server.

After the first data copy is finished, you must freeze the source server and all data changes.

To disconnect users and migrate new or updated files

1. Make sure that users are notified that they should stop using the source server at this time to prevent any possible data loss. You can run the following command to list all the currently open files to determine the potential impact of performing this step.

   ```
   net file
   ```

2. Disconnect all users from the source server by stopping the LanMan server service.

   ```
   Stop-Service LanmanServer -force
   ```

   Stopping the LanMan Server service invalidates all open remote files to the shared folders on the source server, which can lead to potential data loss. It is best to perform this step when the fewest users are expected to access files on this server.

3. Use the following command to run the **Receive-SmigServerData** cmdlet on the destination server. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Receive-SmigServerData
   ```

4. Use the following command to run the **Send-SmigServerData** cmdlet on the source server to migrate data and shared folders. Use a Windows PowerShell session that is running with elevated user rights.

   ```
   Send-SmigServerData -ComputerName <DestinationServer> -
   SourcePath d:\users -DestinationPath d:\shares\users -Recurse
   -Include All -Force
   ```

5. If your scenario requires migrating reparse points, hard links, and mount points, recreate them on the destination server by using the **mklink** command for reparse points and hard links, and using the **mountvol** command for mounted volumes. For more information about these commands, enter `mklink /?` or `mountvol /?` in a Windows Command Prompt.

It is important to maintain the same destination path that you used during the first copy of data and shared folders. The cmdlets transfer files, folders, and shared folders only if they do not exist on the destination server, or if there is a new version on the source server.

Physical data migration

The next sections describe data migration by physically moving external drives or logical unit numbers (LUNs).

Using disk drives or LUNs to migrate data from the source server to the destination server

You can migrate data from the source server by moving the disk drives. Or, if your data resides on a LUN storage device, you have the option of moving the file server data by masking the LUNs from the source server and unmasking them on the destination server.

For the ideal migration, make sure that you maintain the same mapping of the drive letters (for example, drive D) and the volume IDs (see the following explanation) so that relevant data and application information remains as consistent as possible during the move.

Caution

You should not move a disk drive or LUN if it contains both data and the operating system.

Benefits of physical migration:

- For large amounts of data, this is a faster operation.
- You maintain all data on the disk drive, such as hard links and mount points.
- Shadow copies are preserved if the shadow copies are on the migrated disk drive.

Potential issues to be aware of:

- Permissions for local users that are not default computer accounts (such as local administrators) will be lost even if the same user name is used when creating the user account on the destination server.
- Encrypted files (EFS) cannot be migrated.
- Encrypted volumes with BitLocker cannot be migrated without first decrypting the volumes.
- Remote Storage cannot be migrated.
- When you are physically migrating disk drives that have File Server Resource Manager quotas enabled on them, it is a best practice to dismount the drive gracefully to avoid marking the quotas as dirty. Otherwise, unnecessary scans may occur later.

To migrate data by physically moving the disk drive or by masking and unmasking the LUNs

1. Collect information on the source server.

   Tip

   You can use Server Manager or Windows PowerShell on a computer running Windows Server 2012 or Windows 8 to collect information from source computers running Windows Server 2012.

   a. Record the drive letter and volume label for each data volume on the source server that you would like to move to the destination server.

   b. On the source server, export the volume GUID paths by exporting the following registry key to a file: HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices. To do this, open the Registry Editor (regedit.exe), browse to the registry key, right-click the registry key, and clicking **Export**.

      Alternatively, to export the volume GUID paths from a server running Windows Server 2012 or Windows Server 2008 R2, open a Windows PowerShell session, and then type the following commands, where <*SourceServer*> is the name of the source server, <*Domain**User*> is a user account with administrative permissions on the source server and <*LocalPath*>\<*Filename*> is a local path and filename of the exported registry keys:

```
Enter-PSSession <SourceServer> -Credential <Domain\User>
Regedit.exe /E <LocalPath>\<Filename>.reg
"HKEY_LOCAL_MACHINE\SYSTEM\MountedDevices"
```

Note

To user Server Manager or Windows PowerShell to remotely collect information from earlier versions of Windows Server you must first setup the source server for remote management. For more information, see Managing Downlevel Windows-based Servers from Server Manager in Windows Server 2012.

c. Open Notepad and copy the exported .reg file. Remove all entries that are in the following form: \DosDevices\D:. Save the.reg file (all remaining entries should be in the following form: \??\Volume{ef93fe94-5dd7-11dd-961a-001e4cdb4059}).

2. Prepare the destination server.

a. In the Server Manager navigation pane, click **File and Storage Services**, and then click **Volumes** to display the Volumes page. Use the Volumes tile to make sure that the drive letters for the data volumes are available. If there is a drive letter that is currently assigned to an existing volume on the destination server, change the drive letter for that volume.

Alternatively, use the Windows PowerShell **Get-Volume** and **Set-Partition** cmdlets. For example, to get any volumes with the drive letters of F, G, or H, type `Get-Volume F,G,H`. To change the drive letter of a partition with the F drive letter, type `Set-Partition -DriveLetter F -NewDriveLetter Z`

b. To import the volume GUID paths into the destination server, copy the.reg file that you created previously to the destination server, and then double-click that file to update the destination server.

3. Move the disk drives or LUNs from the source server to the destination server.

a. On the source server, remove the disk drives or unassign the LUNs by using Storage Manager for SANs. (To open Storage Manager for SANs, click **Start**, click **Administrative Tools**, and then click **Storage Manager for SANs**.) If the source server is running Windows Server 2012, instead use the File and Storage Services role in Server Manager to view the disks or virtual disks (when using storage pools) that you want to move. If the disk is part of a storage pool, on the **Storage Pools** page of the File and Storage Services role right-click the virtual disk, and then click **Detach Virtual Disk**. For other types of disks, on the **Disks** page, right-click the disk that you want to move and then click **Take Offline**.

b. On the destination server, attach each disk drive or assign the LUNs, and then assign the appropriate drive letter by using the **Disks** and **Storage Pools** pages of the File and Storage Services role in Server Manager.

4. If any files or folders on the migrated drive use local users or local groups permissions (except default users and groups), re-create these permission. Note that all domain users and groups permissions will remain intact, assuming that the source server and the destination server are members of the same domain.

Notes

You can use the `icacls` command to modify file and folder permissions (type `icacls /?` in a Command Prompt window for details). Type this command in a Windows PowerShell session or a command prompt that has been opened with elevated user rights.

The list of the default users and groups is available in the topic Default User Accounts and Groups (http://go.microsoft.com/fwlink/?LinkId=149889) on the Microsoft Web site.

Migrate shared folders

If any of the folders on the migrated drive were shared on the source server, and they must be shared on the destination server, the following steps explain how to migrate shared folders.

1. If any of the migrated shared folders use local users and group permissions, ensure that you have completed the migration of local users and groups.

   Send-SmigServerData and **Receive-SmigServerData** cmdlets must be started on the source server and the destination server within five minutes of each other. By default, **Send-SmigServerData** and **Receive-SmigServerData** operations terminate if a connection cannot be established within 300 seconds (five minutes). The maximum connection time-out for the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets is stored in the following registry key, which is user-defined.

   Key: HKLM\Software\Microsoft\ServerMigration

   Value: MaxConnectionTime (REG_DWORD)

   Data: Between 1 and 3600 (represents connection time-out, in seconds). If a value larger than 3600 is specified, 3600 seconds (one hour) is used as the maximum connection time-out.

   For information about how to create a Windows Registry key, see Add a Registry Key (http://go.microsoft.com/fwlink/?LinkId=147298) on the Microsoft Web site.

2. Open port 7000 on the source server and destination server (if this has not already been done).

   For information about how to open a port in Windows Firewall, see File and Storage Services: Appendix A: Optional Procedures.

3. On the destination server:
   a. Open a Windows PowerShell session with elevated user rights and enter the following command: **Receive-SmigServerData**.

4. On the source server:
   a. Open a Windows PowerShell session in Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012. On computers that are running Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012, the Windows PowerShell session must be opened with elevated user rights. Enter the following command: **Send-SmigServerData -ComputerName <DestinationServerName> -SourcePath <SourcePath> -DestinationPath <DestinationPath> -Recurse -Include Share -Force**

Notes

The <*SourcePath*> value specifies the local path on the source server that contained the shared folder before the drive was migrated. Shared folder information is not stored on the data drive, so do not be concerned that the drive no longer resides on the source server.

The <*DestinationPath*> value specifies the local path on the destination server that contains folders that were previously shared on the source server. Unless the root drive letter or the folder structure has been changed on the migrated drive, <*SourcePath*> and <*DestinationPath*> values should be the same.

During shared folder migration, permissions for local users and groups and domain users and groups are migrated—no manual remapping is required.

LanMan Server service automatically restarts on the destination server, and the shared folders migrate.

DFS Replication migration

If you physically migrated data, clean-up the DFS Replication configuration state, which is stored on the migrated volume:

1. To clean up volumes (for each physically migrated volume)
   a. Navigate to the path <volume>\System Volume Information.

      Note

      This is a hidden system folder. To view this folder: in File Explorer, click **View**, and then select the **Hidden Items** check box.. Also ensure that local administrators are granted **Full Control** of the folder.

   b. Delete the **DFSR** folder and all content in the folder.
   c. Revert any security permissions modifications that you made to perform the migration process.
   d. Repeat this process for all physically migrated volumes.
2. To clean up replicated folders (for replicated folders on physically migrated volumes)
   a. Navigate to the root of a replicated folder.
   b. Delete the **DfsrPrivate** folder and all subfolders.
   c. If the staging folder for the replicated folder is not located in the default location, remove the staging folder and all content in the staging folder.

      Note

      The default location for the staging folder is in the DfsrPrivate folder, and this step is not required if the path is at the default location.

   d. If the Conflict and Deleted folder for the replicated folder is not located in the default location, remove the Conflict and Deleted folder and all content in the Conflict and Deleted folder.

Note

The default location for the Conflict and Deleted folder is in the DfsrPrivate folder, and this step is not required if the path is at the default location.

Use the inventoried information that you collected for the source server to detect all replication groups to which the source server belongs. Add the destination server as a member server to all these replication groups.

Migrate the source server identity

You need to rename the source server and migrate its previous identity to the destination server. You might also need to migrate the source server IP address to the destination server.

Rename the source server

Rename the source server to a temporary name.

Migrate IP address

When a static IP address is used on the source server, it is recommended that the IP address be migrated from the source server to the destination server. This is because client computers locally cache the IP address that is associated with a server name. Client computers will still attempt to access the source server even if it has been renamed.

When the server IP address is not migrated, you must stop the LanMan Server service on the source server. This is done to prevent users from accessing shared folders on the source server after they have been migrated to the destination server. Open a Windows PowerShell session with elevated user rights, and then run the following cmdlet:

```
Stop-Service LanmanServer -Force
```

For more information on IP address migration, see Migrate IP Configuration to Windows Server 2012.

Rename destination server

Rename the destination server to the name that was originally used for the source server.

Configure DFS Replication on the destination server

Configuration of DFS Replication on the destination server is determined by whether you migrated the data by copying or physically moving it

If you migrated the data by copying it

Follow this procedure to add a replication connection between the source server and the destination server for each replication group on the source server:

1. In Server Manager, click **Tools**, and then click **DFS Management**.
2. In the console tree, under the **Replication** node, select **Add Replication Groups to Display**, enter the name of the source, and then click **Show Replication Groups**. Select all of the replication groups that are displayed, and then click **OK**.
3. For each replication group, do the following:
    a. Click the replication group, and then click **New Member**. The **New Member Wizard** appears. Follow the instructions in the wizard to add the destination server to the replication group by using the information from question #2 in the DFS Replication Data Collection Worksheet (File and Storage Services: Appendix B: Migration Data Collection Worksheets).
    b. In the console tree, under the **Replication** node, right-click the replication group that you just added the destination server to, and then click **New Connection**.
    c. Specify the source server and destination server as sending and receiving members, and specify a schedule so that the connection is always enabled. At this point, the replication is one-way.
    d. Select **Create a second connection in the opposite direction** to create a second connection for two-way replication between the sending and receiving members.

If you migrated the data by physically moving it

Follow this procedure to add a replication connection between the destination server and the closest server to the destination server other than the source server:

1. In Server Manager, click **Tools**, and then click **DFS Management**.
2. In the console tree, under the **Replication** node, select **Add Replication Groups to Display**, enter the name of the source, and then click **Show Replication Groups**. Select all of the replication groups that are displayed, and then click **OK**.
3. For each replication group:
    a. Click the replication group, and then click **New Member**. The **New Member Wizard** appears. Follow the instructions in the wizard to add the destination server to the replication group by using the information from question #2 in the DFS Replication Data Collection Worksheet (File and Storage Services: Appendix B: Migration Data Collection Worksheets).
    b. In the console tree, under the **Replication** node, right-click the replication group that you just added the destination server to, and then click **New Connection**.
    c. Specify the destination server as the sending member, and then specify any other server except the source server as the receiving member. Specify the schedule to use for the connection. It is recommended that you select a server that has a good network connection to the destination server as the receiving member.
    d. Select **Create a second connection in the opposite direction** to create a connection for two-way replication between the sending and receiving members.

Notes

The folder does not begin to replicate immediately. The new DFS Replication settings must be replicated to all domain controllers, and each member in the replication group must poll its closest domain controller to obtain these settings. The amount of time this takes depends on Active Directory Domain Services (AD DS) replication latency and the polling interval (60 minutes) on each member.

The **dfsrdiag /pollad** command can be used to force DFS Replication on the source server and destination server to poll AD DS and retrieve the latest configuration information instead of waiting for the next normal polling interval which could be up to 60 minutes.

After DFS Replication on the destination server polls AD DS, it begins to replicate the folders that it configured, and it performs an initial synchronization. Event ID 4102 (MSG_EVENT_DFSR_CS_INITIAL_SYNC_NEEDED) is registered in the event log on the destination server for each replicated folder.

During initial sync, DFS Replication downloads all files in the replicated folders from the source server and builds up a local copy of the database per volume. This process can be time consuming. It is possible to speed up the initial sync by seeding the data from the source server onto the destination server (from the backup that was taken prior to commencing migration).

When the initial sync completes, event ID 4104 (MSG_EVENT_DFSR_CS_INITIAL_SYNC_COMPLETED) is registered for each replicated folder on the destination server. Monitor each replicated folder on the destination server, and check to ensure that all of them have completed the initial sync.

Import settings to the destination server

Follow the procedures in this section to import settings to the destination server.

Note

If the source server is not running Windows Server 2008 R2 or Windows Server 2012, the first procedure in this section does not apply. (This procedure is used to migrate the seed value that is used by BranchCache for the Network Files component, and it enables data that is stored in BranchCache on the source server to be used after it is migrated to the destination server. For information about how to migrate a BranchCache host server, see the BranchCache Migration Guide (http://go.microsoft.com/fwlink/?LinkID=139091).

To set up BranchCache for Network Files migration on the destination server

1. On the destination server, open a Windows PowerShell session with elevated user rights.
2. Type the following command, where *storepath* is the available path that contains the Svrmig.mig file, and then press ENTER.

```
Import-SmigServerSetting -featureid BranchCache -Path
```

```
<storepath\BranchCache> -Force -Verbose
```

Group Policy or local policy specific to server message block and Offline Files

Use a Group Policy object or a local policy on the destination server to change the settings to the same values as the source server. These settings are recorded in the SMB and BranchCache data collection worksheets in File and Storage Services: Appendix B: Migration Data Collection Worksheets.

To import server message block settings

1. Do one of the following:
   - If the policies are set by using Group Policy objects, use the Group Policy editing tools to apply appropriate policies to the destination server.
   - If the policies are set by using a local policy, do the following:
     i. On the destination server, open the Local Group Policy Editor snap-in.
     ii. In the snap-in tree pane, click **Computer Configuration**, click **Windows Settings**, click **Security Settings**, click **Local Policies**, and then click **Security Options**.
2. Use a Group Policy object or a local policy to set the following settings to the same values as noted in Export settings. Set the destination server settings to the same values as were noted on the source server for the following settings:
   - Microsoft network server: Amount of idle time required before suspending a session
   - Microsoft network server: Attempt S4USelf to obtain claim information
   - Microsoft network server: Digitally sign communications (always)
   - Microsoft network server: Digitally sign communications (if client agrees)
   - Microsoft network server: Disconnect clients when logon hours expire

   Note
   For any setting that is controlled by Group Policy, you must have the same Group Policy object apply to the destination server, or you can set the local policy of the destination server to get the same behavior.

   On destination servers that are running the Server Core installation, run the **secedit** command to change local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt).

Note

The following procedure applies only if the source server is Windows Server 2008 R2 or Windows Server 2012.

To import Offline Files settings

1. Do one of the following:
   - If the policies are set by using Group Policy, use the Group Policy editing tools to apply appropriate policies to the destination server.
   - If the policies are set by using local policy, do the following:
     i. On the destination server, open the Local Group Policy Editor snap-in.
     ii. In the snap-in tree pane, click **Computer Configuration**, click **Windows Settings**, click **Administrative Templates**, click **Network**, and then click **LanMan Server**.
2. Use a Group Policy object or a local policy to set the destination server policy settings to the same values as the source server settings for **Hash Publication for BranchCache** and **Hash Version support for BranchCache** settings.

   On destination servers that are running the Server Core installation, run the **secedit** command to change local policy settings (for more information about **secedit**, type **secedit /?** at a command prompt).

DFS Namespace configuration

Complete the configuration of namespaces on the destination server. The procedure you use depends on whether you want a stand-alone or a domain-based namespace.

Stand-alone namespaces

Domain-based namespaces with more than one namespace server

Domain-based namespaces with one namespace server

Stand-alone namespaces

If you want a stand-alone namespace, you must first create the namespace on the destination server. You can do this by using DFS Management, or the **DFSUtil.exe** command-line utility.

To create the namespace on the destination server

1. Do one of the following:
   - On the destination server, open DFS Management, and create the namespace by using the same name as on the source server.
   - On the destination server, in a Command Prompt window opened with elevated user rights, type the following, and then press ENTER.

     ```
     Dfsutil.exe root addstd <\\DestinationServer\Namespace>
     ```

To import a namespace configuration from the export file

1. On the destination server, in a Command Prompt window opened with elevated user rights, type the following (in which *filename* represents the file name into which you exported namespace settings from the source server in To export the namespace configuration to an export file), and then press ENTER.

```
Dfsutil.exe root import set <filename>
<\\DestinationServer\Namespace>
```

Domain-based namespaces with more than one namespace server

If you have more than one domain-based namespace server, you can add namespace servers to your destination server by using DFS Management or the **DFSUtil.exe** command-line utility.

To use DFS Management

1. Select the namespace being migrated in the left pane.
2. Click the **Namespace servers** tab in the right pane.
3. Select **Add Namespace Server**.
4. In the dialog box that opens, type the name of the destination server, and then click **OK**.

The destination server is added to the namespace.

To use DFSUtil.exe

1. On the destination server, open a Command Prompt window.
2. Type the following command, and then press ENTER.

```
DFSUtil.exe target add <\\DestinationServer\Namespace>
```

Domain-based namespaces with one namespace server

This section applies only if a temporary server was not added to the namespace. If you added a temporary server to the namespace as part of your export process, see Domain-based namespaces with more than one namespace server.

To create the namespace on the destination server

1. Do one of the following:
   a. In DFS Management on the destination server, create the namespace with the same name that was used on the source server.
   b. Type the following command at a command prompt, and then press ENTER.

```
Dfsutil.exe root adddom <\\DestinationServer\Namespace>
```

To import a namespace configuration from the export file

1. On the destination server, open a Command Prompt window.
2. Type the following command (in which *filename* represents the export file names you created in To export namespace settings). Run this command for each of the namespaces for which the source server was a namespace server.

```
DFSUtil.exe root import set <Filename>
\\DestinationServer\Namespace
```

Note

For each namespace, there must be a file name from which settings are imported.

To manually reset delegation permissions on the namespace

1. On the destination server, open DFS Management.
2. Set the permissions that you inventoried in DFS Namespace configuration. When complete, close DFS Management.

If any advanced registry keys were configured on *SourceServer*, use **DFSUtil.exe** to configure *DestinationServer* to have the same registry key settings. Run the following commands on the destination server to set the advanced registry keys.

To set advanced registry keys

1. On the destination server, open a Command Prompt window.
2. Run the following commands to set the advanced registry keys by using **DFSUtil.exe**.

```
DFSUtil.exe server registry DfsDnsConfig set
<DestinationServer>
DFSUtil.exe server registry LdapTimeoutValue set <Value>
<DestinationServer>
DFSUtil.exe server registry SyncInterval set <Value>
<DestinationServer>
```

File Server Resource Manager configuration on the destination server

When you are migrating File Server Resource Manager, remember to use the same drive letters for the destination server volumes as for the source server. This is required because File Server Resource Managermigration requires that the drive letter remains the same.

1. Stop the File Server Resource Manager and File Server Storage Reports Manager services.

   Open a Windows PowerShell session with elevated user rights, and then run the following command:

   ```
   Stop-Service -name "srmsvc","srmreports"
   ```

2. Type the following in the Windows PowerShell session, and then press ENTER.

   ```
   Import-SmigServerSetting -FeatureID FS-Resource-Manager -Path
   <storepath\FSRM> -Force
   ```

   Notes

   If the Windows features that you are migrating have not been installed on the destination server, the **Import-SmigServerSetting** cmdlet installs them as part of the

import process, along with any Windows features that they depend on. Some Windows features might require that you restart the destination server to complete the installation. After restarting the computer, you must run the cmdlet again with the **-Force** parameter to complete the import operation.

Importing FSRM settings to the destination server replaces any global FSRM configuration information that is already on the destination server.

3. Set the configuration files for each volume.

   Type the following commands in a Windows PowerShell session, and then press ENTER.

   Note
   Running the following commands on a clean computer returns an error message. It is safe to ignore this error message.

   a. Type the following code to stop the file screen driver:

      ```
      fltmc detach datascrn <VolumeLetter>:
      ```

   b. Type the following code to stop the quota driver:

      ```
      fltmc detach quota <VolumeLetter>:
      ```

   c. Add administrator Write permissions to the "<VolumeLetter>:\System Volume information\SRM" folder and the following subfolders:
      - takeown /F "<VolumeLetter>:\System Volume Information" /A /R /D Y
      - cacls "<VolumeLetter>:\System Volume Information" /T /E /G Administrators:F
      - attrib -S -H "<VolumeLetter>:\System Volume Information*" /S /D

   d. Copy the following files from the external storage to the SRM folder:
      - Quota.xml
      - Quota.md
      - Datascrn.md
      - DataScreenDatabase.xml

   e. Type the following code to start the file screen driver:

      ```
      fltmc attach datascrn <VolumeLetter>:
      ```

   f. Type the following code to start the quota driver:

      ```
      fltmc attach quota <VolumeLetter>:
      ```

4. Restart the File Server Resource Manager and File Server Storage Reports Manager services.

   Type the following in a Windows PowerShell session, and then press ENTER.

   ```
   Start-Service -name "srmsvc","srmreports"
   ```

5. Configure scheduled reports and file management tasks.

   For each scheduled report, you need to create a scheduled task on the destination server.

   Note

File Server Resource ManagerReports and classification rule configurations are dependent on the drive letters and mount points. Any drives or mount points on the source server that are used by report or classification rule configurations must be available on the destination server or the configurations must be updated during import.

After you have an eXtensible Markup Language (XML) file for each task, copy them to the destination server and run the following command in a Windows PowerShell session on the destination server for each task:

```
schtasks /create /xml:"TASKNAME.xml" /tn:"TASKNAME"
```

6. Import the classification schedule. The classification schedule requires a scheduled task on the destination server.

```
schtasks /create /xml:"classification.xml"
/tn:"FsrmAutoClassification{c94c42c4-08d5-473d-8b2d-
98ea77d55acd}"
```

classification.xml is the name of the XML file that was exported from the target server.

Shadow Copies of Shared Folders

Apply the same settings from the source server to the corresponding volumes on the destination server.

To migrate shadow copy settings for Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012

1. To configure shadow copies, right-click each volume on the destination server that had shadow copies configured on the source server, right-click the volume and select **Configure Shadow Copies**.
2. Click **Settings** and verify that the location and size of shadow copy storage matches the settings from the source server.
3. Click **Schedule** and verify that the details for the snapshot creation task match the settings from the source server.

To migrate shadow copy settings for a Server Core installation

1. Log on to the destination server that is remotely running the Server Core installation by doing the following:
   a. In Server Manager, click **Tools**, and then click **Computer Management**.
   b. In the **Computer Management** snap-in tree pane, right-click the top node, and then click **Connect to another computer**.
2. Enter the computer name, and then click **OK**.
3. Expand **System Tools**, right-click **Shared Folders**, click the **All Tasks** tab, and then click **Configure Shadow Copies**.
4. For each volume on the destination server that had shadow copies configured on the

source server, right-click the volume, select **Configure Shadow Copies**, click **Settings**, and verify that the location and size of shadow copy storage match the settings from the source server.

5. Click **Schedule**, and verify that these details for the snapshot creation task match the settings from the source server.

Deduplication

Use the following section to migrate Deduplication.

Migrating Deduplication from Windows Server 2012 to Windows Server 2012

All configuration information needed for migration is included on the deduplicated volume.

If a disk is physically moved, or if a deduplicated volume is restored from a backup onto a different Windows Server 2012 computer, install the Deduplication role service using Server Manager on the new computer. If the Deduplication role service is not installed on the new server, only normal non-deduplicated files will be accessible. Once a volume has been mounted, the deduplication filter will detect that the volume is deduplicated and will redirect input/output requests appropriately.

Note

Any previous custom deduplication job schedules that were created using Task Scheduler must be created again on the new computer using Task Scheduler.

Migrating SIS from Windows Storage Server 2008 to Windows Server 2012

Volumes that have been created and optimized using the down-level Windows Storage Server version of deduplication, Single Instance Storage (SIS), should not be enabled for data deduplication. Microsoft recommends that SIS be removed from the volume by using SISAdmin.exe within Windows Storage Server to remove SIS or by copying the data to a different volume that is not running SIS prior to migrating the volume.

Windows Server 2012 supports reading and writing to SIS-controlled volumes, but you cannot continue to SIS files using Windows Server 2012. You can install the SIS filter driver on Windows Server 2012 by installing the SIS-Limited feature using the following command syntax:

```
dism /online /enable-feature:SIS-Limited
```

The SIS filter driver can be loaded so that you can read SIS-controlled volumes and the data can be copied to a non-SIS controlled volume so that data deduplication can be installed on the volume. Note that Windows Server 2012 does not support sisadmin.exe and cannot be used to remove SIS from a volume.

1. You should remove SIS from your volumes before installing the Windows Server 2012 data deduplication feature. (This process is also known as un-SIS.)
2. Do not restore SIS links from a backup to a Windows Server 2012 deduplicated volume.

3. Restoring SIS volumes to Windows Server 2012 is supported if you load the SIS-Limited filter.

Migrating SIS volumes

You have several options when it comes to migrating Windows Storage Server 2008 volumes to Windows Server 2012 to take advantage of the new Data Deduplication feature.

You can migrate your existing SIS-installed Windows Storage Server 2008 volumes to Windows Server 2012, however, migration is not automatic. Single Instance Storage (SIS) and data deduplication are mutually-exclusive technologies.

Caution

You will need to open the volumes in Windows Storage Server 2008 first, un-SIS them, and then uninstall SIS before migrating to Windows Server 2012 as described in the procedures below.

To unSIS a Windows Storage Server 2008 or 2008 R2 SIS volume type **sisadmin.exe [/m <server>] [/u <volumes>]** where:

/m <server> - Shifts the focus of the command line to a remote server. If the /m option is not specified, the command line is applied to the local server. <server> can be expressed as a host name, fully qualified domain name (FQDN), or as an IP address.

/u <volumes> - Is used to un-SIS a volume (that is, to restore all file copies, and remove reparse points).

For each command option that uses <volumes> as a parameter, <volumes> represents a space-delimited list of volume names (for example: d: e: f: g:). For example:

To unSIS or remove SIS entirely from the F: volume of a remote server using the IP address of the server, you might use the following command: **sisadmin.exe /m 192.168.1.50 /u F:**

See Also

Migrate File and Storage Services to Windows Server 2012

File and Storage Services: Prepare to Migrate

File and Storage Services: Verify the Migration

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Verify the Migration

Verify that the migration was successful. Follow the appropriate verification steps based on the File and Storage Services role services that have been migrated.

The following overview describes the steps to verify the migration.

Verify the File Services migration

Perform the following tasks to verify the File and Storage Services role migration.

- Verify the File Services migration (only if running Windows Server 2008 R2 or Windows Server 2012)
- Verify migration of local users and groups
- Verify data and shared folder migration
- Verify the migration of DFS Namespaces
- Verify the configuration on other computers
- Verify the File Server Resource Manager migration

Verify migration of BranchCache for Network File Services server key

Perform this step only if the source server is running Windows Server 2008 R2 or Windows Server 2012:

Verify that the server key was migrated correctly by checking the key value, and ensure that the key values are identical on source server and the destination server, as shown in the following example:

```
Key: HKLM\Software\Microsoft\WindowsNT\CurrentVersion\PeerDist\SecurityManager\Restricted
Value: Seed
```

Verify migration of local users and groups

Check that all the local users and groups you expected to migrate are present on the destination server by comparing the list of users and groups on the Local Users and Groups snap-in on the source server with the list on the destination server.

To open the Local Users and Groups

1. In Server Manager, click **Tools**, and then click **Computer Management**.

Alternatively, you can compare the list of users and groups on the source server and destination server by typing **net** commands in a Command Prompt window.

- To get the list of all local users and save it in a text file, type the following command:

```
net user > localusers.txt
```

- To get the list of all local groups and save it in a text file, type the following command:

```
net localgroup > localgroups.txt
```

Verify data and shared folder migration

1. Check that all the data you expected to migrate are present at the correct location on the destination server and that they have the correct permissions associated with them.

   To list files and folders with their permissions, type the following command in a Command Prompt window or in a Windows PowerShell session opened with elevated user rights:

   ```
   icacls <path>
   ```

2. Verify that all the expected shared folders have migrated and that they have the correct permissions associated with them. To list all shared folders and their permissions, type the following command in a Windows PowerShell session opened with elevated user rights:

   ```
   gwmi win32_share | %{net share $_.name}
   ```

Verify the migration of DFS Namespaces

The procedure that you use to verify the migration of DFS Namespaces depends on whether your namespaces are stand-alone or domain-based.

To verify the migration of a stand-alone namespace

1. Open DFS Management on the destination server.
2. Right-click **Namespaces**, or click the **Action** menu.
3. Click **Add Namespaces to Display**.
4. Type the name of destination server, and then click the **Show Namespaces** button.
5. Select the namespace that you migrated, and then click **OK**.
6. In the namespaces tree, click the namespace that you migrated.
7. Click the **Namespace** tab, and check that all the namespace links are present.
8. Click the **Namespace server** tab, and check that the destination server is listed.
9. Right-click the destination server name, and then click **Open in Windows Explorer**. All namespace links should be visible in the new window.
10. Using DFSUtil.exe on the destination server, type the following command for each stand-alone namespace:

    ```
    Dfsutil.exe root \\DestinationServer\Namespace
    ```

    The information displayed should contain the destination server and all the namespace links.

To verify the migration of a domain-based namespace

1. Open DFS Management, and then right-click **Namespaces** or click the **Action** menu.

2. Click **Add Namespaces to Display**.
3. Type the name of the domain where the namespace is located, and then click the **Show Namespaces** button.

   Select the namespace that you migrated, and click **OK**.
4. In the namespaces tree, click the namespace that you migrated.
5. Click the **Namespace** tab, and check that all the namespace links are present.
6. Click the **Namespace** server tab, and check that all the namespace servers are listed.
7. Right-click the destination server name, and then click **Open in Windows Explorer**. All namespace links should be visible in the new window.
8. Using DFSUtil.exe on the destination server, type the following command in a command Prompt window, where *domain**namespace* is the name of the appropriate domain and namespace that you migrated.

   ```
   Dfsutil.exe root <\\Domain\Namespace>
   ```

   The information displayed should contain all namespace servers and namespace links.

Verify the configuration on other computers

To verify that File and Storage Services migration completed successfully on other computers, you must test the configuration on the client computers in your enterprise.

To verify DFS Namespaces on a client computer

1. Log on to a client computer with the credentials of a user who has access to the migrated namespace.
2. Verify that you can access the namespace by using File Explorer, a command prompt window, or another application, by entering the same name that you used before the migration.

Verify the File Server Resource Manager migration

Follow these steps to ensure that File Server Resource Manager migrated:

1. If any custom actions are configured for quota notification or file management tasks, the user should ensure that the folders that contain the executable files configured for the actions and the working folders have the correct access control lists.
2. As a best practice, ensure that all e-mail message text for notifications, reports, and so on migrated correctly.
3. Administrators should send a test e-mail message through the File Server Resource Manager console to verify that the Simple Mail Transfer Protocol (SMTP) server is configured correctly for the destination server.
4. Ensure that expiration folders that are used by File Management Tasks are reachable on the destination server.
5. Ensure that executable files that are used by custom actions (such as quota notifications and file management tasks) are accessible or executable on the destination server.

See Also

Migrate File and Storage Services to Windows Server 2012

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Post-Migration Tasks

This topic explains how to complete the migration if it was successful, and how to roll back or troubleshoot the migration if it failed.

Completing the migration

After you verify the migration, retire the source server.

Retire File and Storage Services on the source server

After you complete and verify the migration, the source server can be shut down or disconnected from the network.

Remove DFS Namespaces from the source server

The procedure you use to remove DFS Namespaces from the source server depends on whether the namespaces are stand-alone or domain-based. If you want to remove the namespace from the source server, you must use **DFSUtil.exe**.

Note

By default, clients cache the list of namespace servers for 300 seconds (five minutes), so we recommend that you do not run the **DFSUtil.exe remove** command within five minutes of completing verification of the DFS namespace migration. During migration, clients have only the temporary server in the cache of namespace servers. Waiting five minutes after you add the destination server to the namespace allows clients to list the destination server in their cache.

To remove stand-alone namespaces

1. Open a Command Prompt window on the destination server.

2. Type the following code, and then press **Enter**.

```
Dfsutil.exe root remove <\\SourceServer\Namespace>
```

To remove domain-based namespaces with one namespace server

1. On the destination server, open a Command Prompt window.
2. Type the following, and then press **Enter**.

```
DFSUtil.exe target remove <\\TemporaryServer\Namespace>
```

Notes

This procedure applies only if a temporary server was added to the namespace for migration purposes.

For domain-based namespaces with more than one namespace server, no additional actions are required.

Restoring File and Storage Services in the event of migration failure

The following sections describe how to restore the File and Storage Services server role in the event of migration failure.

Roll back DFS Namespaces

The steps that you perform to roll back DFS Namespaces depend on whether the namespaces are stand-alone or domain-based, and whether a temporary namespace was created during the migration process.

To roll back DFS Namespaces (do one of the following)

1. For stand-alone namespaces, no action is required other than migrating the identity back to the source server.
2. For domain-based namespaces with greater than one namespace server, or if a temporary server was added to a namespace that initially had only one namespace server, do the following:
   a. Remove destination server from the namespace.
   b. Migrate identity and shared folder information to the source server.
   c. Add the source server to namespace.
3. For domain-based namespaces with only one namespace server, where no temporary namespace server was added during migration, do the following:
   a. Migrate identity and shared folder information to source server.
   b. Verify the export file for the namespace that was created during migration is still available.
   c. Delete the namespace.

d. Create the namespace on the source server.
e. Import the namespace configuration from the export file created during the migration.
f. Manually reset delegation permissions to the namespace.

Note

Another option to migrate domain-based namespaces with one namespace server is to temporarily add a second namespace server before the migration, and then remove the temporary server after the migration.

Roll back data and shared folders

If no changes have been made to migrated files, folders, and shared folders on the destination server and this data has not been deleted from the source server, no additional steps to roll back data and shared folders are required.

If the migrated files, folders, or shared folders may have been modified on the destination server by the administrators or users, perform the following steps to synchronize the changes from the destination server back to the source server:

1. Type the following code in a Command Prompt window to copy the updated migrated data (files and folders) from the destination server back to the source server:

   ```
   robocopy <copy from path> <copy to path> /E
   ```

   This command can be executed on the source server or on the destination server, and it will recursively copy updated data. Type `robocopy /?` in a Command Prompt window for additional copy options, including options to copy file and folder permissions.

   Caution

   Permissions that you set for non-default local users and groups will not copy properly and need to be created manually.

2. Compare the lists of shared folders and their permissions on the source server and destination server and manually synchronize any changes.

   To list all shared folders and their permissions, type the following command in a Windows PowerShell session that has been opened with elevated user rights:

   ```
   gwmi win32_share | %{net share $_.name}
   ```

Roll back migration on the other computers in the enterprise

If the migration failed, verify that the other computers in the enterprise can access the source server after you roll back the migration data.

Troubleshooting migration issues

Troubleshooting tips include the following:

- For physical migration issues:

When some files are migrated physically and others are copied, there is a chance that the File Server Resource Manager configuration is not synchronized. To remedy this, delete and create new copies of the Quota.md and Datascrn.md files.

- For domain-joined machines:

  If a custom action (quota notification or file management task) fails to execute with an access-denied failure and a corresponding event log, you should remove the custom action and create it on the destination server.

Troubleshoot data migration that does not complete

If the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets run indefinitely without completing, your destination server might not have sufficient disk space or a large enough File Server Resource Manager or NTFS quota limit to allow for data migration to finish. To determine whether insufficient disk space is preventing the data send-receive process from completing, do the following on the destination server.

1. Open %*localappdata*%/Svrmig/Log/SetupAct.log.
2. Review the most recent log entries. If the following exception occurs, your destination server has insufficient disk space or File Server Resource Manager or NTFS quota limits to complete data migration.

   ```
   Win32Exception: unable to write to FileStream: There is not enough space on the disk.
   ```

 **To resolve this issue, do the following:**

1. Press **Ctrl+C** to cancel **Send-SmigServerData** and **Receive-SmigServerData** on both source and destination servers.
2. Check for sufficient disk space on the destination server's hard disk drive. If the destination server's hard disk drive has insufficient space, do one of the following.
   - Clear additional space.
   - Identify a different hard disk drive that has sufficient space.
3. If the destination server's hard disk drive, the destination path, or any folders that contain the destination path have an File Server Resource Manager or NTFS quota enabled, and the quota limit does not allow for sufficient disk space to migrate data, do one of the following.
   - Increase the quota limit to set sufficient disk space to migrate the data. For more information about FSRM quota management, see one of the following.
     - Quota Management (http://go.microsoft.com/fwlink/?LinkId=154277) for Windows Server 2008, Windows Server 2008 R2, and Windows Server 2012
     - Quota Management (http://go.microsoft.com/fwlink/?LinkId=154241) for Windows Server 2003 R2

For more information about NTFS quota management, see one of the following.

- Setting Disk Quotas (http://go.microsoft.com/fwlink/?LinkId=154243) for Windows Server 2008, Windows Server 2008 R2, and Windows Server 2012
- Enable disk quotas (http://go.microsoft.com/fwlink/?LinkId=154245) for Windows Server 2003 and Windows Server 2003 R2

- Identify a different hard disk drive that already has sufficient space and File Server Resource Manager or NTFS quota limits.

4. Run the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets again, specifying a destination path that has sufficient disk space, and large enough File Server Resource Manager or NTFS quota limits, if applicable.

Troubleshoot data migration connectivity

If the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets cannot establish connectivity, verify the following conditions and then try again:

1. In the **Send-SmigServerData** command on the source server, the *ComputerName* parameter correctly specifies the name of the destination server.
2. The **Receive-SmigServerData** and **Send-SmigServerData** commands are entered on the destination server and the source server respectively within five minutes of one another. This is the default maximum connection time-out for **Send-SmigServerData** and **Receive-SmigServerData**. You can change the maximum connection time-out for the **Send-SmigServerData** and **Receive-SmigServerData** cmdlets by modifying the following user-defined registry key on the source server and destination server.

   Key: HKEY_Local_Machine\Software\Microsoft\ServerMigration

   Value: MaxConnectionTime (REG_DWORD)

   Data: Between 1 and 3600 (represents the connection time-out in seconds). If a value larger than 3600 is specified, 3600 seconds is used as the maximum connection time-out.

   For information about how to create a Windows Registry key, see Add a Registry Key (http://go.microsoft.com/fwlink/?LinkId=147298) on the Microsoft Web site.
3. The same password is entered on the source server and destination server.
4. The source server and destination server are available on the same subnet:
   a. On the destination server, in a command prompt window, type `ipconfig` and note the subnet mask value.
   b. On the source server, in a command prompt window, type `ipconfig` and note the subnet mask value.
   c. Ensure that the subnet mask values are the same on the source server and destination server.
5. Port 7000 is open on the source and destination server, and they are not in use by another application.
   a. To check if port 7000 is open, in a Command Prompt window, enter the command:

      ```
      netsh firewall show portopening
      ```

If port 7000 is not in the list, follow the instructions in File and Storage Services: Appendix A: Optional Procedures to open port 7000.

b. If port 7000 is open, type the following command to check if port 7000 is being used by another application:

```
netstat
```

- In the Local Address column, you will see <IP Address>:<port number>.
- If port 7000 is in the list, it is being used by another application.

Troubleshoot unexpected Windows PowerShell session closure

If a migration cmdlet fails, and the Windows PowerShell session closes unexpectedly with an access violation error message, look for a message similar to the following example in the %*localappdata*%\SvrMig\Logs\setuperr.log file.

```
FatalError [0x090001] PANTHR Exception (code 0xC0000005: ACCESS_VIOLATION) occurred at
0x000007FEEDE9E050 in C:\Windows\system32\migwiz\unbcl.dll (+000000000008E050). Minidump
attached (317793 bytes).
```

This failure occurs when the server cannot contact domain controllers that are associated with domain users or groups who are members of local groups, or who have rights to files or shares that are being migrated. When this happens, each domain user or group is displayed in the GUI as an unresolved security identifier (SID). An example of a SID is **S-1-5-21-1579938362-1064596589-3161144252-1006**.

To prevent this problem, verify that required domain controllers or global catalog servers are running, and that network connectivity allows communication between both source and destination servers and required domain controllers or global catalog servers. Then, run the cmdlets again.

If connections between either the source or destination servers and the domain controllers or global catalog servers cannot be restored, do the following.

1. Before you run **Export-SmigServerSetting**, **Import-SmigServerSetting** or **Get-SmigServerFeature** again, remove all unresolved domain users or groups who are members of local groups from the server on which you are running the cmdlet.
2. Before you run **Send-SmigServerData** or **Receive-SmigServerData** again, remove all unresolved domain users or groups who have user rights to files, folders, or shares on the migration source server.

Locate the deployment log file

The Windows Server Migration Tools deployment log file is located at %windir%\Logs\SmigDeploy.log. Additional Windows Server Migration Tools log files are created at the following locations:

- `%windir%\Logs\ServerMigration.log`

- On Windows Server 2008, Windows Server 2008 R2, and Windows Server 2012: `%localappdata%\SvrMig\Log`
- On Windows Server 2003: `%userprofile%\Local Settings\Application Data\SvrMig\Log`

If migration log files are not created in the preceding locations, ServerMigration.log and SmigDeploy.log are created in `%temp%`, and other logs are created in `%windir%\System32`.

View the content of Windows Server Migration Tools result objects

All Windows Server Migration Tools cmdlets return results as objects. You can save result objects, and query them for more information about the settings and data that were migrated. You can also use result objects as input for other Windows PowerShell commands and scripts.

Result object descriptions

The **Import-SmigServerSetting** and **Export-SmigServerSetting** cmdlets in Windows Server Migration Tools return results in a list of **MigrationResult** objects. Each **MigrationResult** object contains information about the data or setting that the cmdlet processes, the result of the operation, and any related error or warning messages. The following table describes the properties of a **MigrationResult** object.

Property Name	Type	Definition
ItemType	Enum	The type of item being migrated. Values include **General**, **WindowsFeatureInstallation**, **WindowsFeature**, and **OSSetting**.
ID	String	The ID of the migrated item. Examples of values include **Local User**, **Local Group**, and **DHCP**.
Success	Boolean	The value **True** is displayed if the migration was successful; otherwise, **False** is displayed.
DetailsList	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.

Send-SmigServerData and **Receive-SmigServerData** cmdlets return results in a list of **MigrationDataResult** objects. Each **MigrationDataResult** object contains information about the data or shared folder that the cmdlet processes, the result of the operation, any error or warning messages, and other related information. The following table describes the properties of a **MigrationDataResult** object.

Property Name	Type	Definition
ItemType	Enum	The type of migrated item. Values include **File**, **Folder**, **Share**, and **Encrypted File**.
SourceLocation	String	The source location of the item, shown as a path name.
DestinationLocation	String	The destination location of the item shown as a path name.
Success	Boolean	The value **True** is displayed if the migration was successful; otherwise, **False** is displayed.
Size	Integer	The item size, in bytes.
ErrorDetails	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.
Error	Enum	Errors enumeration for errors that occurred.
WarningMessageList	List <String>	A list of warning messages.

The following table describes the properties of objects within the **MigrationResultDetails** object that are common to **MigrationResult** and **MigrationDataResult** objects.

Property name	Type	Definition
FeatureId	String	The name of the migration setting that is related to the item. Examples of values include **IPConfig** and **DNS**. This property is empty for data migration.
Messages	List <String>	A list of detailed event messages.
DetailCode	Integer	The error or warning code associated with each event message.
Severity	Enum	The severity of an event, if

Property name	Type	Definition
		events occurred. Examples of values include **Information**, **Error**, and **Warning**.
Title	String	Title of the result object. Examples of values include the physical address of the network adapter for IP configuration, or the user name for local user migration.

Examples

The following examples show how to store the list of the result objects in a variable, and then use the variable in a query to return the content of result objects after the migration is complete.

To store a list of result objects as a variable for queries

1. To run a cmdlet and save the result in variable, type a command in the following format, and then press **Enter**.

   $*VariableName* **=** $(*Cmdlet*)

   The following is an example.

   $ImportResult = $(Import-SmigServerSetting -FeatureId DHCP -User all -Group -Path D:\rmt\DemoStore -force -Verbose)

   This command runs the **Import-SmigServerSetting** cmdlet with several parameters specified, and then saves result objects in the variable **ImportResult**.

2. After the **Import-SmigServerSetting** cmdlet has completed its operations, return the information contained in the result object by typing a command in the following format, and then pressing **Enter**.

   $*VariableName*

   In the following example, the variable is named **ImportResult**.

   $*ImportResult*

   This command returns information contained in the result objects that were returned by **Import-SmigServerSetting** in the example shown in step 1. The following is an example of the output that is displayed by calling the **ImportResult** variable:

```
            ItemType  ID                                Success
DetailsList
            --------  --                                -------
-----------
          OSSetting  Local User                          True
```

```
{Local User, Loc...
          OSSetting  Local Group                          True
{Local Group, Lo...
     WindowsFeature  DHCP                                 True
{}
```

Each line of the preceding example is a migration result for an item that was migrated by using the **Import-SmigServerSetting** cmdlet. The column heading names are properties of **MigrationResult** objects. You can incorporate these properties into another command to return greater detail about result objects, as shown by the examples that follow in steps 3 and forward.

3. To display a specific property for all result objects in the list, type a command in the following format, and then press **Enter**.

   $<VariableName>| Select-Object -ExpandProperty <PropertyName>

   The following is an example.

   $importResult | Select-Object -ExpandProperty DetailsList

4. You can run more advanced queries to analyze result objects by using Windows PowerShell cmdlets. The following are examples:

   - The following command returns only those details of result objects that have the ID **Local User**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList

   - The following command returns only those details of result objects with an ID of **Local User** that have a message severity equal to **Warning**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Severity -eq "Warning") {$_} }

   - The following command returns only the details of result objects with an ID of **Local User** that also have the title **Remote Desktop Users**.

     $ImportResult | Where-Object { $_.ID -eq "Local Group" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Title -eq "Remote DesktopUsers") {$_} }

More information about querying results

For more information about the cmdlets that are used in the preceding examples, see the following additional resources.

- Where-Object (http://go.microsoft.com/fwlink/?LinkId=134853).
- Select-Object (http://go.microsoft.com/fwlink/?LinkId=134858).
- ForEach-Object (http://go.microsoft.com/fwlink/?LinkId=134860)

For more information about Windows PowerShell scripting techniques, see What Can I Do With Windows PowerShell? - Scripting Techniques on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134862).

See Also

Migrate File and Storage Services to Windows Server 2012

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Appendix A: Optional Procedures

Opening ports in Windows Firewall

The following instructions are for opening ports in Windows® Firewall. If you have a non-Microsoft firewall installed, consult the guide for that firewall about how to open ports. Opening ports in Windows Firewall can be done through the command interface.

Important
Opening ports in your firewall can leave your server exposed to malicious attacks. Make sure that you understand firewall systems before you open ports.

To open Windows Firewall ports by using the command line (do one of the following):

1. Open a Command Prompt window with elevated user rights, type the following, and then press ENTER.
   - On computers that are running Windows Server 2003, type:

     ```
     netsh firewall add allowedprogram
     program=%windir%\System32\WindowsPowerShell\v1.0\powershell.ex
     e name="ServerMigration" mode=ENABLE
     ```
   - On computers that are running Windows Server® 2008, Windows Server 2008 R2, or Windows Server 2012, type the following commands, in order, pressing ENTER after each command.

     i.

ii.

2. If you have changed the default behavior of Windows Firewall to block all outbound traffic on computers that are running Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012, you must explicitly allow outbound traffic on UDP port 7000. To do this, open a Command Prompt window with elevated user rights, type the following, and then press ENTER.

```
netsh advfirewall firewall add rule name=ServerMigration(UDP-Out) dir=out program=%windir%\System32\WindowsPowerShell\v1.0\powershell.exe action=allow protocol=UDP localport=7000
```

Closing ports in Windows Firewall

As a best practice, we recommend that you close Windows Firewall ports after the data transfer operation is completed.

To close Windows Firewall ports by using the command line

- Do one of the following:
  - On computers that are running Windows® Server 2003, open a Command Prompt window, type the following, and then press ENTER.

    ```
    netsh firewall delete allowedprogram program=%windir%\System32\WindowsPowerShell\v1.0\powershell.exe
    ```
  - On computers that are running Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012, open a command prompt window with elevated user rights, type the following two commands. Press ENTER after each.

    ```
    netsh advfirewall firewall delete rule name=ServerMigration(TCP-In)
    ```

    ```
    netsh advfirewall firewall delete rule name=ServerMigration(UDP-Out)
    ```

Detect reparse points and hard links

The following commands can be used to detect reparse points and mounted volumes in any folder and its subfolders. Open a Command Prompt window, type the following commands to detect reparse points, in which **D:\Test** represents the hard disk drive and folder that you want to search, and then press ENTER.

```
dir D:\Test\* /S /A:L
```

The option **/A:L** specifies that only reparse points need to be enumerated. The output is similar to the following:

```
Volume in drive D has no label.
Volume Serial Number is 3AE4-E412

Directory of D:\Test\Links

10/07/2008  03:44 PM    <JUNCTION>     JunctionMSIT [d:\test\targets\msit]
10/07/2008  03:42 PM    <SYMLINK>      LinkMSIT [d:\test\targets\msit]
10/07/2008  03:41 PM    <SYMLINKD>     SymLinkMSIT [d:\test\targets\msit]
1 File(s)              0 bytes

Directory of D:\Test\Targets

10/07/2008  05:35 PM    <JUNCTION>     Volume [\??\Volume{0674413f-760d-11dd-beb3-
806e6f6e6963}\]
0 File(s)              0 bytes

Total Files Listed:
1 File(s)              0 bytes
3 Dir(s)  17,918,840,832 bytes free
```

To enumerate hard links on a file on Windows Server 2008 R2, or Windows Server 2012 open a command prompt window with elevated user rights, type the following, and then press ENTER.

```
fsutil hardlink list D:\Test\File.txt
```

To enumerate hard links on all files in a folder on Windows Server 2008 R2 or Windows Server 2012, run the following command in a Windows PowerShell session that has been opened with elevated user rights:

```
Get-ChildItem D:\* | %{'Links for: ' + $_.FullName; fsutil hardlink list $_.FullName; ""}
```

For more information about enumerating hard links on computers that are running Windows Server 2003 or Windows Server 2008, see FindFirstFileNameW Function (http://go.microsoft.com/fwlink/?LinkId=147392) on MSDN.

Migrated and non-migrated attributes for local users and groups

For more information about the attributes of local users and groups that can be migrated, see the Local User and Group Migration Guide (http://go.microsoft.com/fwlink/?LinkID=128751) on the Microsoft Web site.

See Also

Migrate File and Storage Services to Windows Server 2012

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix B: Migration Data Collection Worksheets

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Appendix B: Migration Data Collection Worksheets

SMB data collection worksheet

Use this server message block (SMB) data collection worksheet to record data for SMB policies that are set on the source server.

#	Source Server Essential Settings	Setting Identification
01	Idle time The setting name is: **Microsoft network server: Amount of idle time required before suspending a session**.	Idle time (in minutes): _____ Group or Local Policy: _____
02	S4USelf The setting name is: **Microsoft network server: Attempt S4USelf to obtain claim information**.	Claim information: __ Default __ Enabled or __ Disabled Group or Local Policy: _____
03	Sign (always) The setting name is: **Microsoft network server: Digitally sign communications (always)**.	Sign always: __ Enabled or __ Disabled Group or Local Policy: _____
04	Sign (if client agrees) The setting name is: **Microsoft network server: Digitally sign communications (if client**	Sign if client agrees: __ Enabled or __ Disabled Group or Local Policy:

#	Source Server Essential Settings	Setting Identification
	agrees).	_____
05	Disconnect when logon hours expire The setting name is: **Microsoft network server: Disconnect clients when logon hours expire**.	Disconnect: __ Enabled or __ Disabled Group or Local Policy: _____

BranchCache data collection worksheet

Use this BranchCache data collection worksheet to record data for the BranchCache policies that are set on the source server.

#	Source Server Essential Settings	Setting Identification
01	BranchCache The setting name is: **Hash Publication for BranchCache**.	BranchCache: __ Not configured, __ Enabled, or __ Disabled Group or Local Policy: _____
	BranchCache The setting name is: **Hash Version support for BranchCache**.	BranchCache: __ Not configured, __ Enabled, or __ Disabled Group or Local Policy: _____

See Also

Migrate File and Storage Services to Windows Server 2012

File and Storage Services: Prepare to Migrate

File and Storage Services: Migrate the File and Storage Services Role

File and Storage Services: Verify the Migration

File and Storage Services: Post-Migration Tasks

File and Storage Services: Appendix A: Optional Procedures

File and Storage Services: Appendix C: Migrate iSCSI Software Target

File and Storage Services: Appendix C: Migrate iSCSI Software Target

This document describes how to migrate Microsoft® iSCSI Software Target 3.2 or 3.3 settings and data from an existing Windows Storage Server 2008 or Windows Storage Server 2008 R2 computer to a destination server that is running the ISCSI Target Server role service that is included with Windows Server® 2012 and Windows Storage Server® 2012.

The naming for iSCSI Software Target has changed. To reduce the potential for confusion, in the context of this document, any naming that refers to “iSCSI Software Target”, refers to prior product versions installed on Windows Storage Server 2008 and Windows Storage Server 2008 R2, which are source servers. By contrast, any naming that refers to “iSCSI Target Server” refers to the new role service included with Windows Server 2012 and Windows Storage Server 2012, which are destination servers.

Note
This document only contains iSCSI-specific migration information. For generic information, such as the use of Windows Server Migration Tools, refer to the application section in the main File and Storage Services Migration Guide.

See Also

iSCSI SoftwareTarget Migration Overview

Prepare to Migrate iSCSI Software Target

Migrate iSCSI Software Target

Verify the iSCSI Software Target Migration

Troubleshoot the iSCSI Software Target Migration

Roll Back a Failed iSCSI Software Target Migration

iSCSI SoftwareTarget Migration Overview

Insert introduction here.

Migration overview

This section describes the high-level migration process, which involves harvesting configuration settings from the source, moving the virtual disks from the source server to the destination server, and restoring the configuration settings.

Migration process

This section describes the high-level migration process.

Migration planning

The migration planning phase involves gathering the following information:

- Are the source server and destination server are configured in a cluster?
- If the servers are configured in a cluster, what are the virtual computer objects or client access points that contain the iSCSI target resources?
- Is the storage system of the destination server capable and configured appropriately to host the virtual disks of the source server, and does it have appropriate space to store the volume snapshots?
- Are there any iSCSI initiators that have a critical dependency on iSCSI targets for the duration of the migration process (such as a computer that uses iSCSI boot nodes, or clusters that use shared storage)?
- Are there any IP address or portal settings that are unique to the source server that need to be accounted for (such as IP addresses that are known to the firmware of devices)?
- Are there any iSNS settings that need to be manually recorded and migrated?
- Are there any virtual disks surfaced as local disks that might need to be exposed?

Preparing to migrate

The preparation to migrate data from the source server to the destination server involves the following steps:

1. If the destination server will have a clustered configuration, install the Failover Clustering feature and form a cluster before performing the migration.
2. If the destination server will have a clustered configuration, create a number of cluster resource groups with client access points and cluster disks as appropriate to replicate the existing configuration. If possible, use the same resource group names for the source clusters and the destination clusters.
3. Install the iSCSI Target Server role service on the destination server.
4. Disconnect all the iSCSI initiators. This step is required to maintain consistent data on the virtual disks while they are being moved.
5. Run the Windows® PowerShell™ script, iSCSITargetSettings.ps1, to capture the existing settings on the source server in an XML file. For a cluster, run the script on each node in the cluster or on each virtual computer object, as appropriate for the scope of the planned migration.

   The Windows PowerShell script displays the virtual disks that are eligible for migration and those that are not (for the snapshot-based reasons discussed previously).

Migration

The actual migration process consists in the following steps:

1. Move the files for all the virtual disks that are eligible for migration from the source server to the destination server. If there are any file path changes, note the source to destination mapping.

2. In a cluster configuration, ensure that the destination path of the file copy is on a cluster disk and that the cluster disk has been assigned to a resource group. Note of the resource group that owns the path.
3. If the file paths have changed between the source and the destination servers, open the settings .xml file in a text editor, and identify the <MigrationDevicePath> tags that need to be changed to reflect the new path.
4. In a cluster configuration, if the file path or the resource group name have changed between the source server and the destination server, open the settings .xml file in a text editor, and identify the <MigrationResourceGroup> tags that need to be changed to reflect the new resource group.
5. Run the Windows PowerShell script, iSCSITargetSettings.ps1, to import the settings to the destination server. In a cluster configuration, the destination server can be specified as a cluster node or as a virtual computer object. The cluster node or virtual computer object must be the owner of the resource group that is indicated in the settings .xml file.
6. If there are snapshot storage settings relevant to the new configuration, apply those settings manually.
7. If there are virtual disks that need to be surfaced as local disks, perform those actions.
8. If there are any iSNS settings that are relevant to the new configuration, apply those settings manually.
9. If there are any iSCSI target portal settings that are relevant to the new configuration, apply those settings manually.
10. If there are any iSCSI initiators that are configured to authenticate by using CHAP and Reverse CHAP, manually restore those settings.

Verification

The verification process for the migration involves the following steps:

1. Validate the iSCSI target portal settings by opening a Command Prompt window and typing **netstat.exe –nao | findstr 3260**. (This assumes that the default TCP port for the iSCSI protocol 3260 is used). Alternatively, type **Get-WmiObject –Namespace root\wmi –Class WT_Portal** to cross-check the results.
2. Inspect the iSCSI Target Server configuration by using the Windows PowerShell cmdlet, **Get-IScsiServerTarget**
3. Inspect the iSCSI virtual disk configuration by using the Windows PowerShell cmdlet, **Get-IScsiVirtualDisk**
4. Validate the configuration for each iSCSI initiator that you expect to use with iSCSI Target Server by using the iscsicpl.exe UI tool or the iscsicli.exe command line tool.

Impact of migration

The migration process does not impact or affect the source server. There are no resources or configuration settings that are altered or deleted as part of the migration process.

No servers in the enterprise, other than the destination servers, will be affected by the migration.

Client computers that are running as iSCSI initiators are expected to be explicitly disconnected during the migration to ensure data integrity. During the migration, the source server will be

unavailable. When the migration process is complete, it is expected that the iSCSI initiators will log on to the destination server without any issues.

The downtime for the iSCSI initiators is expected to be proportionate to the time it takes to move the virtual disk files from the source server to the destination server, plus the time needed to restore the configuration settings and to establish the network identity of the destination server.

Permissions required for migration

Local Administrator permissions are required on the source and the destination server.

If the iSNS server has additional access control policies, permission to alter the iSNS settings are required as appropriate for the iSNS server.

To perform the migration process for the iSCSI initiators, permissions to log on and log off iSCSI sessions are required. For the iSCSI initiator, Local Administrator permissions are required.

For iSCSI initiators that are firmware based, such as a network interface with the option to boot from iSCSI, being at the actual console may be required to configure logon credentials or the network identity of the destination server if the authentication settings (CHAP and Reverse CHAP) have changed.

Estimated time duration

This section detail the various factors that impact how long a migration may take to complete.

Planning

The planning phase is expected to be influenced by the following factors:

- Standalone versus a cluster configuration. A cluster setup may require one to two hours to configure if all the validations are performed.
- Storage configuration. Understanding and configuring a storage array to host potentially huge files requires that you plan the spindle and volume configurations so that they use the tools that are provided by the storage subsystem vendor.
- Network identity. This planning involves understanding if the source server has specially or purposely configured IP addresses, if configuring Level-2 components (such as switches) is required, and if specific DNS or NetBIOS names need to be known to and cached by the iSCSI initiators.

Preparation

The preparation process involves understanding which settings (that are specific to the source server) cannot be automatically migrated, and gathering those settings. For each step in the preparation phase, the mechanism that is used to retrieve the settings depends on which step is applicable and which tool is used to recover those settings.

- Cluster resource group names and configuration. These settings can be gathered from the cluster administration tools and the user interfaces.
- iSCSI target portal configuration. These settings can be gathered by typing the following code at a command prompt: `PS > Get-WmiObject –Namespace root\wmi –Class WT_Portal`

- iSNS Server settings. These settings can be gathered by typing the following code at a command prompt: `PS > Get-WmiObject –Namespace root\wmi –Class WT_ISnsServer`
- CHAP and Reverse CHAP authentication settings. These settings cannot be automatically retrieved because the iSCSI target server does not offer a mechanism to retrieve passwords. These settings have been stored elsewhere in the enterprise, and they need to be retrieved independently.
- Locally mounted virtual disk settings.

Migration

The estimated time for the actual migration process is largely dominated by the time that it takes to move the virtual disk files from the source server to the destination server.

A network-based file copy, using a 1 GB link used at 50% for 1 TB of data, is estimated to take over five hours. Techniques that use a file transfer process involving external media, such as an External Serial Advanced Technology Attachment (eSATA) device, may take less time.

The execution of the Windows PowerShell import script is estimated to take few minutes for approximately 100 resources (with a combination of iSCSI target settings and virtual disk settings).

Verification

The estimated time for the verification is proportionate to the time it takes to reconnect or log on to the iSCSI initiators.

For each iSCSI initiator, the target portal needs to be reconfigured, credentials related to authentication settings must be entered (if required), and the sessions have to be logged on.

The estimated time is 5 to 15 minutes to verify each iSCSI initiator, depending on the process that is being used. iSCSI initiators can be verified through the iscsicpl.exe UI, through the iscsicli.exe command line tool, or through ad hoc Windows Management Instrumentation (WMI)-based scripts).

Supported migration scenarios

This section details both supported and unsupported migration scenarios.

Supported operating systems

The versions of operating systems that are listed are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are supported.

Migrations between physical operating systems and virtual operating systems are supported.

Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shared resources from a computer that is running Windows Server 2008 in the French system UI language to a computer that is running Windows Server 2012 in the German system UI language.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x64-based	Windows Storage Server 2008, full installation options	Windows Server 2012 and Windows Storage Server 2012	x64-based
x64-based	Windows Storage Server 2008 R2	Windows Server 2012 and Windows Storage Server 2012	x64-based
x64-based	Windows Storage Server 2008 R2	Windows Server 2012 and Windows Storage Server 2012	x64-based

x64-based migrations are supported for Windows Storage Server 2012 and Windows Server 2012. All editions of Windows Storage Server 2008 R2 and Windows Server 2008 R2 are x64-based.

x86-based migrations are not supported because Windows Storage Server 2012 is not offered in the x86 platform.

Note
The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Supported role configurations

This migration guide is applicable to standalone and clustered configurations, with certain limitations.

The following general restrictions are applicable to all the supported configurations:

- Authentication settings for iSCSI initiators that use CHAP and Reverse CHAP settings are not automatically migrated.
- Snapshot storage settings for each virtual disk in the configuration are not automatically migrated.
- Configuration settings for virtual disks that are derived from snapshots are not automatically migrated.
- For clustered configurations, the migration process includes iSCSI target settings that are scoped to the virtual computer bject, to a cluster node, or to the cluster node that owns the code cluster group.
- For clustered configurations, the migration of resource groups, network name resources, IP addresses, and cluster disks that are associated with resource groups is outside of scope for this guide, and it needs to be performed independently as a preliminary step.
- iSCSI Naming Services (iSNS) settings for ISCSI Software Target are not automatically migrated.

- iSCSI target portal settings (such as IP addresses that are used by the iSCSI target service to listen for incoming network connections) are not automatically migrated
- The schedule for snapshots of virtual disks is not migrated.

The following configurations are supported:

- Migration from a standalone configuration to standalone configuration
- Migration from a clustered configuration to a standalone configuration (with the restrictions listed previously regarding the scope of the settings).
- Migration from a clustered configuration to a clustered configuration (with the restrictions listed previously regarding the scope of the settings).

Supported role services and features

ISCSI Target Server (as included with Windows Storage Server 2012 and Windows Server 2012) does not have role dependencies or feature dependencies.

It is possible to install ISCSI Target Server with failover clustering, and this configuration is supported with the migration limitations listed previously.

Migrating multiple roles

If you are migrating one clustered configuration to a different clustered configuration, the Failover Clustering feature needs to be migrated or set up prior to migrating iSCSI target settings.

Migration scenarios that are not supported

The following migration scenarios are not supported:

- Migration from Windows Unified Storage Server 2003 R2.
- Migration from a standalone configuration to a clustered configuration. This migration is not supported because there is no default mechanism to associate target and virtual disk settings to resource groups without knowing how the file paths are mapped to the cluster disk and how IP Addresses are mapped to resource groups.
- Snapshots of virtual disks are not automatically migrated. Snapshots are based on a snapshot of the volume that contains the virtual hard disk (VHD) file at the time the snapshot was taken. Their existence and implementation depends on the volume of the computer from which the migration process happens, and it cannot be replicated or exported.
- Snapshot storage settings for virtual disks are not automatically migrated. The snapshot storage settings (such as volume and maximum size per volume) are dependent on the hardware and software configuration of the computer that the settings are being migrate to, and they cannot automatically be migrated. For detailed information about how to manually migrate the snapshot storage settings, see Migrating ISCSI Target.
- The configuration settings of the iSCSI target portal are not automatically migrated. This configuration is based on the IP addresses of the destination server, and those settings cannot be migrated outside the knowledge of the network configuration of the computer that the settings are being migrate to. For detailed information about how to manually configure the portal settings, see Migrating ISCSI Target.

- iSNS settings are not automatically migrated. The iSNS settings are based on the network infrastructure and configuration of the destination server, and those settings cannot be migrated outside the knowledge of the network configuration of the computer that the settings are being migrated to. For detailed information about how to manually configure iSNS settings, see Migrating ISCSI Target.
- Settings for virtual disks that are surfaced as local disks on the source server are not automatically migrated. The ability to surface a disk locally is expected to be a temporary operation that can be replicated if. For detailed information about how to configure settings for virtual disks that are to be surfaced as local disks, see Migrating ISCSI Target.
- The schedule for snapshots of virtual disks is not migrated. Those settings must be manually discovered and replicated from the source to the destination server.

Prepare to Migrate iSCSI Software Target

This topic discusses the tasks that are necessary before you start the migration process. The first step is to install the Windows Server Migration Tools. For more information, see File and Storage Services: Prepare to Migrate.

Prepare the destination server

The destination server is a computer that is configured and shipped by an OEM with Windows Storage Server 2012 pre-installed, or that is running Windows Server 2012.

ISCSI Target Server hardware requirements for the destination server are as follows:

- The amount of free disk space on the destination server must be sufficient to host the iSCSI virtual disk from the source server with adequate room for the snapshot storage.
- For clustered configurations, the resource groups that are created in the destination server must have associated cluster disks with adequate free space to host the iSCSI virtual disk from the source server.
- The destination server must have one or more network interfaces to be utilized for the iSCSI network traffic.

Installing the Failover Cluster feature in Windows Storage Server 2012 or Windows Server 2012 is required if the source server was configured with failover clusters. For more information, see the Failover Clustering Failover Clustering (http://technet.microsoft.com/en-us/library/hh831579.aspx).

Backup the source server

Before you start migration, as a best practice, it is recommended that you back up the source server. For more information, see Windows Server Backup (http://technet.microsoft.com/en-us/library/cc770757.aspx).

Prepare the source server

The following are tasks that are performed on the source server.

Cluster resource group configuration

Use the following steps to obtain the cluster resource groups:

1. Gather the resource groups that have iSCSI Software Target resources by using the following Windows PowerShell command:

```
PS > Import-Module FailoverClusters
PS > $iSCSITargetResources = Get-ClusterResource | Where-Object
{ ( $_.ResourceType.Name -eq "Host" ) –or ($_.ResourceType.Name
-eq "WTDisk") }
PS > $iSCSITargetResources
```

2. From the cluster resources obtained in the previous step, gather the cluster disk dependencies by using the following Windows PowerShell command:

```
PS > $Dependencies = &{ $iSCSITargetResources | Get-
ClusterResourceDependency  }
PS > $Dependencies
```

If the source server is running Windows Storage Server 2008, the following steps can be followed to gather the equivalent information.

1. Gather the iSCSI Software Target resources, and then gather the groups by using the following Windows PowerShell command:

```
PS > $iSCSITargetResources = Get-WmiObject -NameSpace
root\mscluster -Authentication PacketPrivacy -Class
MsCluster_Resource -Filter "Type = `"WTDisk`" or Type =
`"Host`""
PS > $iSCSITargetResources
PS > $Groups = &{foreach($res in $iSCSITargetResources) { Get-
WmiObject -NameSpace root\mscluster -Authentication
PacketPrivacy -Query "associators of {$($res.__RELPATH)}  WHERE
ResultClass = MSCluster_ResourceGroup" }}
PS > $Groups
```

2. From the cluster resources obtained in the previous steps, gather the cluster disk dependencies by using the following Windows PowerShell command:

```
PS > $Dependencies = &{foreach($res in $iSCSITargetResources) {
Get-WmiObject -NameSpace root\mscluster -Authentication
PacketPrivacy -Query "associators of {$($res.__RELPATH)}  WHERE
ResultClass = MSCluster_Resource ResultRole = Dependent" }}
PS > $Dependencies
```

The resource groups obtained in step 1 have network name resources and IP addresses that need to be migrated to the destination server.

For information about how to migrate these settings, see Migrate IP Configuration to Windows Server 2012.

The cluster disk that you obtained in step 2 is the physical disk where the volumes that are hosting the iSCSI Software Target virtual disks reside.

To obtain the volumes from the cluster disk, use the following steps:

1. Obtain the disk signature of the cluster disk by using the following Windows PowerShell command:

```
PS > & cluster.exe res "<cluster resource name>" /priv
```

2. Obtain the Win32_DiskDrive object from the disk signature by using the following Windows PowerShell command:

```
PS > $DiskObj = Get-WmiObject -Namespace root\cimv2 -Class
Win32_DiskDrive -Filter "Signature = <disk signature>"
PS > $DiskObj
```

3. Obtain the Win32_DiskDriveToDiskPartition association by using the following Windows PowerShell command:

```
PS > $DiskToDiskPartition = Get-WmiObject -Namespace root\cimv2
-Class Win32_DiskDriveToDiskPartition | Where-Object {
$_.Antecedent -eq $DiskObj.__PATH }
PS > $DiskToDiskPartition
```

4. Obtain the Win32_LogicalDiskToDiskPartition association that points to the volume association by using the following Windows PowerShell command:

```
PS > Get-WmiObject -Namespace root\cimv2 -Class
Win32_LogicalDiskToPartition | Where-Object { $_.Antecedent -eq
$ DiskToDiskPartition.Dependent }
```

Steps 2 through 4 need to be applied on the source server cluster node that currently owns the physical disk cluster resource.

iSCSI Target portal configuration

Use the following steps to obtain the portal associations:

1. Gather the configured portals association for the iSCSI target portal by using the following Windows PowerShell command:

   ```
   PS> Get-WmiObject -Namespace root\wmi -Class WT_portal | Format-List -Property Address,Listen,Port
   ```

2. The IP addresses that have the Listen state set to True are the IP addresses that an iSCSI initiator can use to reach the server. For more information about migrating the IP addresses, see Migrate IP Configuration to Windows Server 2012.

iSNS configuration

Gather the configured iSCSI Naming Services (iSNS) association for the server by using the following Windows PowerShell command:

```
PS> Get-WmiObject -Namespace root\wmi -Class WT_ISnsServer | Format-List -Property ServerName
```

The server names that are listed need to be added to the list of iSNS servers that can be used to retrieve information about the iSCSI initiators in the enterprise.

CHAP and Reverse CHAP configuration

Gather the UserName and ReverseCHAPUserName association for the servers that are configured with CHAP and Reverse CHAP by using the following Windows PowerShell command:

```
PS > Get-WmiObject -Namespace root\wmi -Class WT_Host | Where-Object { ( $_.EnableCHAP ) -or ( $_.EnableReverseCHAP )  } | Format-List -Property Hostname,CHAPUserName,ReverseCHAPUserName
```

The passwords that are used in conjunction with the credentials listed previously cannot be retrieved, and they must be known through other mechanisms.

Snapshot storage configuration

The snapshot storage configuration can be obtained by using the following Windows PowerShell command:

```
PS > & vssadmin.exe list shadowstorage
```

This command shows the volume snapshot shadow storage configuration for the entire source server. Not all the volumes listed may be relevant to the current iSCSI Software Target server configuration.

For the volumes that are relevant (that is, the volumes that host iSCSI virtual disks), the associated shadow storage volume is listed, in addition to the amount of disk space used with the maximum amount of configured space.

Disconnect the iSCSI initiators

Follow the instruction in the following section to disconnect the iSCSI initiators: Prepare other computers in the enterprise.

Capture the existing settings: standalone configuration

All of the settings on the iSCSI Software Target source server that are not hardware configuration specific and are not dependent on an IP address and the network identity of the server can be captured with the following Windows PowerShell command:

Windows Server 2008 and Windows Server 2008 R2 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"

PS> .\ iSCSITargetSettings.PS1 –Export –FileName <settings XML file>
```

Windows Server 2012 file path:

```
PS > cd "$ENV:SystemRoot\System32\WindowsPowerShell\V1.0\Modules\IscsiTarget"

PS> .\ iSCSITargetSettings.PS1 –Export –FileName <settings XML file>
```

If the procedure is performed on a source server that is running ISCSI Target 3.3 from a destination server that is prepared as illustrated in the previous sections, the settings can be captured using the following Windows PowerShell command:

Windows Server 2012 file path:

```
PS > cd "$ENV:SystemRoot\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget"

PS> .\ iSCSITargetSettings.PS1 –Export –FileName <settings XML file> -ComputerName
<source server computer name>
```

Windows Server 2008 and Windows Server 2008 R2 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"

PS> .\ iSCSITargetSettings.PS1 –Export –FileName <settings XML file> -ComputerName
<source server computer name>
```

At the end of the settings capture process, the Windows PowerShell script will display the set of VHD files that are eligible for migration. This list is needed for the destination server during migration.

Capture the existing settings: clustered configuration

Before capturing the iSCSI Software Target source server settings that are not hardware configuration specific, we recommend that all the resource groups with iSCSI target resources are moved to a single node in the cluster.

This can be accomplished by using the following Windows PowerShell commands. These commands assume that you previously followed the steps in the following section: Cluster resource group configuration.

```
PS > $iSCSITargetResources | Format-List -Property OwnerGroup

PS > foreach($Res in $iSCSITargetResources) { & cluster group  $Res.OwnerGroup
/moveto:$ENV:COMPUTERNAME }
```

After all the resource groups have been moved to a single node, the settings can be gathered by using the following Windows PowerShell commands:

Windows Server 2012 file path:

```
PS > cd "$ENV:Programfiles\ISCSI Target"

PS> .\ iSCSITargetSettings.PS1 -Export -FileName <settings XML file>
```

Windows Server 2008 and Windows Server 2008 R2 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"

PS> .\ iSCSITargetSettings.PS1 --Export -FileName <settings XML file>
```

If the procedure is performed on a source server that is running ISCSI Target 3.2, the resources can be moved to a single node by using the following Windows PowerShell command:

```
PS >  $Groups = &{foreach($res in $iSCSITargetResources) { Get-WmiObject -NameSpace
root\mscluster -Authentication PacketPrivacy -Query "associators of {$($res.__RELPATH)}
WHERE ResultClass = MSCluster_ResourceGroup" }}

PS > foreach($Group in $Groups) { & cluster group $Group.Name /moveto:<node name source
server> }
```

The ISCSI Target Server settings need to be gathered from a destination server that is prepared as illustrated in the previous sections. Run the script for a source server that is running ISCSI Target 3.3 by using the following Windows PowerShell command:

Windows Server 2012 file path:

```
PS > cd "$ENV:Programfiles\ISCSI Target"

PS> .\ iSCSITargetSettings.PS1 -Export -FileName <settings XML file> -ComputerName
<source server computer name>
```

Windows Server 2008 and Windows Server 2008 R2 file path

```
PS > cd "$ENV:SystemRoot\Program Files\Microsoft iSCSI Software Target"
```

```
PS> .\ iSCSITargetSettings.PS1 –Export –FileName <settings XML file> -ComputerName
<source server computer name>
```

In this command, the source server computer name is the name of the node. At the end of the settings capture process, the Windows PowerShell script will display the set of VHD files that are eligible for migration. This list is needed for the destination server during migration.

Remove the network identity of the iSCSI Software Target computer

In a network with an iSCSI Software Target source computer, the identity of the server is known to iSCSI initiators in the form of NetBIOS names, fully qualified domain names (FQDN), or IP addresses. When a server is being replaced, as part of planning, a strategy to replace the server network identity must be devised. Possible scenarios include:

- Transfer the NetBIOS and fully qualified domain names to the destination server, and then assign new IP addresses to the destination server.
- Create new NetBIOS and fully qualified domain names for the destination server, and then assign the existing IP addresses to the destination server.
- Create new NetBIOS and fully qualified domain names for the destination server, and then assign new IP addresses to the destination server.

Each scenario requires potentially updating information in the DNS server, Active Directory, or DHCP server, according to the methodology that is used to assign IP addresses and network names to the servers in the enterprise.

The intent of this step is to ensure that upon completion of the migration steps, the iSCSI initiators are able to locate the destination server (either through explicit reconfiguration, or implicitly through the computer name or IP address re-assignment).

Prepare the iSCSI initiator computers

The other computers in the enterprise that are affected by migration are the iSCSI initiators. The users of the computers that are acting as iSCSI initiators should be sent an outage notification. If the iSCSI Software Target is being used as a boot node for the iSCSI initiator computers, the computers may be completely unusable for the duration of the migration.

Capture the session information

The information regarding the active session for an iSCSI Software Target source server can be obtained by using the following Windows PowerShell command:

```
PS > & iscsicli.exe sessionlist
```

This information is needed to disconnect the session in the following step.

Disconnect the session

The session can be disconnected by using the following Windows PowerShell command:

```
PS > & iscsicli.exe LogoutTarget <session id>
```

Migrate iSCSI Software Target

This topic discusses the actual migration steps for ISCSI Software Target 3.2 or iSCSI Software Target 3.3 for both the standalone configuration and the clustered configuration:

Migrating ISCSI Software Target in a standalone configuration

The migration of ISCSI Software Target 3.2 or iSCSI Software Target 3.3 has equivalent steps, whether you are migrating from Windows Storage Server 2008 or Windows Storage Server 2008 R2 to Windows Server 2012 or Windows Storage Server 2012.

Establish network identity of the iSCSI Target Server computer

As part of the planning process, a strategy should have been devised regarding how iSCSI Target Server will be accessed from the network, including but not limited to:

- Which computer name will be used?
- Which IP addresses on which subnet or set of network interfaces will be used?
- What relationship should be maintained between the IP addresses and computer name of the source server and the destination server—will you keep the same addresses and names or create new ones?

Based on the desired final configuration, configuration changes are potentially needed in the following areas:

- The DHCP Server that might assign IP addresses to the destination iSCSI Target servers
- The DHCP Server that might assign IP addresses to the iSCSI initiators
- The DNS Server or Active Directory domain controller that might perform naming resolution services for the computers in the enterprise

Configure the iSCSI Target Server portal

After you have configured IP addresses for the network interfaces of the iSCSI Target Server computer, it is possible to verify the existing configuration by using the following Windows PowerShell command:

```
PS > $Portals = Get-WmiObject -Namespace root\wmi -Class WT_Portal | Where-Object {
$_.Listen }

PS > $Portals
```

The configuration of the access surface for iSCSI Target Server from the network can be restricted by disabling certain portals. For example, you can disable the fourth portal in the array returned in the previous step by using the following Windows PowerShell commands:

```
PS > $Portals[3].Listen = 0

PS > $Portals[3].Put()
```

The default port can also be changed from 3260 to any available TCP port on the destination server.

Configure iSNS settings

The iSNS servers that were configured for the source server can be configured for the destination server by using the following Windows PowerShell commands:

```
PS > $WT_ISnsServerClass =  Get-WmiObject -namespace root\wmi -class meta_class -filter
"__CLASS = 'WT_ISnsServer'"

PS > $WtiSNSInstanace = $WT_ISnsServerClass.CreateInstance()

PS > $WtiSNSInstanace.ServerName = "<iSNS computer name or IP>"

PS > $WtIsnsInstanace.Put()
```

Note

The set of iSNS servers that are configured for iSCSI Target Server was obtained during the preparation of the source server.

Configure storage

The destination server is expected to have sufficient storage space to host all of the virtual disks that are present on the source server.

The space does not need to be contiguous or in a single volume, and it does not need to replicate the same file system structure or volume mount point structure of the source server.The storage that is prepared to host the virtual disks must not be a nested volume, and it must be formatted with the NTFS file system.

Configure the Volume Shadow Copy Service

For the storage that was prepared in the previous step, it is appropriate to configure the Volume Shadow Copy Service, in case the default per-volume settings are not adequate.To inspect that current configuration, use the following Windows PowerShell command:

```
PS > & vssadmin.exe list shadowstorage
```

To modify the current configuration, use the following Windows PowerShell commands:

```
PS > & vssadmin.exe add ShadowStorage

PS > & vssadmin.exe delete ShadowStorage

PS > & vssadmin.exe resize ShadowStorage
```

Transfer the virtual disk

For all the files in the list of files that was captured in the source server preparation step, copy the files from the source server to the destination server. For more information, see Capture the existing settings section.

You will need the destination paths in the following steps. So if the absolute file path is different between the source server and the destination server, create a table with the mapping, for example

Source path	Destination path
G:\WS08R2_OpsMgr2007_R2.vhd	H:\VHDS\WS08R2_OpsMgr2007_R2.vhd
F:\Dynamic_Spanned_GPT_2.vhd	D:\DYNVHDS\Dynamic_Spanned_GPT_2.vhd

Import the iSCSI Software Target settings in a standalone configuration

To import the iSCSI Software Target settings in a standalone configuration, you need the .xml file that you previously created. For more information, see Capture the existing settings section.

If there is no change in the absolute path of the virtual disk files, the import process can be performed by using the following Windows PowerShell commands:

```
PS > cd "$ENV:SystemRoot\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget"

PS> .\ iSCSITargetSettings.PS1 –Import –FileName <settings XML file>
```

If the absolute path is different between the source server and the destination server, before you import the settings, the settings .xml file needs to be altered to reflect the new path.

Locate the records for the virtual disk, and alter the path in the <MigrationDevicePath> tag to reflect the absolute file path in the destination server, for example:

```
<iSCSIVirtualDisk>

 <DevicePath>F:\Dynamic_Spanned_GPT_2.vhd</DevicePath>

  <MigrationDevicePath>D:\DYNVHDS\Dynamic_Spanned_GPT_2.vhd</MigrationDevicePath>

</iSCSIVirtualDisk>
```

After the XML has been altered to reflect the path in the destination server, you can import the settings by using the Windows PowerShell commands previously presented.

Configure shadow storage for the virtual disks

If certain virtual disks have shadow storage requirements that are different than the ones configured in the section Configure the Volume Shadow Copy Service , it is possible to alter the default or previously configured settings by using the following Windows PowerShell commands:

```
PS > $VirtDisk = Get-WmiObject -Namespace root\wmi -Class WT_Disk | Where-Object {
$_.DevicePath -eq '<full path of virtual disk>' }
PS > $VirtDisk.SnapshotStorageSizeInMB = <new size>
PS > $VirtDisk.Put()
```

Configure CHAP and Reverse CHAP

The authentication settings for iSCSI Target Server that are configured with CHAP and Reverse CHAP need to be manually configured. The list of targets that require CHAP and Reverse CHAP configuration is listed at the end of the import script, as described in the section Import the iSCSI Software Target settings in a standalone configuration.

To configure the CHAP and Reverse CHAP settings, use the following Windows PowerShell commands:

```
PS > $Target = Get-WmiObject -Namespace root\wmi -Class WT_Host | Where-Object {
$_.HostName -eq '<name of the target>' }
PS > $Target.EnableCHAP = 1
PS > $Target.CHAPUserName = "<user name>"
PS > $Target.CHAPSecret = "<CHAP Secret>"
PS  $Target.Put()
```

Migrating iSCSI Software Target in a failover cluster

The migration process for the failover cluster configuration is largely identical to migrating a standalone configuration, with the following differences:

- You will migrate resource groups instead of merely establishing the network identity of the server.
- You will use different Windows PowerShell commands to import the resource groups.

Migrate resource groups

This step replaces the “Establishing the network identity of iSCSI Target Server” step when you migrate a standalone configuration. The reason is that the network identity of an iSCSI Target server in a cluster is given by the union of the client access point. (A client access point in the cluster is the logical union of a network name resource and one or more IP addresses that are assigned to the network name resource.)

Assuming the initial cluster resource groups and network names were configured in the default state, those can be recreated by using the following Windows PowerShell command:

```
PS > Add-ClusterServerRole – Name <resource group name>
```

Use this command for each of the resource groups that were in the original configuration.If the default client access point configuration does not match the initial configuration (for example, because the network name is bound to the incorrect cluster network, or the configuration required statically assigned IP addresses), modifications can be made. For more information, see Migrate IP Configuration to Windows Server 2012.

After the resource groups have been created, clustered disks must be assigned to the network resources to match the configuration that you captured. For more information, see the Cluster resource group configuration section.

Import the iSCSI Software Target settings in a failover cluster

Follow these instructions to import settings in a failover cluster configuration. (This information differs from the how you would import settings in a standalone configuration.)

A prerequisite for the import phase is to have all of the resource groups that will host iSCSI Target Server resources owned by the same cluster node. Use the following Windows PowerShell command to validate the current ownership:

```
PS > Get-ClusterGroup
```

If there is no change in the absolute path of the virtual disk files, the import process can be performed by using the following commands:

```
PS > cd “$ENV:SystemRoot\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget”

PS> .\iSCSITargetSettings.PS1 –Import –FileName <settings XML file>
```

If the absolute path is different between the source server and the destination server, before you import the settings, the settings .xml file needs to be altered to reflect the new path.Locate the records for the virtual disk, and alter the path in the <MigrationDevicePath> tag to reflect the absolute file path in the destination server, for example:

```
<iSCSIVirtualDisk>

      <DevicePath>F:\Dynamic_Spanned_GPT_2.vhd</DevicePath>
```

```
    <MigrationDevicePath>D:\DYNVHDS\Dynamic_Spanned_GPT_2.vhd</MigrationDevicePath>
  </iSCSIVirtualDisk>
```

After the XML has been altered to reflect the path in the destination server, you can import the settings by using the Windows PowerShell commands previously presented.

Verify the iSCSI Software Target Migration

This topic discusses the steps that you can use to verify that the migration successfully completed.

Verifying the destination server configuration

To verify that the destination server has been properly configured after migration, you can verify the listening endpoints and connectivity and run a scan with the Best Practices Analyzer.

Verify the listening endpoints

On the iSCSI Target destination server, you can validate that the target portals have been configured as desired by using the following Windows PowerShell command:

```
PS > & netstat.exe -nao | findstr 3260 | findstr LISTENING
  TCP    10.121.26.107:3260     0.0.0.0:0              LISTENING       1560
  TCP    10.121.26.126:3260     0.0.0.0:0              LISTENING       1560
  TCP    [2001:4898:0:fff:0:5efe:10.121.26.126]:3260  [::]:0                 LISTENING
1560
  TCP    [2001:4898:f0:1001:f063:8fc5:52e6:2310]:3260  [::]:0                 LISTENING
1560
```

The list of IP addresses and port pairs in the listening state needs to match the desired set of target portals.

Note
If ports other than the default 3260 are being used, the command needs to be altered to reflect the alternate IP ports.

Verify the basic connectivity

To validate that the iSCSI Target Server computer is reachable from other computers on the network, from a computer that has the Telnet Client feature installed, use the following Windows PowerShell command:

```
PS > telnet.exe <iSCSI Software Target machine name or IP> 3260
```

If there is a successful connection, Telnet Client will show a blinking cursor at the top of the window. Press any key to close Telnet Client.

Perform a Best Practices Analyzer scan

To verify that ISCSI Target Server is optimally configured on Windows Server 2012 or Windows Storage Server 2012 after migration, we recommend that you run a Best Practices Analyzer (BPA) scan on the role.

BPA is a server management tool that is available in Windows Server 2012. After the migration of ISCSI Target 3.3 is complete, BPA can help you ensure that your server is configured according to best practices. You can use the Server Manager console UI or Windows PowerShell to perform BPA scans and view results. For detailed information about how to scan your role and view results, see the Best Practices Analyzer Help (http://go.microsoft.com/fwlink/?LinkId=122786).

Verifying the configuration of iSCSI initiator computers

Validating the migration of ISCSI Software Target to the destination server includes ensuring that the iSCSI initiators can discover and fully access all features of the iSCSI protocol.

Verify that the iSCSI initiators can discover iSCSI Target Server

To verify that the iSCSI initiators can discover iSCSI Target Server, use the following Windows PowerShell commands:

```
PS > & iscsicli AddTargetPortal <ip-address> 3260
```

```
PS > & iscsicli.exe ListTargets
```

If the commands execute without errors, the initiator is capable of discovering the targets that are offered by the server

Verify that the iSCSI initiators can log on

The second step is to verify that the iSCSI initiators are able to log on to the iSCSI targets that are exposed by iSCSI Target Server. This can be accomplished by using the following Windows PowerShell command:

```
PS > & iscsicli.exe  LoginTarget <target IQN> T <ip address> 3260 Root\ISCSIPRT\0000_0 *
* * * * * * * * * * * * *
```

Note

If you are using CHAP and Reverse CHAP authentication, you may need to specify more parameters. For more information, consult the documentation in the iscsicli.exe.

If the command executes without errors, the iSCSI initiator has successfully logged on to the target, and the disks are exposed to iSCSI Target Server.

Troubleshoot the iSCSI Software Target Migration

Troubleshooting iSCSI Software Target migration issues involves first viewing the contents of the Windows Server Migration Tools deployment log and the result objects. For more information, see Locate the deployment log file and View the content of Windows Server Migration Tools result objects.

Understanding the messages from the iSCSI Target Migration tool

The iSCSI migration tool (iSCSITargetSettings.PS1) does not produce a log file, but it prints diagnostics messages on the console. These messages show the outcome of the operations that are being attempted and performed.

For example, the following message shows information about saved settings:

```
PS C:\Windows\System32\WindowsPowerSehll\V1.0\Modules\IscsiTarget>

.\iSCSITargetSettings.PS1 -Export -FileName $env:temp\test00000000.xml


Number of Target(s) saved in the Export settings: 4.

Target Names:

   test000

   test001

   test002

   test1111


Number of Virtual Disk(s) saved in the Export settings: 3.

Virtual Disk DevicePaths:

   s:\test000.vhd

   t:\test000.vhd

   z:\test000.vhd
```

```
Number of Virtual Disk(s) NOT saved in the Export settings: 0.
Virtual Disk DevicePaths:
```

The following message shows that not all the virtual disks are eligible for migration:

```
PS D:\Program Files\ISCSI Target> .\iSCSITargetSettings.PS1 -Export -FileName
$env:temp\test00000001.xml

Number of Target(s) saved in the Export settings: 4.
Target Names:
   test000
   test001
   test002
   test1111

Number of Virtual Disk(s) saved in the Export settings: 3.
Virtual Disk DevicePaths:
   s:\test000.vhd
   t:\test000.vhd
   z:\test000.vhd

Number of Virtual Disk(s) NOT saved in the Export settings: 1.
Virtual Disk DevicePaths:
    \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy{B6B3C77C-93CC-11DF-B3FE-
001CC0C60A6E}\test000.vhd
```

The following message shows information about the settings restore phase:

```
PS C:\Program Files\ISCSI Target> .\iSCSITargetSettings.PS1 -Import -file
$env:temp\test00000000.xml

Importing settings from file
'E:\Users\administrator\AppData\Local\Temp\test00000001.xml'.
The operation may take a long time.

Number of Target(s) imported from the Import settings: 4.
Targets:
```

```
    test000

    test001

    test002

    test1111


Number of Virtual Disk(s) imported from the Import settings: 3.

Virtual Disk:

   s:\test000.vhd

   t:\test000.vhd

   z:\test000.vhd
```

Roll Back a Failed iSCSI Software Target Migration

If iSCSI initiators have successfully reconnected to the iSCSI Target Server computer, the migration is successful and complete. This topic discusses the tasks that should be performed in the event of a failed migration.

Restoring the role if the migration failed

If migration does not complete successfully, a rollback procedure is required to undo any changes to the source server, other servers, and client computers, and then restore the source server back into service.

Rollback requirements

The rollback procedure requires that the source server is available in the same state as it was after the “Remove the network identity of the iSCSI Software Target server” step in the “Prepare your source server” section. For more information, see Remove the network identity of the iSCSI Software Target server.

During the source server preparation steps, none of the steps performed permanently changed the existing configuration of the server because all of the operations were substantially read operations.

The estimated time to complete the rollback is equivalent to the time that it takes to re-establish the network identity of the source server. This operation may require rolling back changes to the DHCP servers, DNS server, or Active Directory Domain controllers.

Roll back iSCSI initiators on other computers

The other computers in the enterprise that are affected by migrating ISCSI Software Target are the iSCSI initiators.

In the case of a rollback, the iSCSI initiators that were configured to log on to the destination server need to be rolled back to the source server. Use the following Windows PowerShell commands:

1. To log out of an existing iSCSI session:

   ```
   PS > & iscsicli.exe sessionlistPS > & iscsicli.exe LogoutTarget <session id>
   ```

2. To discover the iSCSI Software Target source server:

   ```
   PS > & iscsicli AddTargetPortal <source server ip address> 3260PS > iscsicli.exe
   ListTargets
   ```

3. To log on to the targets on the iSCSI Software Target source server:

   ```
   PS > & iscsicli.exe  LoginTarget <target IQN> T < source server ip address> 3260
   Root\ISCSIPRT\0000_0 * * * * * * * * * * * * *
   ```

Roll back iSCSI Software Target on a standalone source server

This step will undo the network identity removal that is described in “Remove the network identity of the iSCSI Software Target server’.

Possible scenarios include:

- Restore the NetBIOS fully qualified domain name to the source server, and assign the required IP addresses to the source server.
- Restore any DNS assignments (for example, reverse lookup and DHCP assignment).
- Restore any identities that were previously assigned in Active Directory.

Each scenario requires potentially updating information in the DNS server, Active Directory, or DHCP server, according to the methodology that is used to assign IP addresses and network names to the servers in the enterprise.

The intent of this step is to ensure that upon completion of the migration steps, the iSCSI initiators are able to locate the source server (either through explicit reconfiguration, or implicitly through the computer name or IP address re-assignment).

Roll back iSCSI Software Target on a clustered source server

Rolling back iSCSI Software Target on a clustered source server requires two steps:

Step 1: Roll back cluster network name changes

This step will undo the network identity removal described in “Remove the network identity of the iSCSI Software Target server”.

In a clustered configuration, network names are established by the Server Principal Name that is assigned in Active Directory to the cluster when the cluster was formed.

To re-establish network names that were possibly deleted or retired, the cluster administration utilities must be used. For more information, see Migrating Settings to a Failover Cluster Running Windows Server 2008 R2.

Step 2: Move resource groups to the preferred owner node

After the client access points have been re-established, the resource groups need to be moved back to their preferred owner node.

The resource groups were moved to a single node as part of the steps performed in “Capture the existing settings: clustered configuration”.

To move the resource groups back to their preferred owner node, use the following Windows PowerShell command:

```
PS > & cluster.exe /cluster:<cluster name> GROUP <group name> /moveto:<node name>
```

Note
The group name and the node names were obtained during the previous preparation tasks.

Roll back iSCSI Target Server on a standalone destination server

To roll back iSCSI Target Server on a standalone destination server that is running Windows Server 2012 or Windows Storage Server 2012, uninstall the **iSCSI Target Server** role service using Server Manager.

Roll back iSCSI Target Server on a clustered destination server

To roll back iSCSI Target Server on a destination server that is running Windows Server 2012 or Windows Storage Server 2012 in a clustered configuration, first remove any client access point that was created for iSCSI Target Server and then uninstall the **iSCSI Target Server** role service using Server Manager.

Retiring iSCSI Software Target on a source server

Retiring ISCSI Software Target 3.2 or iSCSI Software Target 3.3 on your source server requires using the following Windows PowerShell commands:

Retire iSCSI Software Target

1. Find the package GUID:

   ```
   PS > Get-WmiObject -Class Win32_product | Where-Object { $_.packageName -match
   'iscsitarget'}
   ```

2. Uninstall the package:

   ```
   PS > & msiexec /uninstall <package GUID> /qr
   ```

Retiring a source server

In a standalone configuration, there are no particular procedures for retiring the source server. In a clustered configuration, the client access points that are devoted to iSCSI Software Target access can be removed by using the following Windows PowerShell command:

```
PS > Remove-ClusterGroup -Name <resource group name> -RemoveResources -Force
```

Migrate Health Registration Authority to Windows Server 2012

This document provides guidance for migrating the Health Registration Authority (HRA) role service from an x86-based or x64-based server running Windows Server® 2008, Windows Server® 2008 R2, or Windows Server® 2012 to a new Windows Server 2012 server.

About this guide

Note

Your detailed feedback is very important, and helps us to make Windows Server Migration Guides as reliable, complete, and easy to use as possible. Please take a moment to rate this topic by clicking the stars in the upper-right corner of the page (1=poor, 5=excellent), and then add comments that support your rating. Describe what you liked, did not like, or want to see in future versions of the topic. To submit additional suggestions about how to improve Migration guides or utilities, post on the Windows Server Migration forum.

This guide describes the steps for migrating existing HRA server settings to a server that is running Windows Server 2012. By using this documentation, you can simplify migration, reduce or eliminate server downtime, and help eliminate possible conflicts that might otherwise occur during HRA migration.

Target audience

This guide is intended for information technology (ITOS) administrators, IT professionals, and other knowledge workers who are responsible for the operation and deployment of HRA servers in a managed environment.

What this guide does not provide

This guide does not provide detailed steps to migrate the configuration of other services used with NAP, such as Network Policy Server (NPS) or Active Directory Certificate Services (AD CS). These procedures are found in the Migrate Network Policy Server to Windows Server 2012 and the Active Directory Certificate Services Migration Guide

(http://go.microsoft.com/fwlink/p/?LinkID=156771). Instructions to perform specific procedures in these other guides are provided as necessary to complete migration of the HRA server.

Supported migration scenarios

This guide provides you with instructions for migrating an existing server that is running the HRA role service to a server that is running Windows Server 2012. This includes guidance for installing the prerequisite IIS server role and NPS role service. If your server is running additional services, it is recommended that you design a custom migration procedure specific to your server environment based on the information provided in other role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server 2012.

Caution

If your source server provides other roles and services in addition to HRA, migrating the computer name and IP configuration can cause these services to fail. You must verify the impact of these procedures before performing them during HRA migration.

Supported operating systems

The following table displays the minimum operating system requirements.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2008	Windows Server 2012	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2012	x64-based
x64-based	Windows Server 2012	Windows Server 2012	x64-based

- The NPS and HRA roles services are not available in Server Core editions. Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported as either source or destination servers. However, If you have configured AD CS on the source server as an enterprise certification authority (CA), the destination CA server must be running Enterprise or Datacenter editions of Windows Server 2012.
- Migration from a source server to a destination server that is running an operating system with a different installed language is not supported. For example, migration of server roles from a computer that is running Windows Server 2008 with a system language of French to a computer that is running Windows Server 2012 with a system language of German is not supported. The system language is the language of the localized installation package that was used to set up the Windows operating system.
- Both x86- and x64-based migrations are supported for Windows Server 2008. All editions of Windows Server 2008 R2 and Windows Server 2012 are x64-based.

Supported role configurations

This guide provides procedures to migrate all HRA server settings, including any custom CA and request policy settings. This guide also provides instructions for configuring minimum IIS role requirements on the destination server.

Migrating prerequisite roles

HRA is a role service under the Network Policy and Access Services (NPAS) server role. To install HRA, you must also install NPS and IIS on the same computer. If these services are not already installed, they will be added automatically by the Add Roles and Features Wizard when you choose to install HRA.

HRA also requires a connection to one or more servers running AD CS that are configured to provide NAP health certificates. AD CS can be installed on the same computer with HRA, or it can be installed on another computer. If any HRA severs in your organization are configured to use AD CS on the source server for health certificate requests, you must install AD CS on the destination HRA server and configure it to provide health certificates, or you can change the CA configuration of your HRA servers.

Consider the following information about prerequisite roles and required services on the destination HRA server.

1. **NPS**. The NPS role service must be migrated before you can test and verify the functionality of HRA on the destination server. If NPS on the source server is only used with HRA, either as a standalone NAP IPsec health policy server or as a RADIUS proxy for another health policy server, this guide provides references to specific procedures in the Migrate Network Policy Server to Windows Server 2012 that are required to migrate required NPS policies and settings. If the NPS role on the source server is used for purposes other than IPsec NAP, or if the source server is a member of RADIUS clients or remote RADIUS server groups on other servers in your organization, consult the Migrate Network Policy Server to Windows Server 2012 for detailed migration instructions prior to migrating HRA.
2. **AD CS**. During installation of HRA, you can choose to install AD CS on the same computer, to use an existing NAP CA on a different computer, or to select a CA later. You can also choose to install AD CS as an enterprise CA or a standalone CA.

   Warning
   After you install AD CS on the HRA server, you cannot change the name of the HRA server.

   - If you install AD CS on the same computer with HRA, you must configure AD CS on the destination HRA server to provide NAP health certificates.
     - If AD CS is installed as an enterprise CA, use procedures in this guide to configure permission settings for the NAP CA. See the Active Directory Certificate Services Migration Guide (http://go.microsoft.com/fwlink/p/?LinkID=156771) for procedures to migrate health certificate templates to the destination server.
     - If AD CS is installed as a standalone CA, this guide provides all permission setting procedures that are required to configure a NAP CA on the destination server. If you use the local CA for other purposes than issuing NAP health certificates, or you have

a custom configuration, see the Active Directory Certificate Services Migration Guide (http://go.microsoft.com/fwlink/p/?LinkID=156771) for detailed instructions to migrate CA settings.

- If you use an existing NAP CA on a different computer, you do not need to configure AD CS on the destination server.
- If you choose to select a CA later, you do not need to configure AD CS on the destination server. If you choose to install AD CS on the destination HRA server later, see Deploying NAP Certification Authorities.

If AD CS on the source server is also used to issue certificates that are not health certificates, see the Active Directory Certificate Services Migration Guide (http://go.microsoft.com/fwlink/?LinkID=156771) for procedures to migrate AD CS.

3. **IIS**. If the prerequisite IIS server role is used for any purposes other than the HRA, or is run with customized settings beyond adding an SSL certificate, follow procedures in the Internet Information Services Migration Guide prior migrating the HRA. If the IIS server role is only used with HRA, use the procedures in this guide to migrate IIS.

   Important
   To maintain HRA performance, the default IIS connection settings must be modified to increase the maximum number of concurrent connections. To perform this procedure, see the Configure IIS connection settings section in Configure an HRA server for NAP.

Migration scenarios that are not covered

The following migration scenarios are not covered in this document:

- **Upgrade**. Guidance is not provided for scenarios in which the new operating system is installed on existing server hardware by using the **Upgrade** option during setup.
- **Workgroup**. Guidance is not provided for migration of HRA settings to or from a non-domain-joined server.

Overview of migration process for this role

HRA server migration is divided into the following major sections:

- HRA Server Migration: Preparing to Migrate
- HRA Server Migration: Migrating the HRA Server
- HRA Server Migration: Verifying the Migration
- HRA Server Migration: Post-migration Tasks

The pre-migration process involves establishing a storage location for migration data, collection of information that will be used to perform the server migration, and operating system installation on the destination server. The HRA migration process includes using the Network Shell (netsh) utility from a command line on the source server to obtain the required HRA settings, and procedures on the destination server to install the required roles and migrate the HRA settings. Verification

procedures include testing the destination server to ensure it works correctly. Post-migration procedures include retiring or repurposing the source server.

Impact of migration

If your migration plan involves configuring the destination server with a different host name from the source server, the trusted server group settings on NAP client computers that use the source HRA server must be updated to use the destination HRA server. This approach has the advantage that it allows the source and destination HRA servers to run simultaneously until testing and verification is complete.

If your migration plan involves configuring the destination server with the same name as the source server, then the source server must be decommissioned and taken offline prior to joining the destination server to the same domain with the same host name. To eliminate downtime in this scenario, NAP client computers must have access to a secondary HRA server in addition to the source and destination servers. To eliminate short term name resolution issues, use the same IP address configuration on the source and destination server.

If the NPS role on the source server is used for purposes other than IPsec NAP, client computers might fail to access the network during the server migration process. For example, if the source server is used for VPN client authentication, consult the Migrate Network Policy Server to Windows Server 2012 for detailed migration instructions prior to migrating HRA.

Impact of migration on the source server

- When deploying the destination server with a different host name, there is no impact to the source server.
- When deploying the destination server with the same host name, the source server must be decommissioned and taken offline prior to joining the destination server to the domain.

Impact of migration on other computers in the enterprise

- When deploying the destination server with a different host name, the NAP client settings for all machines configured to use the HRA must be updated. There is little to no downtime in this scenario if the procedures in this guide are followed.
- When deploying the destination server with the same host name, clients will not be able to obtain a health certificate shortly after the source server is decommissioned, unless a secondary HRA server is deployed.

Permissions required to complete migration

The following permissions are required on the source server and the destination server:

- Domain administrative rights are required to configure and authorize the HRA server, and configure group policy settings for NAP clients.
- Local administrative rights are required to install or manage the server running HRA.

Estimated duration

The migration can take two to three hours, including testing.

See Also

HRA Server Migration: Preparing to Migrate

HRA Server Migration: Migrating the HRA Server

HRA Server Migration: Verifying the Migration

HRA Server Migration: Post-migration Tasks

Network Access Protection Design Guide

Network Access Protection Deployment Guide

HRA Server Migration: Preparing to Migrate

Migration of Health Registration Authority (HRA) Server includes the following tasks:

- Choose a migration file storage location
- Prepare your source server
- Prepare your destination server

Complete the steps or procedures in these sections to prepare your environment for migration.

Membership in the **Domain Admins** group, or equivalent, is the minimum required to complete this procedure. Review details about using the appropriate accounts and group memberships at Local and Domain Default Groups (http://go.microsoft.com/fwlink/?LinkId=83477).

Choose a migration file storage location

First, choose a location where migration files will be kept.

To choose a storage location

1. Select a file storage location where migration files will be kept. The storage location can be a network share that is accessible by both the source and destination server, or portable media that can be transferred from one server to another.

Prepare your source server

Follow these steps to prepare an x64 or x86-based server running Windows Server® 2008, Windows Server® 2008 R2, or Windows Server® 2012 for HRA migration.

To prepare the source server

1. Determine the domain, server name, IP address, and passwords on the source server.
2. Determine the group membership of the source server in Active Directory Domain Services (AD DS), including security group and OU membership. This can be done using the Active Directory Users and Computers console (dsa.msc) or Server Manager on a domain controller.

Prepare your destination server

Follow these steps to prepare an x64-based destination server running Windows Server 2012 for HRA migration.

To prepare the destination server

1. Install Windows Server 2012 on the destination server.
2. Configure the host name of the computer, and configure network settings as desired. Do not join the computer to the domain yet.
3. Install all critical updates and service packs on the destination server.
4. Verify the server has access to the migration file storage location.

See Also

Migrate Health Registration Authority to Windows Server 2012

HRA Server Migration: Migrating the HRA Server

HRA Server Migration: Verifying the Migration

HRA Server Migration: Post-migration Tasks

Network Access Protection Design Guide

Network Access Protection Deployment Guide

HRA Server Migration: Migrating the HRA Server

This topic contains steps and procedures for migrating the Health Registration Authority (HRA) role service from a legacy source server to a new x64-based destination server running Windows Server® 2012.

Important

The NPS role service must be installed before HRA can be configured on the destination server. If NPS on the destination server will only be used with HRA, you can use the Add Roles and Features Wizard in Server Manager to install both HRA and NPS role services together. Following service installation, see the Migrate Network Policy Server to

Windows Server 2012 for procedures to migrate NPS settings to the destination server. When you have completed migration of NPS, continue performing the procedures in this guide to complete HRA migration.

This topic includes sample Windows PowerShell cmdlets that you can use to automate some of the procedures described. For more information, see Using Cmdlets.

Migrating settings from the source server

Use the following procedures to export the HRA settings from your x86-based or x64-based source HRA server prior to migrating to an x64-based server running Windows Server 2012.

Important

If your migration plan involves configuring the destination server with the same host name as the source server, then the source server must be decommissioned and taken offline prior to joining the destination server to the domain. To eliminate downtime in this scenario, a secondary HRA server should already be deployed before proceeding. For information about deploying a new HRA server, see Install HRA using the Add Roles and Features Wizard.

To export settings from the source server

1. On the source HRA server, type the following command at an elevated command prompt, and then press ENTER:

   ```
   netsh nap hra export filename=c:\hra_export.xml
   ```

2. Copy the *hra_export.xml* file from the *c:*\ directory to the migration file storage location you have chosen.
3. Configuration settings for the NPS role service must also be exported from the source server. Use the procedures provided in the **Migrating settings from the source server** section of the NPS Server Migration: Migrating the NPS Server topic to export these settings.
4. Copy the exported HRA configuration file to the migration file storage location you have chosen.

Configuring the destination server

Use the following procedures to configure the destination with the required identity, certificates, and services. If the destination server will have a different host name and IP address from the source server, then the source server can remain online and in service until testing and verification of the destination server is complete. When you have completed configuring the destination server's identity, certificates, and services, you can begin migrating HRA settings from the source to destination server.

Important

Some services and settings on the destination server might already be migrated due to the migration of prerequisite roles. Before you configure the destination HRA server, consult the **Migrating prerequisite roles** topic in this guide to determine the configuration settings for NPS, AD CS, and IIS that must be migrated first.

To configure the destination server

1. Add the destination server to the domain of the source server. If the destination server will use the same name as the source server, you must ensure the source server is decommissioned as described in the **Impact of migration** topic.
2. Add the destination server to all security groups and organizational units (OUs) of which the source HRA server is a member. In most cases, the HRA server is a member of the IPsec boundary OU. Members of the boundary OU typically have IPsec policies applied that allow communication with both compliant and noncompliant computers. For more information on OUs and required IPsec policy settings, see Checklist: Deploy IPsec Policies for NAP (http://go.microsoft.com/fwlink/p/?linkid=229649).
3. To update Group Policy settings on the destination server, run the following command at an elevated command prompt:

   ```
   gpupdate /force
   ```

   Note
   To apply new security group membership settings, you must restart the destination server.
4. If client computers will use SSL to request health certificates from HRA, you must provision the destination server with an SSL certificate. For more information, see Configure an SSL Certificate for HRA (http://go.microsoft.com/fwlink/p/?LinkId=229650), or use the process defined within your organization for provisioning an SSL certificate.
5. Install the HRA role service on the destination server.

Install HRA using the Add Roles and Features Wizard

a. In Server Manager, click **Manage** and click **Add Roles and Features**.
b. On the **Before you begin** page, click **Next**.
c. On the **Select Installation Type** page, click **Role/Feature Based Install** and then click **Next**.
d. On the **Select destination server** page, click **Select a server from the server pool**, click the names of the servers where you want to install HRA and then click **Next**.
e. On the **Select server roles** page, click **Network Policy and Access Services**, and then click **Next** three times.

   Note
   If the Network Policy Server role service is already installed, expand the NPAS node and select **Health Registration Authority**. Click

Next five times and continue with step below.

f. On the **Select Role Services** page, click **Health Registration Authority**, and in the **Add Roles and Features Wizard** dialog box, verify that **Include management tools (if applicable)** is selected, click **Add Features**, and then click **Next** five times.

g. On the **Certification Authority** page, choose **Select a CA later using the HRA console**, and then click **Next**.

> **Note**
> Certification Authority settings for HRA will be configured when you migrate settings from the source server.

h. On the **Authentication Requirements** page, choose **No, allow anonymous requests for health certificates**, if the destination HRA will provide health certificates to workgroup computers. If health certificates will be issued to domain-joined clients only, choose **Yes, require requestors to be authenticated as members of a domain (recommended)**. Click **Next** to continue.

i. On the **Server Authentication Certificate** page, click **Choose an existing certificate for SSL encryption (recommended)**, click the certificate displayed under this option, and then click **Next**. If multiple certificates are displayed, or you are not sure if the certificate displayed can be used for SSL encryption, see Install the HRA Role Service for more information.

j. Click **Next**, and then click **Install**.

k. On the **Installation Results** page, verify that installation was successful and then click **Close**.

The following Windows PowerShell command performs the same function:

```
Add-WindowsFeature NPAS-Health
```

Migrating settings to the destination server

Follow the procedure below to migrate HRA settings from the source to destination server.

To migrate the settings to the destination server

1. On the destination server, type the following command at an elevated command prompt, and then press ENTER:

   ```
   netsh nap hra import filename = c:\hra_export.xml
   ```

   Replace *c:\hra_export.html* with the path and file name of the HRA configuration file that you exported in the previous procedure: Migrating settings from the source server.

   > **Note**
   > If you receive the error message “Cannot create a file when that file already

exists," reset the HRA configuration and then perform this procedure again. To reset the HRA configuration, type the following command at an elevated command prompt and then press ENTER: **reg delete HKLM\Software\Microsoft\HCS\CAServers**.

2. Verify that the settings have been imported successfully. To review HRA settings, type the following command at a command prompt and then press ENTER:

```
netsh nap hra show configuration
```

3. If the name of the certification authority will change as a result of the migration, type the following commands at an elevated command prompt to add the name of the correct CA and delete the name of the old CA. Replace *\\srv1.woodgrovebank.com\woodgrovebank-srv1-CA* and *1* with the name and processing order of the CA you wish to use.

```
netsh nap hra delete caserver name =
"\\srv1.woodgrovebank.com\woodgrovebank-srv1-CA"
netsh nap hra add caserver name =
"\\srv2.woodgrovebank.com\woodgrovebank-srv2-CA"
processingorder = "1"
```

You can use the output of the **netsh nap hra show configuration** command to view the name and processing order format for the previous CA. For more information, see HRA Certification Authority Commands.

Configuring the Certification Authority

The destination HRA server name must be given security permissions to request, issue, and manage certificates. It must also be granted permission to manage the CA so that it can periodically clear expired certificates from the certificate store.

If the host name of the destination server is different from the source server, then the certification authority for the NAP deployment must be configured with permissions settings for the new HRA. If the destination HRA server is already a member of an OU or group that has permissions to manage the NAP CA, then this procedure is not required.

To configure the Certification Authority with permissions for the destination HRA

1. On the **Start** screen, type **certsrv.msc**, and then press ENTER on the CA server.
2. In the Certification Authority console tree, right-click the CA name, and then click **Properties**.
3. Click the **Security** tab, and then click **Add**.
4. Click **Object Types**, click the **Computers** check box, and then click **OK**.
5. If the CA is located on a different computer than the destination HRA server, type the name of the destination HRA server under **Enter the object names to select**, and then click **OK**.

Note

If the CA is installed on the same computer as the destination HRA server, type **NETWORK SERVICE** under **Enter the object names to select**, and then click OK.

6. Click the name of the destination server, or click **NETWORK SERVICE**, select **Allow** for the **Issue and Manage Certificates**, **Manage CA**, and **Request Certificates** check boxes, and then click **OK**.
7. Close the Certification Authority console.

Configuration tips for migrating the Certification Authority

If the HRA uses a CA that was recently migrated in parallel using the Active Directory Certificate Services Migration Guide (http://go.microsoft.com/fwlink/?LinkID=156771), consider the following:

1. If the HRA uses an Enterprise CA that was recently migrated, the template for the System Health Authentication certificate used by the HRA must be re-issued in Active Directory before it can be used. This procedure is described in the Restoring the certificate templates list section of the AD CS Migration: Migrating the Certification Authority topic and in the Backing up a CA templates list procedure of the AD CS Migration: Preparing to Migrate topic in the Active Directory Certificate Services Migration Guide (http://go.microsoft.com/fwlink/?LinkID=156771).
2. If the HRA uses a Root CA that was recently migrated, then all NAP IPsec policies configured in Group Policy need to be edited to use the correct Root CA. For more information, see Configure IPsec GPOs.

See Also

Migrate Health Registration Authority to Windows Server 2012

HRA Server Migration: Preparing to Migrate

HRA Server Migration: Verifying the Migration

HRA Server Migration: Post-migration Tasks

Network Access Protection Design Guide

Network Access Protection Deployment Guide

HRA Server Migration: Verifying the Migration

After the migration of your Health Registration Authority (HRA) server is complete, you can perform some tasks to verify that the migration was successful.

Verifying HRA Functionality

In order to verify the HRA functionality, the URL of the destination server must be configured in the NAP client trusted server group settings. This is typically done using Group Policy.

To test the destination server with minimal impact to your current NAP deployment, you can add a secondary trusted server group to NAP client settings. The new trusted server group can contain the URL of the newly migrated destination server. When a secondary trusted server group is configured, compliant client computers will receive a health certificate from both the source HRA and the destination HRA. Once you have verified that client computers are successfully receiving health certificates from the destination server, the new trusted server group can be removed, and the original trusted server group can be updated to use the destination server instead of the source server.

Adding a new trusted server group for testing

To add a new trusted server group in group policy that will be used to test the destination HRA, see Configure Trusted Server Groups in Group Policy.

The new trusted server group should be ordered below any other groups configured, and only the URL of the destination server (for example: *https://destination.contoso.com/domainhra/hcsrvext.dll*) should be added.

Note

If there are multiple GPOs for NAP clients in your organization, you can make these changes to one GPO that applies to a group of clients you wish to test.

Testing the HRA with a NAP client

Use the following procedure to test the functionality of the destination server using a domain-joined NAP client in your deployment.

To test the HRA functionality using a NAP client

1. On the client computer, On the **Start** screen, type **gpupdate /force**, and then press ENTER. This updates the Group Policy configuration for the client.
2. On the **Start** screen, type **cmd**, type **netsh nap client show grouppolicy**, and then press ENTER.
3. In the command output, under **Enforcement clients**, verify that the **Admin** status of the **IPSec Relying Party** is **Enabled**.
4. In the command output, under **Trusted server group configuration**, verify that the trusted server group and destination server URL you configured previously are displayed.
5. Next, the NAP Agent service will be restarted to verify that the client computer successfully receives a health certificate from the new destination HRA.
6. To restart the NAP Agent service, at the command prompt, type `net stop napagent && net start napagent`, and then press ENTER. Verify that the commands completed

successfully.

7. At the command prompt, type `eventvwr.msc`, and then press ENTER. This launches the Event Viewer.
8. In Event Viewer, browse to **Windows Logs /Application and Services Logs/Microsoft/Windows/Network Access Protection/Operational**.
9. In the details pane, under **Event ID**, locate the most recent occurrences of event 22. Event 22 is displayed each time a client computer acquires a health certificate from HRA. Double-click these events to review detailed information about the certificate acquisition. Verify that the URL of the destination server is displayed in at least one event as the source of the certificate.
10. Close Event Viewer.

See Also

Migrate Health Registration Authority to Windows Server 2012

HRA Server Migration: Preparing to Migrate

HRA Server Migration: Migrating the HRA Server

HRA Server Migration: Post-migration Tasks

Network Access Protection Design Guide

Network Access Protection Deployment Guide

HRA Server Migration: Post-migration Tasks

After all migration steps are complete and you have verified the migration of the Health Registration Authority (HRA) role service, perform the following post-migration tasks.

Deploying final client settings

To finish deploying the destination server, all NAP clients must be updated to obtain a health certificate from the destination server URL instead of the source server URL (if different). These settings are typically configured using Group Policy. If the source and destination URLs are different, each GPO in your NAP deployment that uses the new trusted server group settings must be modified. If your organization uses other methods to push NAP client settings to clients, then you might also need to modify those procedures.

Warning

If you have configured HRA automatic discovery on your network and the name of your source and destination HRA servers are different, you must modify DNS service (SRV) records to deploy the new trusted server group setting to client computers. For more information, see Configure HRA Automatic Discovery.

To configure final NAP client settings in group policy

1. On a domain controller or member server with the Group Policy Management feature installed, click **Start**, click **Run**, type **gpmc.msc**, and then press ENTER.
2. In the Group Policy Management console tree, open **Group Policy Objects**, right-click the name of the GPO you want to edit, and then click **Edit**. The Group Policy Management Editor opens.
3. In the console tree, open **Computer Configuration/Policies/Windows Settings/Security Settings/Network Access Protection/NAP Client Configuration/Health Registration Settings/Trusted Server Groups**.
4. Delete the secondary trusted server group that was added for testing purposes. To delete this group, right-click the name of the trusted server group, and click **Delete**.
5. Double-click the name of the primary trusted server group you wish to edit.
6. Click the URL of the source server in the list, and then click **Edit**.
7. Replace the source server URL with the destination server URL.
8. Click **OK**.
9. In the console tree, right-click **NAP Client Configuration**, and then click **Apply**.
10. Close the Group Policy Management Editor window.
11. If you are prompted to apply settings, click **Yes**.
12. Repeat the testing procedure as described in HRA Server Migration: Verifying the Migration to verify that deployment of the destination server is successful.

Restoring the role in the event of migration failure

If the destination server is deployed simultaneously with the source server using a different host name, then the configuration prior to migration can be restored by changing the NAP client settings to use the URL of the source HRA server. To restore the previous configuration, perform the steps described in the **Deploying final client settings** section of the HRA Server Migration: Verifying the Migration topic, replacing the destination server URL with the source server URL.

If the destination server replaced the source server using the same host name, then the destination server will need to be renamed, unjoined from the domain, or otherwise decommissioned in order to bring the source server back online.

Retiring the Source Server

Once the destination HRA has been configured, tested, and verified, and the URL of the source HRA has been removed from group policy, then the HRA role on the source server may be retired.

- The source server can be taken offline and physically retired or repurposed. Follow your organization’s policy regarding server decommissioning.

- To retire only the HRA role on the source server, in the Server Manager console tree, click **Network Policy and Access Services**. In the details pane, click **Remove Role Services**, and then use the Remove Role Services wizard to select and remove the HRA role service.

Note
If the source server was configured to use a certification authority on a different machine, consider removing permissions for the source server from the certification authority.

Troubleshooting migration

If you encounter problems while verifying the HRA migration, see Fixing Health Certificate Problems in the NAP Troubleshooting Guide for help troubleshooting these problems.

See Also

Migrate Health Registration Authority to Windows Server 2012

HRA Server Migration: Preparing to Migrate

HRA Server Migration: Migrating the HRA Server

HRA Server Migration: Verifying the Migration

Network Access Protection Design Guide

Network Access Protection Deployment Guide

Migrate Hyper-V to Windows Server 2012 from Windows 2008 R2

Hyper-V enables you to create a virtualized server computing environment using a technology that is part of Windows. This guide provides information and instructions about migrating the Hyper-V role—including virtual machines, data, and operating system settings—from the source server running Hyper-V in an earlier version of Windows to the destination server that is running the Windows Server® 2012 operating system.

About this guide

Note
Your detailed feedback is very important, and helps us to make Windows Server Migration Guides as reliable, complete, and easy to use as possible. Please take a moment to rate this topic, and then add comments that support your rating. If you are viewing this topic in Lightweight View, click **Rate this topic** at the top of the page. In Classic View, click the stars in the upper-right corner of the page (1=poor, 5=excellent). Describe what you liked, did not like, or want to see in future versions of the topic. To

submit additional suggestions about how to improve Migration guides or utilities, post on the Windows Server Migration forum.

This guide describes how to migrate the Hyper-V role by providing preparation, migration, and verification steps.

Migration documentation and tools ease the migration of server role settings and data from an existing server to a destination server that is running Windows Server 2012. By using the tools that are described in this guide, you can simplify the migration process, reduce migration time, increase the accuracy of the migration process, and help to eliminate possible conflicts that might otherwise occur during the migration process. For more information about installing and using the migration tools on both source and destination servers, see Install, Use, and Remove Windows Server Migration Tools.

Target audience

This document is intended for information technology (IT) professionals who are responsible for operating and deploying Hyper-V in a managed environment.

What this guide does not provide

The following items are not covered in this guide because they are not supported by the migration tools:

- Clustering scenarios are not supported by this migration process. For information about how to perform a migration in a clustered environment, see the Migrating Clustered Services and Applications to Windows Server 2012 Step-by-Step Guide Migrating Clustered Services and Applications to Windows Server 2012.
- Upgrading roles on the same computer is out of scope for this guide.
- Migrating more than one server role at one time.
- Migrating Hyper-V from one server running Windows Server 2012 to another server running 2012. Instead, this process is supported by several of the new Hyper-V management tools and features. The general process is as follows:
    - Determine whether you will use export and import or live migration to move the virtual machines. Export and import can be used in either a workgroup or a domain environment but requires that the virtual machine is turned off. Live migration requires a domain environment as well as some configuration, but allows you to move running virtual machines.
    - Add the Hyper-V role to the destination server. You can configure default storage locations and live migration when you add the role. For instructions, see Install Hyper-V and Configure a Virtual Machine.
    - Configure virtual switches and, optionally, other networking features on the destination server. Management tools include the cmdlets New-VMSwitch and Set-VMSwitch in the Hyper-V module, and the Virtual Switch Manager in the Hyper-V Manager snap-in.
    - Move the virtual machines by exporting and importing them, or performing live migrations. Management tools include the cmdlets Export-VM and Import-VM, and the

Export, **Import**, and **Move** menu commands in Hyper-V Manager. For more information about using live migration to move a virtual machine, see Configure Live Migration and Migrating Virtual Machines without Failover Clustering.

- For a list of the cmdlets included in the Hyper-V module, see http://technet.microsoft.com/library/hh848559.

Supported migration scenarios

This guide provides you with instructions for migrating an existing server that is running the Hyper-V role on an earlier version of Windows Server to a server that is running Windows Server 2012. This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, it is recommended that you design a custom migration procedure specific to your server environment, based on the information provided in other role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server 2012.

Caution

If your source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other roles that are running on the source server to fail.

Supported operating systems

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x64-based	Windows Server 2008 with Service Pack 2, full installation option only	Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2012, both full and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2008 R2	Windows Server 2012, both full and Server Core installation options	x64-based

The versions of operating systems shown in the preceding table are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are

supported. If an operating system is not listed, then it is not supported. The stand-alone product Microsoft Hyper-V Server is not supported.

Standard, Enterprise, and Datacenter editions of Windows Server running Hyper-V are supported as either source or destination servers.

Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shares from a computer that is running Windows Server 2008 in the French system UI language to a computer that is running Windows Server 2012 in the German system UI language.

Note
The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Supported role configurations and settings

This section identifies the configurations and settings that can be migrated by using the migration tools, and the configurations and settings that must be migrated manually. The following table provides a summary.

Configurations and settings	Type of migration
Virtual machine (configuration and data)	Automated, except as noted below
Hyper-V settings	Automated
Virtual network adapter settings in the management operating system	Automated
External virtual networks	Partially automated, as described below
Virtual machine queue (VMQ) networking settings	Automated
Customized remote administration settings	Manual

The following configurations and settings can be migrated automatically:

- **Most virtual machine configurations**. Virtual machines and their data are moved as part of the migration, but some configurations require manual intervention, as described below.
- **Hyper-V settings**. These include the system-wide settings and the authorization store.

  Note
  If you are migrating from a source server running Windows Server 2008 R2 and have set a MAC address range, that value also is automatically migrated to the destination server.

- **Internal and private virtual networks**.
- **Virtual network adapter settings in the management operating system**. When Hyper-V is configured to use a physical network adapter as a bridge that virtual machines can use to access a physical network, a virtual network adapter is created in the management operating system (which runs the Hyper-V role). For this virtual network adapter, the migration process automatically migrates the IP settings, bindings, and MAC address of this virtual network adapter. However, the connection between the virtual network adapter and the physical network adapter must be re-established manually, as described in the migration steps.
- **Virtual machine queue (VMQ) settings for networking**.

The following configurations and settings require manual intervention after the migration tools are used:

- **Firewall settings**. Firewall settings are recreated on the destination server using the default values that Hyper-V is installed with. If you have modified any of the firewall settings from these default values, you will need to make the same modifications on the destination server.
- **External virtual networks**. The migration tool recreates the virtual networks on the destination server, but recreates external virtual networks as internal virtual networks. You will need to modify each of these networks to connect it to the appropriate physical network adapter on the destination server, as described in the migration steps.
- **VFD and ISO files**. These files are not migrated because they are not required for the virtual machine to operate and are not supported by the Import and Export cmdlets. To make them available to a migrated virtual machine, manually copy these files to the destination server and then reattach them to the virtual machine after it is migrated.
- **Connections to physical disks directly attached to virtual machines**. These connections (sometimes referred to as “pass-through disks”) are not migrated because the disk references might not be valid on the destination server. To make a physical disk available to a migrated virtual machine, connect the disk to the destination server and then to the virtual machine after it is migrated, as described in the migration steps.
- **Customized remote administration settings**. If you have customized Hyper-V for remote access, you will need to perform some additional procedures to recreate the DCOM and WMI Namespace settings. The migration steps identify the point at which you should take perform these procedures, as well as provide a recommended tool or script to complete the procedure.

Migration dependencies

The Hyper-V role is not dependent on any other roles. As a best practice, we recommend that no other roles are installed on a server running Hyper-V.

Migration scenarios that are not supported

The following migration scenarios are not supported:

- The saved state of a virtual machine.
- Virtual machine configuration under one of the following conditions:

    - When the number of virtual processors configured for the virtual machine is more than the number of logical processors on the destination server.
    - When the memory configured for a virtual machine is greater than the available memory on the destination server.
- Consolidation of physical servers to virtual machines, or consolidation of multiple instances of Hyper-V to one instance.

Hyper-V migration overview

Hyper-V role migration involves moving the virtual machines, virtual networks, and all the associated settings from one physical computer to another physical computer in the enterprise. The process supports moving from a server running Hyper-V in Windows Server® 2008 R2 to a server running Hyper-V in Windows Server 2012. The Hyper-V role is not dependent on any other roles.

The migration tools include cmdlets that you use to perform some of the tasks required to migrate the Hyper-V role. The Export cmdlet captures the majority of the Hyper-V settings that are required to perform a successful migration, including the virtual machine configurations, virtual networks, and virtual hard disks. The DCOM and WMI namespace security settings must be migrated separately. The instructions for this are provided later in the guide. On the destination server, the import cmdlets will recreate the virtual machines.

Impact of migration

The following section describes the impact of migration on the source server and on other computers in the enterprise.

Impact of migration on the source server

The source server should be turned off or removed from the network before you run the import cmdlets on the destination server so that there are no conflicts between the virtual machines running on the source server and the virtual machines that will be recreated on the destination server. The point at which you should perform this task is identified in the migration steps, later in this guide.

Impact of migration on other computers in the enterprise

This migration may impact any computer (either virtual or physical) that relies on the applications or workloads running in the virtual machines to be migrated as part of the Hyper-V role migration, because the virtual machines will be offline for the duration of the migration. For example, if a virtual machine hosts a database, any applications in the enterprise that require access to that database will be impacted. As a result, you will need to plan for this downtime by either scheduling a planned outage or by redirecting traffic to other servers to provide the services.

Access rights required to complete migration

The user account that runs the cmdlets and tools must be a member of the local Administrators group on the source server and the destination server.

Estimated duration

The length of time it takes to migrate the Hyper-V role depends on the size of the data to be transferred. Of the various types of files to be transferred, the .vhd files have the largest file sizes (from a few gigabytes to many gigabytes in size). The length of time is affected by the size of the .vhd files and by the network bandwidth.

Additional references

- Hyper-V: Prepare to Migrate
- Hyper-V: Migrate the Hyper-V Role
- Hyper-V: Verify the Migration
- Hyper-V: Post-migration Tasks
- Hyper-V Overview
- Hyper-V Migration Guide (for migration of Hyper-V running in Windows Server 2008 to Windows Server 2008 R2.)
- Windows Server Migration Portal

Hyper-V: Prepare to Migrate

Follow these steps to prepare for migration.

Select and prepare your destination server

To select and prepare the destination server for migration, perform the steps in the order they are given.

Hardware requirements for the destination server

The computer you select as the destination server must meet the following hardware requirements:

- **Storage**. The destination server requires enough storage to hold the virtual hard disks from the source server.
- **Network**. The destination server requires at least as many physical network adapters as the number of physical network adapters in use as external virtual networks on the source server. To determine this, open Hyper-V Manager on the source server, and then open Virtual

Network Manager. Under **Virtual Networks** (in the left pane) note the number of the networks designated as “External”.

- **Memory**. The destination server requires enough memory to run all the virtual machines you plan to run at the same time, as well as run the Hyper-V role. For example, if you run all the virtual machines configured on the source server at the same time, the destination server must have at least as much memory as the sum of memory configured for all virtual machines, plus memory to run the Hyper-V role in Windows Server® 2012.
- **Processor**. The destination server requires at least as many logical processors as the largest number of processors configured on a virtual machine on the source server. Note that if you want to migrate virtual machines with saved states, the processor on the destination server must be compatible with the processor on the source server. The processors must be from the same manufacturer and have compatible steppings.

Software requirements for the destination server

After you select the destination server, prepare the software by doing the following:

Important
If you install Windows Server Migration Tools before you install the Hyper-V role, you must remove the tools and then install the Hyper-V role before you install the tools. For removal instructions, see Install, Use, and Remove Windows Server Migration Tools.

1. Install Windows Server 2012. For more information, see Installing Windows Server 2012.
2. Add the Hyper-V role. For instructions, see Install Hyper-V and Configure a Virtual Machine.

Back up your source server

Before you start migration, back up the source server. If the migration fails, you can use this backup to restore the source server. For information about the different types of backups, see Planning for Backup (http://go.microsoft.com/fwlink/?LinkId=178128).

Install migration tools

Windows Server Migration Tools in Windows Server® 2008 R2 allows an administrator to migrate some server roles, features, operating system settings, shares, and other data from computers that are running certain editions of Windows Server 2003, Windows Server 2008, or Windows Server 2008 R2 to computers that are running Windows Server 2008 R2 or Windows Server 2012.

Complete installation, configuration, and removal instructions for Windows Server Migration Tools are available on the World Wide Web, in Install, Use, and Remove Windows Server Migration Tools. Windows Server Migration Tools must be installed on both the destination server and the source server, in that order.

Migration documentation and tools ease the process of migrating server role settings and data from an existing server that is running Windows Server 2003 and later releases of the Windows operating system to another computer. By using these tools to migrate roles, you can simplify

migration, reduce migration time, increase accuracy of the migration process, and help eliminate conflicts that could otherwise occur during the migration process.

Windows Server Migration Tools is a set of Windows PowerShell® cmdlets. For more information about Windows PowerShell and working with cmdlets, see Windows PowerShell Core on Microsoft TechNet.

Collect configuration details from your source server

Collect the following configuration details about the source server. You will use this information as part of the verification process after you perform the migration.

- Gather identifying information about the set of virtual machines on the source server. If there is a relatively small number of virtual machines on the server (for example, less than 20), you could take a screenshot of the list of virtual machines displayed in the Hyper-V Manager snap-in. Additionally, record the following configuration information for each virtual machine:
  - Amount of memory
  - Number of virtual processors
  - Virtual hard disks (.vhd files) connected to the virtual machine
- For each virtual machine that has snapshots, gather information about the number of snapshots and the structure of the snapshot tree.
- Record information about the external virtual networks. Include information such as the name of each external virtual network. If you plan to migrate the IP settings of the physical network adapters (for example, if the network adapters use static IP addresses that you want to retain), save the IP configuration settings by using the following command:

  ```
  IPConfig /all > IPSettings.txt
  ```

- If you have made any customizations to the Hyper-V security policy, gather all the information about scopes and roles from Authorization Manager (see Using Authorization Manager for Hyper-V Security (http://go.microsoft.com/fwlink/?LinkId=183469)).
- If you have turned off any exceptions in Windows Firewall, record that information.
- If you have granted remote access to the server for Hyper-V management to any user account that is not a member of the Administrators group, record that information. The Hyper-V Remote Management Configuration Utility is a tool is that you can use for this task. However, this tool is not published or supported by Microsoft. For more information, see Hyper-V Remote Management Configuration Utility (http://go.microsoft.com/fwlink/?LinkId=178138).
- If you have granted remote access to the server for Hyper-V management to any user account that is not a member of the Administrators group or Distributed COM Users group, open the registry and navigate to the HKLM\Software\Microsoft\OLE\ key. Find the MachineLaunchRestriction value and record all the information about that value.

  Caution

Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on the computer.

Prepare other computers in the enterprise

Important

Before you run the **Import-SmigServerSetting**, **Export-SmigServerSetting**, or **Get-SmigServerFeature** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with domain users or groups who are members of local groups on the source server.

Before you run the **Send-SmigServerData** or **Receive-SmigServerData** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with those domain users who have rights to files or shares that are being migrated.

Depending on the workloads that you have deployed in your virtual machines, you need to take the necessary actions to ensure that the users and clients that obtain services from the virtual machines are not negatively impacted during the migration. In other words, you could either have a planned downtime or redirect the clients and users to alternate, redundant virtual machines while the migration is in progress. The specific actions you take depend on the best practices you have in place for the workloads deployed in the virtual machines.

Additional references

1. Migrate Hyper-V to Windows Server 2012 from Windows 2008 R2
2. Hyper-V: Migrate the Hyper-V Role
3. Hyper-V: Verify the Migration
4. Hyper-V: Post-migration Tasks
5. Hyper-V Overview
6. Hyper-V Migration Guide (for migration of Hyper-V running in Windows Server 2008 to Windows Server 2008 R2.)
7. Windows Server Migration Portal

Hyper-V: Migrate the Hyper-V Role

Migrate the Hyper-V Role

The steps to migrate the Hyper-V role are the same for all of the scenarios defined in “Supported migration scenarios,” earlier in this guide.

Perform migration steps on the source server

To perform migration steps on the source server

1. Prepare the virtual machines for migration by shutting down the virtual machines.
2. Open a Windows PowerShell session with elevated user rights by doing one of the following:
   - To run Windows PowerShell as an administrator from the **Start** screen, right-click the **Windows PowerShell** tile, and in the app bar, click **Run as administrator**.
   - To run Windows PowerShell as an administrator from the desktop, right-click the **Windows PowerShell** shortcut in the taskbar, and then click **Run as Administrator**.
3. Load Windows Server Migration Tools into your Windows PowerShell session.

   Only load the Windows Server Migration Tools snap-in into a Windows PowerShell session that was opened by using some other method (and into a session where it has not already been loaded). To load Windows Server Migration Tools, type the following, and then press ENTER.

   ```
   Add-PSSnapin Microsoft.Windows.ServerManager.Migration
   ```
4. From Windows PowerShell, collect data from the source server by running the **Export-SmigServerSetting** cmdlet as an administrator. The **Export-SmigServerSetting** cmdlet creates an XML file, StoragePathMappings.xml, that contains information about where the virtual machine storage (.vhd and .avhd) files are stored, in the form of folder paths on the source server. The import process uses the StoragePathMappings.xml file to associate the storage files to the appropriate virtual machines on the destination server. If the destination server will use the same drive mapping and folder structure for virtual machine storage as the source server, you do not need to edit the file after the **Export-SmigServerSetting** cmdlet creates it. Otherwise, you must edit this file before you import it to the destination server. Before you run this command, review the following information to determine the information to include in the command:
   - Determine where to store StoragePathMappings.xml. The **-Path** parameter is a required parameter that specifies the location. You can either choose a location that can be accessed by both the source server and the destination server, such as a network location, or copy it from the source server to the destination server.
   - To export user groups, which are used for access control by the Hyper-V security policy and the Hyper-V remote management tools, include the **-User** and **-Group** parameters:

     ```
     -User <Enabled | Disabled | All> -Group
     ```
   - To recreate the same IP settings on the physical network adapters on the destination server as are configured on the source server, include the **-IPConfig** parameter. The **-IPConfig** parameter collects IP information when it is used with the **Export-SmigServerSetting** cmdlet on the source server. The **-IPConfig** parameter applies settings when the **Import-SmigServerSetting** cmdlet is used on the destination server.

     ```
     -IPConfig
     ```

5. After you determine the parameters, run the **Export-SmigServerSetting** cmdlet, where <storepath> specifies the path to the folder where the configuration data file (Svrmig.mig) will be stored. (For example, C:\Migration.) The StoragePathMappings.xml file is created in a subfolder of the <storepath> folder, named VirtualMachines. (For example, C:\Migration\VirtualMachines.)

```
Export-SmigServerSetting -FeatureId Hyper-V -IPConfig -User
All -Group -path <storepath> -Verbose
```

Migrate virtual machine data

The following steps show you how to use Robocopy and a Windows PowerShell script to copy the data from the source server to the destination server. The script parses the folder paths specified in the StoragePathMappings.xml to migrate the data. You can use the StoragePathMappings.xml file stored under <storepath> as a reference to determine which folders need to be transferred.

Important
If you want to use a different drive mapping and/or folder structure on the destination server, edit the StoragePathMappings.xml file before you attempt to migrate the data to the destination server.

To migrate virtual machine data

1. Copy the data from the source server to the destination server. The recommended way to do this is to use the Robocopy command. You can run the command for each file and specify the source and destination locations. (You can use the StoragePathMappings.xml file to determine the source paths.) Or, you can automate this process by using a Windows PowerShell script. Before you run the script, update the StoragePathMappings.xml file with the locations where you want to paste the files on the destination server. The script can parse the StoragePathMappings.xml file and then call the Robocopy command to copy and paste the files.

   The following is an example of such a script. To use this sample, copy the code and paste it into a text editor, then save the file with a .ps1 file name extension in a directory where you want to run the script from. For example, save CopyData.ps1 to C:\migration\.

```
param(
    [string]$xmlFilePath = $(throw "Must pass the fully
qualified file name of Storage Path XML in the command
string"),
    [string]$destinationHost = $(throw "Must pass the
Destination Host Name (NetBiosName), where the files will be
copied to")
)
```

```
Write-Host "XML File Path: " $xmlFilePath
Write-Host "Destination Host Name: " $destinationHost

# Get the content of the XML file
[xml]$xmlFile = Get-content $xmlFilePath

# For each storage path, if the "Copy" attribute is true copy
the files to the destination Host.
foreach ($storagePath in $xmlFile.StoragePaths.storagePath)
{
         if($storagePath.Copy -eq "true")
         {
                # Get the Source directory
                $sourceDirectory = $storagePath.Source

                # Get the Destination directory
                $destinationDirectory =
$storagePath.Destination
                $destinationDirectory = $destinationDirectory
-replace ":","$"
                $destinationDirectory = "\\" +
$destinationHost + "\" + $destinationDirectory

                # Copy the files to the destination host
                robocopy $sourceDirectory
$destinationDirectory /E /XF *.xml /R:5 /W:60 /V
    }
}
```

To run the script, type the full path to the script at the command prompt and pass the fully qualified file name of the StoragePathMappings.xml file (full path and file name) and the name of the destination server as parameters. The file name extension of the script is optional. For more information, see Support for Scripting (http://go.microsoft.com/fwlink/?LinkID=178144).

For example, if you used the folder and file name example shown above, type:

```
c:\migration\copyData.ps1 <XMLPathName>
<DestinationServerName>
```

2. Disconnect the source server from the network so that you avoid any potential MAC address conflicts between the virtual machines on the source and destination servers. MAC address conflicts may impact the availability of the workloads that run on the virtual machines.

Perform migration steps on the destination server

1. If the <storepath> is located anywhere other than locally on the destination server, edit the permissions of the shared folder to grant **Full Control** to the following accounts:
   - The user account that will run the import and export commands. If the same account is used, only one entry is required.
   - The computer account of the source server.
   - The computer account of the destination server.
2. If you used another method instead of the Robocopy command to copy data to the destination server, check the destination folder and delete any .xml files that were copied to that folder.
3. Open a Windows PowerShell session with elevated user rights by doing one of the following:
   - To run Windows PowerShell as an administrator from the **Start** screen, right-click the **Windows PowerShell** tile, and in the app bar, click **Run as administrator**.
   - To run Windows PowerShell as an administrator from the desktop, right-click the **Windows PowerShell** shortcut in the taskbar, and then click **Run as Administrator**.
4. Load Windows Server Migration Tools into your Windows PowerShell session.

   Only load the Windows Server Migration Tools snap-in into a Windows PowerShell session that was opened by using some other method (and into a session where it has not already been loaded). To load Windows Server Migration Tools, type the following, and then press ENTER.

   ```
   Add-PSSnapin Microsoft.Windows.ServerManager.Migration
   ```
5. To import the Hyper-V settings to the destination server, run the **Import-SmigServerSetting** cmdlet and all additional parameters that you used with the **Export-SmigServerSetting**.

   ```
   Import-SmigServerSetting -FeatureId Hyper-V
   <additionalparameters> -path <storepath> -Verbose -Force
   ```

   Additional parameters:

   - To import the same IP settings on the destination server that were on the source server, where <*SourcePhysicalAddress-1*> and <*SourcePhysicalAddress-2*> are comma-separated lists of the physical addresses of the source network adapter, and <*TargetPhysicalAddress-1*> and <*TargetPhysicalAddress-2*> are comma-separated lists of the physical addresses of the destination network adapter, include:

     ```
     -IPConfig All -SourcePhysicalAddress
     ```

```
"<SourcePhysicalAddress1>","<SourcePhysicalAddress2>" -
TargetPhysicalAddress
"<TargetPhysicalAddress1>","<TargetPhysicalAddress2>"
```

- To import the user groups that are used by the Hyper-V security policy and remote administration, include:

```
-User <Enabled | Disabled | All> -Group
```

6. If a failure occurred while running the **Import-SmigServerSetting** cmdlet, review the Setupact.log and Setuperr.log under %localappdata%\SvrMig\Log.
7. Use the information you gathered about physical-to-virtual network connections to establish the connections between the physical network adapters and external virtual switches on the destination server. (Virtual networks are now referred to as virtual switches in Windows Server 2012.) All external virtual networks are migrated to the destination server as internal virtual switches because the import process cannot map virtual switches to physical networks. To establish the connections:
   a. .a. Open Hyper-V Manager. (From the **Start** screen, click the **Hyper-V Manager** tile.)
   b. In the **Action** pane, click **Virtual Switch Manager**.
   c. In the left pane, under **Virtual Switches**, click the name of the first internal switch that you want to convert to an external switch.
   d. In the right pane, under **Connection type**, select **External**. From the drop-down list, select the physical network adapter to use for access to the physical network.
   e. Click **OK** to save the changes and close Virtual Switch Manager, or click **Apply** to save the changes and modify another virtual switch.
8. For each virtual machine that used a physical disk connected directly to the physical computer, establish this connection on the destination server:
   f. .a. Open the Disk Management snap-in and verify that the disk is in an Offline state. If the disk is not in an Offline state, it will not be available when configuring storage for a virtual machine.
   b. Open Hyper-V Manager. (From the **Start** screen, click the **Hyper-V Manager** tile.)
   c. Under **Virtual Machines**, select the virtual machine that you want to connect to the physical disk.
   d. In the **Action** pane, under the virtual machine name, click **Settings**.
   e. In the navigation pane (left pane), click the controller that you want to attach the disk to. If you plan to use the disk as a startup disk, make sure you attach it to an IDE controller. Click **Add**.
   f. On the **Hard Drive** page, select the location on the controller to attach the disk.
   g. Under **Media**, specify the physical hard disk. If the disk does not appear in the drop-down list under **Physical hard disks**, make sure the disk is in an Offline state in Disk Management.
   h. Select **Physical hard disk**, and then click **OK**.
9. Restore any customizations you made to the WMI namespace security settings. For more information, see [WMI namespace security customizations are missing after upgrading to

Windows Server 2008 R2 (http://go.microsoft.com/fwlink/?LinkId=178143).

10. If you turned off any exceptions in Windows Firewall on the source server, turn off those same exceptions on the destination server.
11. If you have granted remote access to the server for Hyper-V management to any user account that is not a member of the Administrators group or Distributed COM Users group, open the registry and navigate to the HKLM\Software\Microsoft\OLE\ key. Add the MachineLaunchRestriction value that you recorded from the source server.

    Caution
    Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on the computer.

Hyper-V: Verify the Migration

After you have completed the migration, perform the following verification steps to ensure that the migration succeeded.

Verify the Hyper-V security policy

Use the Hyper-V security policy information (that you gathered when you prepared for migration) to verify that the roles and scopes are the same on the destination server as the corresponding roles and scopes on the source server.

Verify the networking configuration

Use the information you gathered about the virtual and physical networks to verify the following:

- The virtual networks are the same on the destination server as they were on the source server.
- If you migrated IP settings for the physical network adapters, such as static IP addresses, they are applied to the corresponding network adapter on the destination server.

Verify the configuration and availability of the virtual machines

Perform the following steps to determine whether the migrated virtual machines will operate as expected.

To verify the virtual machines

1. Use the virtual machine information you gathered when you prepared for migration to verify the following:
    a. Check to see that the set of virtual machines on the destination server has all the virtual machines that were on the source server.
    b. For each virtual machine, verify that the state of the virtual machine on the destination server is the same as it was on the source server before the migration.
    c. For each virtual machine, verify that the snapshots it has are identical in number and structure to the snapshots of the corresponding virtual machine on the source server.
    d. Verify that the memory and number of virtual processors are the same as they were on the source server.
    e. Verify that the storage configuration (virtual hard disks and/or physical disks attached directly to the virtual machine) is identical to that on the source server.
2. Start each migrated virtual machine. If a virtual machine does not start, check the event log under Applications and Service Logs\Microsoft\Windows\Hyper-V virtual machineMS\Admin to see why it failed to start. Common reasons for failure include:
    - The virtual machine is not in the correct scope in the authorization policy.
    - The storage is misconfigured. For example, one or more virtual hard disks might not be in the specified location. Check the hard disk settings for the virtual machine to make sure that the path to the .vhd file is valid. If the virtual machine is configured to use a directly attached physical disk, make sure it is attached to the destination server and shows as offline in Disk Management on the server.
    - One or more virtual hard disks do not have the required security permissions in the file system where the .vhd files are stored.
3. Run some basic operations that change the state of each virtual machine to verify that the operations work as expected on the migrated virtual machine. For example, saving and restoring, pausing and resuming, starting and stopping, or taking and applying or deleting snapshots.
4. Delete any of the snapshots you have taken as part of the previous step, turn off the virtual machine to merge the snapshot disks, and then turn on the virtual machine.
5. After the virtual machine has booted into the operating system, run the necessary application-specific tests to ensure that the application on the virtual machine can provide the same service levels as it provided before the virtual machine was migrated.
6. Verify that you can access the desktop of each virtual machine using Remote Desktop or Virtual Machine Connection, if you had access to the desktop on the source server.
7. If the virtual machine passes all of the above tests, it is ready to be put into production.

Hyper-V: Post-migration Tasks

After you have performed the verification steps, you are ready to complete the migration. Completing the migration for Hyper-V consists of either retiring the source server if the migration succeeded, or rolling back the source server to its pre-migration state if the migration failed.

Retiring your source server

If the migration succeeded, you can repurpose the server for another use or retain it as a backup.

Important
We recommend that you remove the Hyper-V role as soon as you verify that the migration succeeded, to avoid unintentionally placing the source server back online, which could result in running duplicate virtual machines on the same network.

Restoring the role in the event of migration failure

If verification was not successful, follow these steps to roll back the migration.

Roll back migration of Hyper-V on the source server

To roll back migration of Hyper-V on the source server

1. Disconnect the destination server from the network.
2. If you removed the Hyper-V role from the source server, add the Hyper-V role.
3. Reconnect the source server to the network.
4. Restart all the virtual machines.

Roll back migration of Hyper-V on the destination server running Windows Server 2012

To roll back migration of Hyper-V on the destination server

1. Delete the migrated virtual machines.

   Important
   Do not delete the migrated data if you plan to retry the migration to the destination server. This will allow you to save time by not having to copy the data files again.

2. Remove the Hyper-V role.

Roll back migration changes on other computers in the enterprise

For each client that depends on workloads running on virtual machines on the source server, verify that the clients can communicate with the virtual machines.

Troubleshooting cmdlet-based migration

The Windows Server Migration Tools deployment log file is located at %*windir*%\Logs\SmigDeploy.log. Additional Windows Server Migration Tools log files are created at the following locations.

- %*windir*%\Logs\ServerMigration.log
- On Windows Server 2008, Windows Server 2008 R2, and Windows Server 2012: %*localappdata*%\SvrMig\Log
- On Windows Server 2003: %*userprofile*%\Local Settings\Application Data\SvrMig\Log

If migration log files cannot be created in the preceding locations, **ServerMigration.log** and **SmigDeploy.log** are created in %*temp*%, and other logs are created in %*windir*%\System32.

If a migration cmdlet fails, and the Windows PowerShell session closes unexpectedly with an access violation error message, look for a message similar to the following example in the %*localappdata*%\SvrMig\Logs\setuperr.log file.

```
FatalError [0x090001] PANTHR Exception (code 0xC0000005: ACCESS_VIOLATION) occurred at
0x000007FEEDE9E050 in C:\Windows\system32\migwiz\unbcl.dll (+000000000008E050).  Minidump
attached (317793 bytes).
```

This failure occurs when the server cannot contact domain controllers that are associated with domain users or groups who are members of local groups, or who have rights to files or shares that are being migrated. When this happens, each domain user or group is displayed in the GUI as an unresolved security identifier (SID). An example of a SID is **S-1-5-21-1579938362-1064596589-3161144252-1006**.

To prevent this problem, verify that required domain controllers or global catalog servers are running, and that network connectivity allows communication between both source and destination servers and required domain controllers or global catalog servers. Then, run the cmdlets again.

If connections between either the source or destination servers and the domain controllers or global catalog servers cannot be restored, do the following.

1. Before you run **Export-SmigServerSetting**, **Import-SmigServerSetting** or **Get-SmigServerFeature** again, remove all unresolved domain users or groups who are members of local groups from the server on which you are running the cmdlet.
2. Before you run **Send-SmigServerData** or **Receive-SmigServerData** again, remove all unresolved domain users or groups who have user rights to files, folders, or shares on

the migration source server.

Viewing the content of Windows Server Migration Tools result objects

All Windows Server Migration Tools cmdlets return results as objects. You can save result objects, and query them for more information about settings and data that were migrated. You can also use result objects as input for other Windows PowerShell commands and scripts.

Result object descriptions

The Windows Server Migration Tools **Import-SmigServerSetting** and **Export-SmigServerSetting** cmdlets return results in a list of **MigrationResult** objects. Each **MigrationResult** object contains information about the data or setting that the cmdlet processes, the result of the operation, and any related error or warning messages. The following table describes the properties of a **MigrationResult** object.

Property name	Type	Definition
ItemType	Enum	The type of item being migrated. Values include **General**, **WindowsFeatureInstallation**, **WindowsFeature**, and **OSSetting**.
ID	String	The ID of the migrated item. Examples of values include **Local User**, **Local Group**, and **DHCP**.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise, **False** is displayed.
DetailsList	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.

Send-SmigServerData and **Receive-SmigServerData** cmdlets return results in a list of **MigrationDataResult** objects. Each **MigrationDataResult** object contains information about the data or share that the cmdlet processes, the result of the operation, any error or warning messages, and other related information. The following table describes the properties of a **MigrationDataResult** object.

Property name	Type	Definition
ItemType	Enum	The type of migrated item. Values include **File**, **Folder**,

Property name	Type	Definition
		Share, and **Encrypted File**.
SourceLocation	String	The source location of the item, shown as a path name.
DestinationLocation	String	The destination location of the item, shown as a path name.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise, **False** is displayed.
Size	Integer	The item size, in bytes.
ErrorDetails	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.
Error	Enum	Errors enumeration for errors that occurred.
WarningMessageList	List <String>	A list of warning messages.

The following table describes the properties of objects within the **MigrationResultDetails** object that are common to both **MigrationResult** and **MigrationDataResult** objects.

Property name	Type	Definition
FeatureId	String	The name of the migration setting that is related to the item. Examples of values include **IPConfig** and **DNS**. This property is empty for data migration.
Messages	List <String>	A list of detailed event messages.
DetailCode	Integer	The error or warning code associated with each event message.
Severity	Enum	The severity of an event, if events occurred. Examples of values include **Information**, **Error**, and **Warning**.

Property name	Type	Definition
Title	String	Title of the result object. Examples of values include NIC physical address for IP configuration, or user name for local user migration.

Examples

The following examples show how to store the list of the result objects in a variable, and then use the variable in a query to return the content of result objects after migration is complete.

To store a list of result objects as a variable for queries

1. To run a cmdlet and save the result in variable, type a command in the following format, and then press **Enter**.

   $*VariableName* **=** $(*Cmdlet*)

   The following is an example.

   $ImportResult = $(Import-SmigServerSetting -FeatureId DHCP -User all -Group -Path D:\rmt\DemoStore -force -Verbose)

   This command runs the **Import-SmigServerSetting** cmdlet with several parameters specified, and then saves result objects in the variable **ImportResult**.

2. After the **Import-SmigServerSetting** cmdlet has completed its operations, return the information contained in the result object by typing a command in the following format, and then pressing **Enter**.

   $*VariableName*

   In the following example, the variable is named **ImportResult**.

   $*ImportResult*

   This command returns information contained in the result objects that were returned by **Import-SmigServerSetting** in the example shown in step 1. The following is an example of the output that is displayed by calling the **ImportResult** variable.

```
            ItemType  ID                                Success
DetailsList
            --------  --                                -------
-----------
          OSSetting  Local User                          True
{Local User, Loc...
          OSSetting  Local Group                         True
{Local Group, Lo...
```

```
        WindowsFeature  DHCP                                    True
{}
```

Each line of the preceding sample is a migration result for an item that was migrated by using the **Import-SmigServerSetting** cmdlet. The column heading names are properties of **MigrationResult** objects. You can incorporate these properties into another command to return greater detail about result objects, as shown by examples in step 3 and forward.

3. To display a specific property for all result objects in the list, type a command in the following format, and then press **Enter**.

   $<VariableName>| Select-Object -ExpandProperty <PropertyName>

   The following is an example.

   $importResult | Select-Object -ExpandProperty DetailsList

4. You can run more advanced queries to analyze result objects by using Windows PowerShell cmdlets. The following are examples.

   - The following command returns only those details of result objects that have the ID **Local User**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList

   - The following command returns only those details of result objects with an ID of **Local User** that have a message severity equal to **Warning**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Severity -eq "Warning") {$_} }

   - The following command returns only the details of result objects with an ID of **Local User** that also have the title **Remote Desktop Users**.

     $ImportResult | Where-Object { $_.ID -eq "Local Group" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Title -eq "Remote DesktopUsers") {$_} }

More information about querying results

For more information about the cmdlets that are used in the preceding examples, see the following additional resources.

- Where-Object on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134853).
- Select-Object on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134858).
- ForEach-Object on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134860)

For more information about Windows PowerShell scripting techniques, see What Can I Do With Windows PowerShell? - Scripting Techniques on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134862).

Migrate IP Configuration to Windows Server 2012

Migration of IP configuration data is a necessity for the migration of some server roles to Windows Server 2012, including DHCP Server, Domain Name System (DNS) Server, and Active Directory Domain Services (AD DS). This guide describes how to migrate core IPv4 and IPv6 configuration settings and data.

Supported operating systems

The following table indicates the Windows Server operating systems that are supported by Windows Server Migration Tools.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 with Service Pack 2	Windows Server 2008 R2, both full and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2003 R2	Windows Server 2008 R2, both full and Server Core installation options	x64-based
x86- or x64-based	Full installation option of Windows Server 2008	Windows Server 2008 R2, both full and Server Core installation options	x64-based
x64-based	Windows Server 2008 R2, both full and Server Core installation options	Windows Server 2008 R2, both full and Server Core installation options	x64-based
x64-based	Windows Server 2012, both full and Server Core installation options	Windows Server 2012, both full and Server Core installation options	x64-based

The versions of operating systems shown in the preceding table are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are supported.

Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported as either source or destination servers.

Migrations between physical operating systems and virtual operating systems are supported.

Windows Server Migration Tools does not support migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shares from a computer that is running Windows Server 2008 in the French system UI language to a computer that is running Windows Server® 2012 in the German system UI language.

Note
The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Both x86- and x64-based migrations are supported for Windows Server 2003, Windows Server 2008 R2 and Windows Server 2012. All editions of Windows Server 2008 R2, and Windows Server 2012 are x64-based.

Roles that are running on Server Core installations of Windows Server 2008 cannot be migrated, because there is no .NET Framework available on Server Core installations of Windows Server 2008.

Supported scenarios and features

Windows Server Migration Tools supports migration of the following frequently used IP configuration settings and data. Settings are migrated in the order in which they are listed in the Windows interface. For example, DNS server settings are migrated in the order in which they are used.

Setting type	Supported settings and notes
Manually-configured IP settings for all enabled network adapters (also known as network interface cards, or NICs) that are connected to the network	IPv4 addresses
	IPv4 subnet mask
	IPv4 DHCP status
	IPv4 default gateway addresses (but not gateway metrics)
	IPv4 interface metric
	IPv4 Windows Internet Name Service (WINS) server settings

Setting type	Supported settings and notes
	• WINS server addresses • NetBIOS setting (Default, Enable NetBIOS over TCP/IP, or Disable NetBIOS over TCP/IP)
	IPv6 addresses and corresponding subnet prefix lengths **Note** Migration of IPv6 subnet prefix lengths is supported only on destination servers that are running Windows Server 2008, Windows Server 2008 R2 or Windows Server® 2012. If an IPv6 address is imported from a Windows Server 2003 migration store, the subnet prefix length is set to the default value of 64 on the destination server.
	IPv6 router discovery setting
	IPv6 **Managed address configuration** flag, and **Other stateful configuration** flag **Notes** These settings are supported only on Windows Server 2008, Windows Server 2008 R2 and Windows Server® 2012. Windows Server 2003 does not support DHCPv6-based IPv6 address configuration. If **Router Discovery** is enabled on the source server, you must ensure that the **Advertising** setting is the same on both source and destination servers to make sure that **Managed address configuration** and **Other stateful configuration** values are configured the same on both source and destination servers. For example, if **Managed address configuration** is configured automatically on the source server,

Setting type	Supported settings and notes
	then the value for the **Advertising** setting must be the same on both source and destination servers for **Managed address configuration** to be configured automatically on the destination server. **Managed address configuration** and **Other stateful configuration** settings are not imported to the destination server if they are configured as **Automatic**. In other words, they are not imported if the **Router Discovery** setting is enabled, yet the **Advertising** setting is disabled.
	IPv6 default gateway addresses (but not gateway metrics)
	IPv6 interface metric  **Note** IPv6 interface metric is supported only on Windows Server 2008, Windows Server 2008 R2 and Windows Server® 2012. If this setting is manually configured on Windows Server 2003 or Windows Server 2003 R2, it will not be migrated. For more information about how to migrate this setting manually from Windows Server 2003 or Windows Server 2003 R2, see **IP Configuration: Appendix**.
	DNS settings • IPv4 DNS server addresses • IPv6 DNS server addresses • **DNS suffix for this connection** • **Register this connection's addresses in DNS** • **Use this connection's DNS suffix in DNS registration**
For global (Windows-based) IP configuration	• For resolution of unqualified names:

Setting type	Supported settings and notes
	• **Append primary and connection-specific DNS suffixes** • **Append parent suffixes of the primary DNS suffixes** • **Append these DNS suffixes and the list of DNS suffixes** (also known as the DNS search list)
	Enable LMHOSTS lookup (but not LMHOSTS file)
	IPv6 **DisabledComponents** property **Note** This setting is supported only on Windows Server 2008, Windows Server 2008 R2 and Windows Server® 2012. If this setting is configured in Windows Server 2003 or Windows Server 2003 R2, it will not be migrated.

Scenarios and features that are not supported

Group Policy settings or other autoconfigured settings related to IP configuration are not supported.

If the source server uses additional settings for advanced IP configuration that are not in the previous list, define a custom migration procedure based on the configuration of your organization's network environment.

For more information about IP configuration settings that are not supported, see IP Configuration: Appendix.

See Also

IP Configuration: Prepare to Migrate

IP Configuration: Migrate IP Configuration Data

IP Configuration: Post-migration Tasks

IP Configuration: Appendix

IP Configuration: Prepare to Migrate

This topic helps you prepare to migrate IP configuration settings and data.

Impact on the source server

To prevent IP address conflicts when the source server has static IP addresses that you want to migrate, you must do one of the following after you export configuration from the source server, but before you import IP configuration data from the migration store to the destination server.

- Disconnect the source server from the network.
- Change static IP addresses on the source server.

Important
Changing the source server's IP address can cause roles that are running on the source server to fail.

Impact on the destination server

The destination server has only an intermittent connection to the network from the start of the migration data importation process until importation is complete. During migration, IP configuration settings that are migrated from the source server overwrite IP configuration settings on the destination server.

Important
If you migrate the static IP address from the source server to the destination server, changing the IP address can cause roles that are running on the destination server to fail.

Impact on other servers in your enterprise

Other servers in the enterprise might be affected during IP configuration migration if they depend on server roles or features that are running on the source server.

Impact on other client computers in your enterprise

Client computers cannot access either the source or destination servers during the import process. If the servers from which you are migrating IP configuration data are configured as routers, computers that are configured to use the router will not be able to connect to some networks.

Expected downtime during IP configuration migration

The following conditions can be expected during IP configuration migration.

- Users cannot access the source server while it is disconnected from the network.
- After the import operation starts, users cannot access the destination server until the import operation is fully completed.
- The destination server must be restarted for changes to the IPv6 **DisabledComponents** property to take effect.

User rights required to perform migration on both source and destination servers

Local Administrator rights are required on both the source and destination servers to perform IP configuration migration.

Preparing the destination server

Prepare the destination server for IP configuration migration by using the following steps.

To prepare the destination server

1. Install Windows Server Migration Tools on the destination server. For more information, see Install, Use, and Remove Windows Server Migration Tools.
2. Verify that all network adapters that you want to configure are enabled and connected to the network.
3. If you choose to import global IP configuration settings, verify that you can restart the server after the import operation is completed.

Preparing the source server

Perform the following steps to prepare the source server for IP configuration migration.

To prepare the source server

1. Install Windows Server Migration Tools on the destination server. For more information, see Install, Use, and Remove Windows Server Migration Tools.
2. Verify that all network adapters that have configurations that you want to migrate are enabled and connected to the network.

Important

Before you run the **Import-SmigServerSetting**, **Export-SmigServerSetting**, or **Get-SmigServerFeature** cmdlets, verify that during migration, both source and destination

servers can contact the domain controller that is associated with domain users or groups who are members of local groups on the source server.

Before you run the **Send-SmigServerData** or **Receive-SmigServerData** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with those domain users who have rights to files or shares that are being migrated.

Preparing other computers in the enterprise

Perform the following steps to prepare the source server for IP configuration migration.

To prepare other computers

- Notify users that they will be unable to access either source or destination servers after the import operation has started, and will not have access until IP configuration migration is complete.

See Also

Migrate IP Configuration to Windows Server 2012

IP Configuration: Migrate IP Configuration Data

IP Configuration: Post-migration Tasks

IP Configuration: Appendix

IP Configuration: Migrate IP Configuration Data

After you have prepared for IP configuration migration by performing steps in IP Configuration: Prepare to Migrate, migrate IP configuration settings and data by using procedures in this section.

Migrating Global and NIC IP configuration

Perform steps in this section to migrate IP configuration data.

IP configuration migration tools

Windows PowerShell cmdlets that are used for data and share migration include **Export-SmigServerSetting** (used on the source server), and **Import-SmigServerSetting** (used on the destination server).

The **Export-SmigServerSetting** cmdlet lets the user copy all supported IP configuration settings from the source server to migration store at a specified location.

On the destination server, the **Import-SmigServerSetting** cmdlet applies the IP configuration settings specified in the migration store to the destination computer. To import the IP configuration settings of network adapters, you must provide the mapping between the source and destination network adapters by listing the physical addresses (also called MAC addresses) of all network adapters.

Note
To import and export IP configuration settings for a network adapter, it must be enabled and connected to a network.

For more information, see Help for Windows PowerShell cmdlets. To view Help for a cmdlet, in a Windows PowerShell session, type **Get-Help** <*cmdlet_name*>, and then press **Enter**.

Note
Windows Server Migration Tools must be installed on the computer on which you want to view Help for the migration cmdlets.

Migrating IP configuration by using Windows Server Migration Tools

Use the two procedures in this section to migrate global and network adapter-specific configuration settings by using Windows Server Migration Tools.

Export IP configuration settings from the source server

If you have already exported IP configuration settings from your source server as part of another migration guide, go to the procedure Import IP configuration settings to the destination server.

To export IP configuration settings from the source server

1. Do one of the following.
   - To open a Windows Server Migration Tools custom Windows PowerShell session on computers that are running Windows Server® 2012 go to **Start** and then click **Windows Server Migration Tools**.
   - To open a Windows Server Migration Tools custom Windows PowerShell session on computers that are running Windows Server 2008 R2 or Windows Server 2008 click **Start**, point to **Administrative Tools**, open the **Windows Server Migration Tools** folder, right-click **Windows Server Migration Tools**, and then click **Run as administrator**.
   - To open a Windows Server Migration Tools custom Windows PowerShell session on computers that are running Windows Server® 2012 go to **Start** and then click **Windows Server Migration Tools**.
   - To open a Windows Server Migration Tools custom Windows PowerShell session on computers that are running Windows Server 2003, click **Start**, point to **Administrative Tools**, open the **Windows Server Migration Tools** folder, and then click **Windows Server Migration Tools**.

2. In the same Windows PowerShell session, run the **Export-SmigServerSetting** cmdlet on the source server by typing the following command, in which *MigrationStorePath* represents the path of your migration store location, and then pressing **Enter**.

```
Export-SmigServerSetting -IPConfig -Path <MigrationStorePath>
-Verbose
```

Notes

Because network connectivity might be interrupted during the import operation, be sure to verify that the migration store is created on the destination computer.

You are prompted to provide a password to encrypt the migration store. Remember this password, because you must provide the same password to import settings from the migration store.

3. Because it contains information that you must have to perform the import operation, save the output of the **Ipconfig -all** command. Type the following, and then press **Enter**, in which *FileName* represents the path of the location in which you want to save the output text file, and the file name.

```
IPConfig -all > <FileName>
```

4. If the source server has a static IP address, disconnect the source server, or change the static IP address.

Import IP configuration settings to the destination server

To import IP configuration settings to the destination server

1. For network adapter IP configuration migration, map physical (MAC) addresses for both source and destination network adapters. View the IPConfig output you generated by using the **IPConfig -all > <FilePath>** command in To export IP configuration settings from the source server to determine network adapter physical address mapping.
2. If the migration store is not already on the destination server, copy the migration store to a local drive on the destination server by typing the following and then pressing **Enter**, in which *NetworkPath* is the path of the location of the migration store, and *LocalPath* is the path of a location on the destination server.

```
Copy <NetworkPath> <LocalPath>
```

3. Log on to the destination server as a member of the Administrators group, if you have not already done so.
4. On the destination server, migrate all IP configuration by using the **Import-SmigServerSetting** cmdlet as shown in the following example, in which each *SourcePhysicalAddress* and *TargetPhysicalAddress* in quotation marks represents the physical address of a network adapter that you want to migrate, and *MigrationStorePath* represents the path of the location of your migration store. Specify each network adapter physical address in the format *AA-AA-AA-AA-AA-AA*, and separate the physical addresses of multiple network adapters by using commas.

```
Import-SmigServerSetting -IPConfig All -SourcePhysicalAddress
```

```
"<SourcePhysicalAddress1>","<SourcePhysicalAddress2>" -
TargetPhysicalAddress
"<TargetPhysicalAddress1>","<TargetPhysicalAddress2>" -Path
<MigrationStorePath> -Verbose
```

You can use one of the following values with the **-IPConfig** parameter. For **All** or **Global** IP configuration migration, the destination server must be restarted for modifications to the IPv6 **DisabledComponents** property to take effect. You cannot use any of the Windows Server Migration Tools cmdlets until the server has restarted.

- **Global:** only import global Windows IP configuration settings.
- **NIC:** only import specific IP configuration settings for certain network adapters. You must specify the physical address mapping by using the **-SourcePhysicalAddress** and **-TargetPhysicalAddress** parameters.
- **All:** import both global and network adapter IP configuration-specific settings. You must also specify the physical address mapping by using the **-SourcePhysicalAddress** and **-TargetPhysicalAddress** parameters.

For the list of supported settings for network adapters, see IP Configuration: Prepare to Migrate.

5. You are prompted to provide the same password that was provided during the export process to decrypt the migration store. Type the password, and then press **Enter**.

See Also

Migrate IP Configuration to Windows Server 2012

IP Configuration: Prepare to Migrate

IP Configuration: Post-migration Tasks

IP Configuration: Appendix

IP Configuration: Post-migration Tasks

After you have migrated IP configuration settings and data as directed in IP Configuration: Migrate IP Configuration Data, verify the migration and, if necessary, roll back IP configuration migration by using the procedure in this section.

Verifying the migration

Perform the following steps to verify your IP configuration migration.

To verify the IP configuration migration

1. Open a Command Prompt window on the destination server. To do this, click **Start**, click **Run**, type **cmd**, and then press **Enter**.

2. Verify that all IP configurations that you wanted to migrate exist on the correct network adapters on the destination server. To do this, type the following, , and then press **Enter**.

   ```
   IPConfig -all
   ```

3. Compare the results of the **IPConfig -all** command with the IPConfig output you generated on the source server in the procedure “To export IP configuration settings from the source server” in IP Configuration: Migrate IP Configuration Data.
4. For static IP address migration, verify that you can access the destination server by using the same IP address as the source server had before the migration. You can verify this by using the **ping** command in a Windows Command Prompt session.

Rolling back migration

If necessary, perform the following steps to roll back IP configuration migration.

To roll back migration

1. If you obtained a different static IP address for your source server, and migrated the statically-configured IP address to the destination server, either disconnect the destination server from the network or obtain a new static IP address for the destination server.
2. Set the IP address of the source server back to the pre-migration static IP address.
3. Connect the source server back to the network if you disconnected it in step 1.

Troubleshooting cmdlet-based migration

The Windows Server Migration Tools deployment log file is located at %*windir*%\Logs\SmigDeploy.log. Additional Windows Server Migration Tools log files are created at the following locations.

- %*windir*%\Logs\ServerMigration.log
- On Windows Server® 2012, Windows Server 2008 and Windows Server 2008 R2: %*localappdata*%\SvrMig\Log
- On Windows Server 2003: %*userprofile*%\Local Settings\Application Data\SvrMig\Log

If migration log files cannot be created in the preceding locations, **ServerMigration.log** and **SmigDeploy.log** are created in %*temp*%, and other logs are created in %*windir*%\System32.

If a migration cmdlet fails, and the Windows PowerShell session closes unexpectedly with an access violation error message, look for a message similar to the following example in the %*localappdata*%\SvrMig\Logs\setuperr.log file.

```
FatalError [0x090001] PANTHR Exception (code 0xC0000005: ACCESS_VIOLATION) occurred at
0x000007FEEDE9E050 in C:\Windows\system32\migwiz\unbcl.dll (+000000000008E050).  Minidump
attached (317793 bytes).
```

This failure occurs when the server cannot contact domain controllers that are associated with domain users or groups who are members of local groups, or who have rights to files or shares

that are being migrated. When this happens, each domain user or group is displayed in the GUI as an unresolved security identifier (SID). An example of a SID is **S-1-5-21-1579938362-1064596589-3161144252-1006**.

To prevent this problem, verify that required domain controllers or global catalog servers are running, and that network connectivity allows communication between both source and destination servers and required domain controllers or global catalog servers. Then, run the cmdlets again.

If connections between either the source or destination servers and the domain controllers or global catalog servers cannot be restored, do the following.

1. Before you run **Export-SmigServerSetting**, **Import-SmigServerSetting** or **Get-SmigServerFeature** again, remove all unresolved domain users or groups who are members of local groups from the server on which you are running the cmdlet.
2. Before you run **Send-SmigServerData** or **Receive-SmigServerData** again, remove all unresolved domain users or groups who have user rights to files, folders, or shares on the migration source server.

Viewing the content of Windows Server Migration Tools result objects

All Windows Server Migration Tools cmdlets return results as objects. You can save result objects and query them for more information about settings and data that were migrated. You can also use result objects as input for other Windows PowerShell commands and scripts.

Result object descriptions

The Windows Server Migration Tools **Import-SmigServerSetting** and **Export-SmigServerSetting** cmdlets return results in a list of **MigrationResult** objects. Each **MigrationResult** object contains information about the data or setting that the cmdlet processes, the result of the operation, and any related error or warning messages. The following table describes the properties of a **MigrationResult** object.

Property name	Type	Definition
ItemType	Enum	The type of item being migrated. Values include **General**, **WindowsFeatureInstallation**, **WindowsFeature**, and **OSSetting**.
ID	String	The ID of the migrated item. Examples of values include **Local User**, **Local Group**, and **DHCP**.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise,

Property name	Type	Definition
		False is displayed.
DetailsList	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.

Send-SmigServerData and **Receive-SmigServerData** cmdlets return results in a list of **MigrationDataResult** objects. Each **MigrationDataResult** object contains information about the data or share that the cmdlet processes, the result of the operation, any error or warning messages, and other related information. The following table describes the properties of a **MigrationDataResult** object.

Property name	Type	Definition
ItemType	Enum	The type of migrated item. Values include **File**, **Folder**, **Share**, and **Encrypted File**.
SourceLocation	String	The source location of the item, shown as a path name.
DestinationLocation	String	The destination location of the item, shown as a path name.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise, **False** is displayed.
Size	Integer	The item size, in bytes.
ErrorDetails	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.
Error	Enum	Errors enumeration for errors that occurred.
WarningMessageList	List <String>	A list of warning messages.

The following table describes the properties of objects within the **MigrationResultDetails** object that are common to both **MigrationResult** and **MigrationDataResult** objects.

Property name	Type	Definition
FeatureId	String	The name of the migration setting that is related to the item. Examples of values

Property name	Type	Definition
		include **IPConfig** and **DNS**. This property is empty for data migration.
Messages	List <String>	A list of detailed event messages.
DetailCode	Integer	The error or warning code associated with each event message.
Severity	Enum	The severity of an event, if events occurred. Examples of values include **Information**, **Error**, and **Warning**.
Title	String	Title of the result object. Examples of values include NIC physical address for IP configuration, or user name for local user migration.

Examples

The following examples show how to store the list of the result objects in a variable, and after migration is complete, use the variable in a query to return the content of result objects.

To store a list of result objects as a variable for queries

1. To run a cmdlet and save the result in variable, type a command in the following format, and then press **Enter**.

   ```
   $VariableName = $(Cmdlet)
   ```

   The following is an example.

   ```
   $ImportResult = $(Import-SmigServerSetting -FeatureId DHCP -
   User all -Group -Path D:\rmt\DemoStore -force -Verbose)
   ```

   This command runs the **Import-SmigServerSetting** cmdlet with several parameters specified, and then saves result objects in the variable **ImportResult**.

2. After the **Import-SmigServerSetting** cmdlet has completed its operations, return the information contained in the result object by typing a command in the following format, and then pressing **Enter**.

   ```
   $VariableName
   ```

   In the following example, the variable is named **ImportResult**.

```
$ImportResult
```

This command returns information contained in the result objects that were returned by **Import-SmigServerSetting** in the example shown in step 1. The following is an example of the output that is displayed by calling the **ImportResult** variable.

```
            ItemType  ID                                Success
DetailsList
            --------  --                                -------
-----------
          OSSetting  Local User                          True
{Local User, Loc...
          OSSetting  Local Group                         True
{Local Group, Lo...
     WindowsFeature  DHCP                                True
{}
```

Each line of the preceding sample is a migration result for an item that was migrated by using the **Import-SmigServerSetting** cmdlet. The column heading names are properties of **MigrationResult** objects. You can incorporate these properties into another command to return greater detail about result objects, as shown by examples in step 3 and forward.

3. To display a specific property for all result objects in the list, type a command in the following format, and then press **Enter**.

   ```
   $<VariableName>| Select-Object -ExpandProperty <PropertyName>
   ```

   The following is an example.

   ```
   $importResult | Select-Object -ExpandProperty DetailsList
   ```

4. You can run more advanced queries to analyze result objects by using Windows PowerShell cmdlets. The following are examples.

   - The following command returns only those details of result objects that have the ID **Local User**.

     ```
     $ImportResult | Where-Object { $_.ID -eq "Local User" } |
     Select-Object -ExpandProperty DetailsList
     ```

   - The following command returns only those details of result objects with an ID of **Local User** that have a message severity equal to **Warning**.

     ```
     $ImportResult | Where-Object { $_.ID -eq "Local User" } |
     Select-Object -ExpandProperty DetailsList | ForEach-Object {
     if ($_.Severity -eq "Warning") {$_} }
     ```

   - The following command returns only the details of result objects with an ID of **Local Group** that also have the title **Remote Desktop Users**.

     ```
     $ImportResult | Where-Object { $_.ID -eq "Local Group" } |
     ```

```
Select-Object -ExpandProperty DetailsList | ForEach-Object {
if ($_.Title -eq "Remote DesktopUsers") {$_} }
```

See Also

Migrate IP Configuration to Windows Server 2012

IP Configuration: Prepare to Migrate

IP Configuration: Migrate IP Configuration Data

IP Configuration: Appendix

IP Configuration: Appendix

This Appendix contains additional information that might help you prepare for a custom IP configuration migration when your migration is not supported according to the list of supported migration scenarios and features in IP Configuration: Prepare to Migrate.

Migrating manually-configured IPv6 interface metrics from Windows Server 2003

If you have manually-configured IPV6 interface metrics on a computer that is running Windows Server 2003, manually migrate the interface metrics to Windows Server® 2012 by using **netsh** commands.

To obtain index numbers for source and destination network adapters

1. On source servers that are running either Windows Server 2003 or Windows Server 2003 R2, open a Command Prompt session by clicking **Start**, clicking **Run**, typing **cmd** in the **Open** box, and then either clicking **OK** or pressing **Enter**.
2. On the destination server, open a Command Prompt session with elevated user rights. To do this, click **Start**, click **All Programs**, click **Accessories**, right-click **Command Prompt**, and then click **Run as administrator**.
3. On both source and destination servers, obtain the index numbers of source and destination network adapters. Type the following in each Command Prompt session, and then press **Enter**.

   ```
   netsh interface ipv6 show interface
   ```
4. Record the numbers in the **Index** column that correspond to the names of the interfaces that you want to migrate.

To obtain the manually-configured IPv6 metric from the source server

1. If a Command Prompt window is not already open on the source server, open one as directed in Sstep 1 of To obtain index numbers for source and destination network

adapters.

2. Type the following, in which *Index* represents the index numbers that you obtained in To obtain index numbers for source and destination network adapters, and then press **Enter**.

   ```
   netsh interface ipv6 show interface <Index>
   ```

   For example, if your interface has an index number of 11, use the following command.

   ```
   netsh interface ipv6 show interface 11
   ```

3. Record the IPv6 metric that you want to migrate to the destination server in the **Metric** field.

To migrate the manually-configured IPv6 metric to the destination server

- If a Command Prompt window is not already open on the destination server, open one as directed in step 1 of To obtain index numbers for source and destination network adapters.
- Type the following, in which *Index* represents the number that you obtained in To obtain index numbers for source and destination network adapters, and *Integer* represents the number that you obtained in To obtain the manually-configured IPv6 metric from the source server, and then press **Enter**.

  ```
  netsh interface ipv6 set interface <Index> metric=<Integer>
  ```

  For example, for an interface with an index of 22, use the following command to set the metric to **2**.

  ```
  netsh interface ipv6 set interface 22 metric=2
  ```

Additional resources

Some advanced IPv4 and IPv6 configuration settings for a network adapter are not displayed in the Windows interface. Depending upon the configuration of the destination network, you might need to migrate these settings manually to a destination server.

- If you have manually configured nondefault routes for an interface, use the following command to view these settings on the source server. The value for *ipvx* can be either IP (for IPv4) or IPv6.

  ```
  netsh interface <ipvx> show route
  ```

  To set these settings on the destination server, see **netsh** help by entering the following command, in which the value for *ipvx* can be either IPv4 or IPv6.

  ```
  netsh interface <ipvx> add route
  ```

- If you have manually configured general interface settings for a network adapter, use the following **netsh** command to view these settings on the source server. The value for *ipvx* can be either IP (for Ipv4) or IPv6.

  ```
  netsh interface <ipvx> show interface <InterfaceIndex>
  ```

  To set these configuration settings on the destination server, view **netsh** help by entering the following command, in which the value for *ipvx* can be either IPv4 or IPv6.

```
netsh interface <ipvx> set interface
```

For the complete list of all settings that can be viewed and configured by using the **netsh** command, see the following articles on the Microsoft Web site.

- Netsh commands for Interface IP (http://technet.microsoft.com/en-us/library/cc738592.aspx)
- Netsh commands for Interface IPv6 (http://technet.microsoft.com/en-us/library/cc740203.aspx)

Additional general TCP/IP configuration parameters are stored in registry keys. For the complete list of general TCP/IP configuration settings that are stored in registry keys, see TCP/IP Configuration Parameters on the Microsoft Web site (http://technet.microsoft.com/en-us/library/cc739819.aspx).

For additional information about IP configuration, the following resources are recommended.

- TCP/IP Fundamentals for Microsoft Windows (http://www.microsoft.com/downloads/details.aspx?FamilyID=c76296fd-61c9-4079-a0bb-582bca4a846f&displaylang=en)
- Understanding IPv6, Second Edition (http://www.microsoft.com/MSPress/books/11607.aspx)

See Also

Migrate IP Configuration to Windows Server 2012

IP Configuration: Prepare to Migrate

IP Configuration: Migrate IP Configuration Data

IP Configuration: Post-migration Tasks

Migrate Network Policy Server to Windows Server 2012

This document provides guidance for migrating the Network Policy Server (NPS) or Internet Authentication Server (IAS) role service from an x86-based or x64-based server running Windows Server 2003, Windows Server® 2008, Windows Server® 2008 R2, or Windows Server® 2012 to a new Windows Server® 2012 server.

About this guide

Note

Your detailed feedback is very important, and helps us to make Windows Server Migration Guides as reliable, complete, and easy to use as possible. Please take a moment to rate this topic by clicking the stars in the upper-right corner of the page (1=poor, 5=excellent), and then add comments that support your rating. Describe what you liked, did not like, or want to see in future versions of the topic. To submit additional

suggestions about how to improve Migration guides or utilities, post on the Windows Server Migration forum.

NPS migration documentation and tools ease the migration of NPS role service settings and data from an existing server to a destination server that is running Windows Server 2012. By using the tools that are described in this guide, you can simplify the IAS/NPS migration process, reduce migration time, increase the accuracy of the IAS/NPS migration process, and help to eliminate possible conflicts that might otherwise occur during the migration process.

Target audience

This guide is intended for the following IT professionals:

- IT architects responsible for computer management and security throughout an organization.
- IT operations engineers who are responsible for the day-to-day management and troubleshooting of networks, servers, client computers, operating systems, or applications.
- IT operations managers who are accountable for network and server management.

What this guide does not provide

This guide does not provide detailed steps to migrate the configuration of other services that might be running on the source server.

Guidance is not provided for scenarios in which the new operating system is installed on existing server hardware by using the upgrade option during setup.

Supported migration scenarios

This guide provides the instructions for migrating an existing server that is running NPS or IAS to a server that is running Windows Server 2012. This guide does not contain instructions for Network Policy Server migration when the source server is running multiple roles. If your server is running multiple roles, it is recommended that you design a custom migration procedure specific to your server environment, based on the information provided in other role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server 2012.

Caution

If your source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other roles that are running on the source server to fail.

Supported operating systems

The following table displays the minimum operating system requirements that are supported by this guide.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 SP2	Windows Server 2012	x64-based
x86- or x64-based	Windows Server 2003 R2	Windows Server 2012	x64-based
x86- or x64-based	Windows Server® 2008	Windows Server 2012	x64-based
x64-based	Windows Server 2008 R2	Windows Server 2012	x64-based
x64-based	Windows Server 2012	Windows Server 2012	x64-based

- The NPS role service is not available in Server Core editions. Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported as either source or destination servers. Windows Server Foundation edition is not available for Windows Server 2003.
- Migration from a source server to a destination server that is running an operating system with a different installed language is not supported. For example, migration of server roles from a computer that is running Windows Server 2008 with a system language of French to a computer that is running Windows Server 2012 with a system language of German is not supported. The system language is the language of the localized installation package that was used to set up the Windows operating system.
- Both x86-based and x64-based migrations are supported for Windows Server 2003 and Windows Server 2008. All editions of Windows Server 2012 are x64-based.

Supported NPS role configurations

Migration of the following NPS settings are supported by this guide:

1. **Policies**. Migration of NPS policy configuration, including connection request policies, network policies, and health policies is supported by using this guide.
2. **Authentication methods**. All supported authentication method settings can be migrated using this guide. For more information about authentication methods, see NPS Authentication Methods (http://go.microsoft.com/fwlink/?LinkId=169629).
3. **System Health Validators (SHVs)**. Migration of SHV configuration settings implemented using Microsoft published SDK are supported.
4. **NPS templates**. Template settings are migrated using NPS UI export and import functionality. You cannot migrate template settings using the command line.
5. **RADIUS clients and remote RADIUS servers**. RADIUS clients and remote RADIUS server configuration settings, including shared secrets can be migrated using this guide.

6. **SQL accounting**. The configuration of SQL parameters, including connection, description, accounting, authentication, periodic accounting status, periodic authentication status, and max sessions settings can be migrated using this guide. It is recommended to manually configure SQL connection string settings. For more information, see Configure SQL Server Logging in NPS (http://go.microsoft.com/fwlink/?LinkId=169631).

IP address and host name configuration

This guide supports the following scenarios:

1. The destination server is configured with the same host name or IP address as source server.
2. The destination server is configured with a different host name or IP address than the source server.

Migration scenarios that are not supported

The following migration scenarios are not covered in this document:

- **Upgrade**. Guidance is not provided for scenarios in which the new operating system is installed on existing server hardware by using the **Upgrade** option during setup.
- **Extension DLLs**. This guide does not support migration of extension DLL registry key settings. For more information about extension DLL registry key migration, see Setting Up the Extension DLLs (http://go.microsoft.com/fwlink/?LinkId=169632).
- **Non-Microsoft authentication methods**. The migration of settings for non-Microsoft authentication methods is not supported. To migrate these settings, refer to your vendor documentation.
- **Non-Microsoft SHVs**. The migration of settings for non-Microsoft SHVs is supported only if the SHV is developed using guidance from the NAP SHA/SHV SDK. To migrate these settings, refer to your vendor documentation.

Overview of migration process for this role

Network Policy Server (NPS) is the Microsoft implementation of a Remote Authentication Dial-in User Service (RADIUS) server and proxy in Windows Server 2012. NPS is the replacement for Internet Authentication Service (IAS) in Windows Server 2003.

The current topic provides an overview of the NPS migration process. The NPS migration guide also includes the following major sections:

- NPS Server Migration: Preparing to Migrate
- NPS Server Migration: Migrating the NPS Server
- NPS Server Migration: Verifying the Migration
- NPS Server Migration: Post-migration Tasks

The pre-migration process involves establishing a storage location for migration data, collection of information that will be used to perform the server migration, and operating system installation on the destination server. The NPS migration process includes using the **iasmigreader** tool if the source server is running Windows Server 2003. If the source server is running Windows

Server 2008 or Windows Server 2008 R2, the Network Shell (netsh) utility is used to obtain NPS settings. When migrating a source server running Windows Server 2012, you can use netsh or Windows PowerShell®. Procedures are then performed on the destination server to install the required roles and migrate NPS settings. Verification procedures include testing the destination server to ensure it works correctly. Post-migration procedures include retiring or repurposing the source server.

Process diagram

The following diagram provides an overview of the migration process.

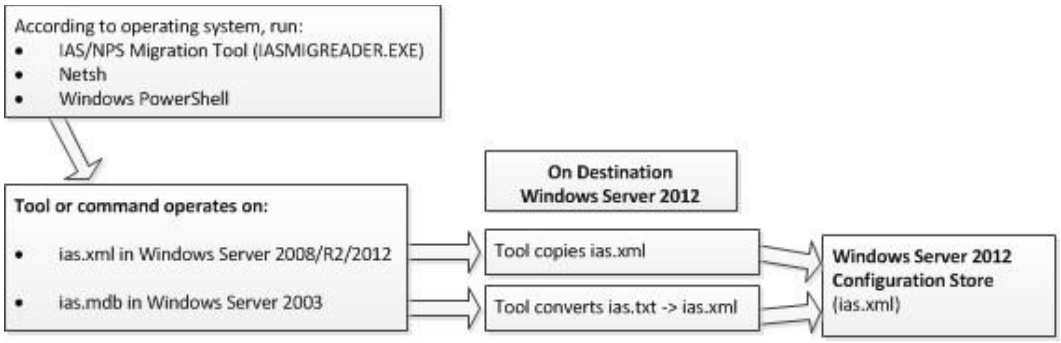


Figure 1. NPS server migration overview

Impact of migration

In its recommended configuration, the destination server has the same host name and IP address as the source server. In this scenario, the source server will be unavailable to process network access requests for the duration of the migration process (estimated 1-2 hours).

This guide also includes procedures for migration of the NPS server configuration from the source server to a destination server with a different host name or IP address. This allows the source and destination NPS servers to run simultaneously until all testing and verification is complete, and reduces service disruption. If you change the name or IP address of the server running NPS, RADIUS clients must also be updated with the new NPS server name and IP address.

Impact of migration on the source server

- When deploying the destination server with the same host name and IP address as the source server, the source server must be decommissioned and taken offline prior to renaming the destination server from *tempNPS* to the host name of the source server.
- When deploying the destination server with a different host name and IP address, there is no impact to the source server.

Impact of migration on other computers in the enterprise

- When deploying the destination server with the same host name and IP address, network access requests cannot be evaluated by NPS while the source server is offline and before the destination server brought online with the same name and IP address. During this time, client computers requesting access to the network cannot authenticate and are denied network access.
- When deploying the destination server with a different host name and IP address, RADIUS client settings for all network access servers that are configured to use the source server must be updated.

Permissions required to complete migration

The following permissions are required on the source server and the destination server:

- Membership in the **Administrators** group, or the equivalent, is the minimum required to install and configure server running NPS.
- Membership in the SQL database rights are required for SQL settings migration.
- If the destination server is a domain member, membership in the **Domain Admins** group, or the equivalent, is the minimum required to authorize the NPS server.

Estimated duration

The work required to migrate NPS settings from the source to destination server, including testing, can require 1 to 2 hours. Additional time may be required for migration of non-Microsoft authentication methods, SHVs or extension DLLs.

See Also

NPS Server Migration: Preparing to Migrate

NPS Server Migration: Migrating the NPS Server

NPS Server Migration: Verifying the Migration

NPS Server Migration: Post-migration Tasks

NPS Server Migration: Appendix A - Data Collection Worksheet

NPS Server Migration: Preparing to Migrate

Migration of Network Policy Server (NPS) includes the following tasks:

- Choose a migration file storage location
- Prepare your source server
- Prepare your destination server

Complete the steps or procedures in these sections to prepare your environment for migration.

If the server running NPS will be joined to a domain, membership in the **Domain Admins** group, or equivalent, is the minimum required to complete this procedure. If the server running NPS is not domain joined, membership in the **Administrators** group, or equivalent, is required. Review details about using the appropriate accounts and group memberships at Local and Domain Default Groups (http://go.microsoft.com/fwlink/?LinkId=83477).

Choose a migration file storage location

First, choose a location where migration files will be kept.

To choose a storage location

1. Select a file storage location where migration files will be kept. The storage location can be a network share that is accessible by both the source and destination server, or portable media that can be transferred from one server to another.

Prepare your source server

Follow these steps to prepare an x86-based or x64-based server running Windows Server 2003, Windows Server® 2008, Windows Server® 2008 R2 or Windows Server® 2012 for NPS migration.

To prepare the source server

1. Determine the domain, server name, IP address, and passwords on the source server.
2. If the source server is domain joined, determine the group membership of the source server in Active Directory Domain Services (AD DS), including security group and OU membership. This can be done using the Active Directory Users and Computers console (dsa.msc) or Server Manager on a domain controller.

Prepare your destination server

Follow these steps to prepare an x64-based destination server running Windows Server 2012 for NPS migration.

Scenario 1: Prepare the destination server using the same host name and IP address

1. Install Windows Server 2012 on the destination server.
2. If the source server host name is used by RADIUS clients or remote RADIUS server groups, name the destination server with a temporary server name, for example: *TempNPS*.
3. If the source server IP address is used by RADIUS clients or remote RADIUS server groups, assign a different temporary static IP address to the destination server.
4. If the source server is domain joined, add the destination server to the domain of the source server. Configure AD DS group membership settings on the destination server

that are identical to the source server, including security group and OU membership.

5. Install the NPS role service using the steps provided in Install Network Policy Server (NPS) (http://go.microsoft.com/fwlink/?LinkId=169633).
6. If the source server has non-Microsoft authentication methods installed, then install same authentication methods on the destination server using your vendor documentation before importing the source server configuration.
7. If the source server has extension DLLs installed, install the same extension DLLs on the destination server before importing the source server configuration. For more information, see Setting Up the Extension DLLs (http://go.microsoft.com/fwlink/?LinkId=169632).
8. If the source server has non-Microsoft SHVs installed, then install same SHVs on the destination server using your vendor documentation before importing the source server configuration.

Scenario 2: Prepare the destination server using a different host name and IP address

1. Follow the same steps as provided for scenario 1, replacing the temporary server name with the new destination server host name, and assigning a permanent static IP address.

The destination server is now prepared for migration.

See Also

Migrate Network Policy Server to Windows Server 2012

NPS Server Migration: Migrating the NPS Server

NPS Server Migration: Verifying the Migration

NPS Server Migration: Post-migration Tasks

NPS Server Migration: Appendix A - Data Collection Worksheet

NPS Server Migration: Migrating the NPS Server

This topic contains steps and procedures for migrating the Network Policy Server (NPS) role service from a legacy source server to a new x64-based destination server running Windows Server 2012.

This topic includes sample Windows PowerShell cmdlets that you can use to automate some of the procedures described. For more information, see Using Cmdlets.

Known issues

If you previously created conditional attributes for your remote access policy using **Called Station ID** and **Calling Station ID**, the comparison of these attributes in Windows Server 2012 now uses

a regular expression instead of matching the exact string. For a description of these attributes, see Remote Access Policy Conditions in the **IAS Authorization** section.

Exporting settings from the source server

Use the following procedures to export the NPS settings from your x86-based or x64-based source server prior to migrating to an x64-based server running Windows Server 2012. Follow the steps in the appropriate section based on the version of Windows Server that is running on the source server:

- Exporting settings from Windows Server 2003
- Exporting settings from Windows Server 2008
- Exporting settings from Windows Server 2008 R2
- Exporting settings from Windows Server 2012

Warning

When you use the following procedures to export configuration settings, apply appropriate precautions when moving these files from the source server to destination servers. NPS server configurations are not encrypted in the exported XML file, and contain shared secrets for RADIUS clients and members of remote RADIUS server groups. Therefore, sending these files over a network connection might pose a security risk. You can add the file to an encrypted, password protected archive file before moving the file to provide greater security. In addition, store the file in a secure location to prevent access by unauthorized users.

Exporting settings from Windows Server 2003

Configuration settings for Internet Authentication Service (IAS) in Windows Server 2003 are stored in **.MDB** files. Configuration settings for Network Policy Server (NPS) in Windows Server 2012 are stored in **.XML** files. **Iasmigreader.exe** is a command-line tool that exports the configuration settings of IAS on a computer running Windows Server 2003 to a text file. You can obtain the **iasmigreader.exe** command line migration tool for migrating Windows Server 2003 IAS settings to Windows Server 2012 from the following locations:

1. Windows Server 2012 installation media provides a copy of the migration tool in the **\sources\dlmanifests\microsoft-windows-iasserver-migplugin** directory.
2. The migration tool is available in the **%windir%\syswow64** directory on a server running Windows Server 2012.

To export settings from a source server running Windows Server 2003

1. Copy **iasmigreader.exe** to the source server into a directory configured in the **%path%** environment variable.

   Tip

   To review the source server's **%path%** configuration, type **echo %path%** at a

command prompt and press Enter.

2. At an elevated command prompt, type **iasmigreader.exe**, and then press Enter. The migration tool will automatically export settings to a text file.

   Important
   Configuration changes made to IAS will take at least one minute to be available for export.

3. IAS settings are stored in the file **ias.txt** located in the **%windir%\system32\ias** directory on the source server. If you are running a 64-bit version of Windows Server 2003, the **ias.txt** file is located in the **%windir%\syswow64\ias** directory.
4. You must manually copy SQL log configuration settings on the source server to a file (example: sql.txt).

   To record these settings:

   a. At an elevated command prompt, type **ias.msc**, and then press Enter.
   b. In the IAS console tree, click **Remote Access Logging**, right-click **SQL Server**, and then click **Properties**.
   c. Record the configuration settings on the **Settings** tab, and then click **Configure**.
   d. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.

5. Copy the ias.txt and sql.txt files to the migration store file location.

Warning

Store the ias.txt and sql.txt files in a secure location. These files contain shared secret information and SQL connection strings.

Important

When you migrate the configuration settings of the IAS role service that is running on a 32-bit or a 64-bit Windows Server 2003–based source server to the NPS role service that is running on a Windows Server 2012–based destination server, the import procedure seems to complete successfully. However, the Extensible Authentication Protocol (EAP) method is misconfigured. This occurs because the migration tool generates a faulty parameter that is stored in the configuration text file (ias.txt). For more information about this issue and for a workaround, see The EAP method is configured incorrectly during the migration process from Windows Server 2003 32-bit or a 64-bit to Windows Server 2008 R2 (http://go.microsoft.com/fwlink/?LinkID=181982).

Exporting settings from Windows Server 2008

Configuration settings for NPS in Windows Server 2008 are stored in **.XML** files that can be directly imported to the destination server. The Network Shell (NetSh) command line utility can be

used to export and import these settings. You can also use the Windows interface to import and export these settings.

Warning

You cannot use the Windows interface or a command line to export or import detailed SQL configuration settings. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.

To export settings from a source server running Windows Server 2008 using a command line

1. On the source NPS server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps export filename="path\file.xml" exportPSK=YES
   ```

   Replace *path* with the directory location where you want to save the source server configuration file, and replace *file* with the name of the .XML file that you want to save.
2. Confirm that a message appears indicating that the export to file was successful.
3. On the source server, type the following command and then press Enter:

   ```
   netsh nps show sqllog > path\sql.txt
   ```

   Replace *path* with the directory location where you want to save the source server SQL configuration file, and replace *sql* with the name of the .TXT file that you want to save. This file contains the basic configuration for SQL logging that is found on the **Settings** tab in SQL logging properties. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.
4. Copy the **file.xml** and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server running Windows Server 2008 using the Windows interface

1. On the source server, open Server Manager.
2. In the Server Manager console tree, open **Roles\Network Policy and Access Services\NPS**.
3. Right click **NPS**, and then click **Export Configuration**.
4. In the dialog box that appears, select the check box next to **I am aware that I am exporting all shared secrets**, and then click **OK**.
5. Next to **File name**, type **file.xml**, navigate to the migration store file location, and then click **Save**.
6. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

   To record these settings:

a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.

b. Record the configuration settings on the **Settings** tab, and then click **Configure**.

c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.

7. Copy the ias.txt and sql.txt files to the migration store file location.

Exporting settings from Windows Server 2008 R2

Configuration settings for NPS in Windows Server 2008 R2 are stored in **.XML** files that can be directly imported to the destination server. The Network Shell (NetSh) command line utility can be used to export and import these settings. You can also use the Windows interface to import and export settings.

Warning

You cannot use the Windows interface or a command line to export or import detailed SQL configuration settings. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.

Important

The netsh utility does not support migration of template configuration settings. To migrate these settings, you must use the Windows interface.

To export settings from a source server running Windows Server 2008 R2 using a command line

1. On the source NPS server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps export filename="path\file.xml" exportPSK=YES
   ```

   Replace *path* with the directory location where you want to save the source server configuration file, and replace *file* with the name of the .XML file that you want to save.

2. Confirm that a message appears indicating that the export to file was successful.

3. On the source server, type the following command and then press Enter:

   ```
   netsh nps show sqllog > path\sql.txt
   ```

   Replace *path* with the directory location where you want to save the source server SQL configuration file, and replace *sql* with the name of the .TXT file that you want to save. This file contains the basic configuration for SQL logging that is found on the **Settings** tab in SQL logging properties. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data

Collection Worksheet.

4. Copy the **file.xml** and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server running Windows Server 2008 R2 using the Windows interface

1. On the source server, open Server Manager.
2. In the Server Manager console tree, open **Roles\Network Policy and Access Services\NPS**.
3. Right click **NPS**, and then click **Export Configuration**.
4. In the dialog box that appears, select the check box next to **I am aware that I am exporting all shared secrets**, and then click **OK**.
5. Next to **File name**, type **file.xml**, navigate to the migration store file location, and then click **Save**.
6. In the console tree, right-click **Templates Management** and then click **Export Templates to a file**.
7. Next to **File name**, type **iastemplates.xml**, navigate to the migration store file location, and then click **Save**.
8. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

   To record these settings:

   a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.

   b. Record the configuration settings on the **Settings** tab, and then click **Configure**.

   c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.
9. Copy the **file.xml**, **iastemplates.xml**, and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

Exporting settings from Windows Server 2012

Configuration settings for NPS in Windows Server 2012 are stored in **.XML** files that can be directly imported to the destination server. You can use the following methods to export and import these settings:

1. The Network Shell (NetSh) command line utility
2. The Windows interface
3. Windows PowerShell cmdlets

Warning

You cannot use Windows PowerShell, the Windows interface or a command line to export or import detailed SQL configuration settings. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.

Important
The netsh utility and Windows PowerShell do not support migration of template configuration settings. To migrate these settings, you must use the Windows interface.

To export settings from a source server running Windows Server 2012 using Windows PowerShell

1. On the source server, create a new folder for your settings (for example: C:\ConfigSettings).
2. Export your configuration settings to an .xml file in that folder, by following these steps.
   a. On the **Start** screen, type **PowerShell**, and then click Enter.
   b. To switch to the NPS context enter the following Windows PowerShell command and then press Enter:

      Import-Module NPS

   c. To export the configuration file to an .xml file, enter the following Windows PowerShell command, using the -path parameter to identify the name of the .xml file to be created and the folder into which it should be placed:

      Export-NpsConfiguration [-Path] <String>

      Tip
      For example:

      Export-NpsConfiguration –Path C:\ConfigSettings -Path nps01.xml

      Caution
      The exported file contains unencrypted shared secrets for RADIUS clients and members of remote RADIUS server groups. Because of this, you should ensure that the file is stored in a secure location to prevent malicious users from accessing the file.
3. Confirm that no errors were reported by Windows PowerShell.
4. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

   To record these settings:

   a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.
   b. Record the configuration settings on the **Settings** tab, and then click **Configure**.
   c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text

logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.

5. Copy the **file.xml**, **iastemplates.xml**, and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server running Windows Server 2012 using the Netsh utility

1. On the source NPS server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps export filename="path\file.xml" exportPSK=YES
   ```

   Replace *path* with the directory location where you want to save the source server configuration file, and replace *file* with the name of the .XML file that you want to save.
2. Confirm that a message appears indicating that the export to file was successful.
3. On the source server, type the following command and then press Enter:

   ```
   netsh nps show sqllog > path\sql.txt
   ```

   Replace *path* with the directory location where you want to save the source server SQL configuration file, and replace *sql* with the name of the .TXT file that you want to save. This file contains the basic configuration for SQL logging that is found on the **Settings** tab in SQL logging properties. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.
4. Copy the **file.xml** and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

To export settings from a source server running Windows Server 2012 using the Windows interface

1. On the source server, open Server Manager.
2. In the Server Manager console tree, click **ALL SERVERS**, then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
3. Right click the root node **NPS**, and then click **Export Configuration**.
4. In the dialog box that appears, select the check box next to **I am aware that I am exporting all shared secrets**, and then click **OK**.
5. Next to **File name**, type **file.xml**, navigate to the migration store file location, and then click **Save**.
6. In the console tree, right-click **Templates Management** and then click **Export Templates to a file**.
7. Next to **File name**, type **iastemplates.xml**, navigate to the migration store file location, and then click **Save**.
8. If you have configured SQL logging, you must manually record detailed SQL configuration settings.

   To record these settings:

a. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties**.

b. Record the configuration settings on the **Settings** tab, and then click **Configure**.

c. Manually record all configuration settings from the Connection and Advanced tabs by copying them into the **sql.txt** file. Alternatively, you can click the **All** tab and enter **Name** and **Value** settings displayed on each line into the **sql.txt** file. For a list of text logging and SQL configuration settings that you need to record manually, see NPS Server Migration: Appendix A - Data Collection Worksheet.

9. Copy the **file.xml**, **iastemplates.xml**, and **sql.txt** files to the migration store file location. This information will be required for configuration of the destination server.

Importing settings to the destination server

Use the following procedures to import the NPS settings from your x86-based or x64-based source server to an x64-based destination server running Windows Server 2012.

- Importing settings from Windows Server 2003
- Importing settings from Windows Server 2008 or Windows Server 2008 R2
- Importing settings from Windows Server 2012

Importing settings from Windows Server 2003

The configuration file **ias.txt** that was exported from the source server is in a format that can be imported to a destination server running Windows Server 2012. If SQL accounting settings were saved, these settings are recorded manually in the **sql.txt** file.

Important

When you migrate the configuration settings of the IAS role service that is running on a 32-bit or a 64-bit Windows Server 2003–based source server to the NPS role service that is running on a Windows Server 2012–based destination server, the import procedure seems to complete successfully. However, the Extensible Authentication Protocol (EAP) method is misconfigured. This occurs because the migration tool generates a faulty parameter that is stored in the configuration text file (ias.txt). For more information about this issue and for a workaround, see The EAP method is configured incorrectly during the migration process from Windows Server 2003 32-bit or a 64-bit to Windows Server 2008 R2 (http://go.microsoft.com/fwlink/?LinkID=181982).

To import settings from a source server running Windows Server 2003

1. Copy the configuration file **ias.txt** that was exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command.
2. On the destination server, use either netsh or Windows PowerShell to import the configuration.

- To use netsh, do the following:

a. Open an elevated command prompt, type the following command and then press Enter:

```
netsh nps import filename="path\ias.txt"
```

Replace *path* with the directory where the **ias.txt** file is located. Verify that a message appears indicating that the import process was successful.

 **Tip**

If the configuration file is located on a network share, provide full path to the file. For example: **netsh nps import filename = "\\fileserver1\Data\ias.txt"**.

- To use Windows PowerShell, do the following:

a. On the **Start** screen, type **PowerShell**, and then click Enter.
b. Switch to the NPS context, enter the following Windows PowerShell command:

Import-Module NPS

c. To import the configuration, enter the following:

Import-NpsConfiguration [-Path] <String>

Replace *String* with the directory where the **ias.txt** file is located. Verify that a message appears indicating that the import process was successful.

 **Tip**

For example:

Import-NpsConfiguration –Path c:\temp\ias.txt

3. If required, configure SQL accounting. To configure SQL accounting:
   a. In the Server Manager console tree, click **ALL SERVERS**, then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
   b. Click **Accounting** and then click **Change SQL Server Logging Properties**.
   c. Manually enter SQL settings from the **sql.txt** file that you created.

Importing settings from Windows Server 2008 or Windows Server 2008 R2

The configuration file **file.xml** that was exported from the source server is in a format that can be imported to a destination server running Windows Server 2012. SQL accounting settings are saved in the **sql.txt** file.

Note

For source servers running Windows Server 2008 R2: If you saved a templates configuration file, **iastemplates.xml**, you must use the Windows interface to import these settings.

To import settings from a source server running Windows Server 2008 or Windows Server 2008 R2

1. Copy the configuration files **file.xml** and **sql.txt** that were exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command.
2. On the destination server, use either netsh or Windows PowerShell to import the configuration.
   - To use netsh, do the following:

     a. Open an elevated command prompt, type the following command and then press Enter:

     ```
     netsh nps import filename="path\file.xml"
     ```

     Replace *path* with the directory where the **file.xml** file is located. Verify that a message appears indicating that the import process was successful.

     Tip

     If the configuration file is located on a network share, provide full path to the file. For example: **netsh nps import filename = "\\fileserver1\Data\file.xml"**.

   - To use Windows PowerShell, do the following:

     a. On the **Start** screen, type **PowerShell**, and then click Enter.

     b. Switch to the NPS context, enter the following Windows PowerShell command:

     Import-Module NPS

c. To import the configuration, enter the following:

Import-NpsConfiguration [-Path] <String>

Replace <*String*> with the directory where the **file.xml** file is located.

Tip

For example:

Import-NpsConfiguration –Path c:\temp\file.xml

d. Confirm that no errors were reported by Windows PowerShell.

3. If required, configure SQL accounting. To configure SQL accounting:
   a. In the Server Manager console tree, click **ALL SERVERS**, then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
   b. Click **Accounting** and then click **Change SQL Server Logging Properties**.
   c. Manually enter SQL settings from the **sql.txt** file.

Importing settings from Windows Server 2012

The configuration file **file.xml** that was exported from the source server is in a format that can be imported to a destination server running Windows Server 2012. SQL accounting settings are saved in the **sql.txt** file. If you saved a templates configuration file, **iastemplates.xml**, you must use the Windows interface to import these settings.

To import settings from a source server running Windows Server 2012

1. Copy the configuration files **file.xml** and **sql.txt** that were exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command.
2. On the destination server, open an elevated command prompt, type the following command and then press Enter:

   ```
   netsh nps import filename="path\file.xml"
   ```

   Replace *path* with the directory where the **file.xml** file is located. Verify that a message appears indicating that the import process was successful.

   Tip

   If the configuration file is located on a network share, provide full path to the file. For example: **netsh nps import filename = "\\fileserver1\Data\file.xml"**.

   The following Windows PowerShell command performs the same function:

   ```
   Import-NpsConfiguration –Path c:\temp\file.xml
   ```

3. If required, configure SQL accounting. To configure SQL accounting:
   a. In the Server Manager console tree, click **ALL SERVERS**, then from the list of

servers in the right pane, right-click the relevant server and select **Network Policy Server**.

b. Click **Accounting** and then click **Change SQL Server Logging Properties**.

c. Manually enter SQL settings from the **sql.txt** file.

Using the NPS console to migrate NPS settings

You can also use the Windows interface on the destination server to import configuration settings.

To import settings from a source server using the Windows interface

1. Copy the configuration files **file.xml**, **iastemplates.xml**, and **sql.txt** that were exported to the migration store file location to the destination NPS server. Alternatively you can import configuration settings directly from the migration store file location by supplying the appropriate path to the file in the import command. If you have custom settings that were recorded using the NPS Server Migration: Appendix A - Data Collection Worksheet, these must be configured manually on the destination server.
2. On the destination server, open Server Manager.
3. In the Server Manager console tree, click **ALL SERVERS**, and then from the list of servers in the right pane, right-click the relevant server and select **Network Policy Server**.
4. To import template configuration settings, follow steps 5 to 13. If you do not have template settings, skip to step 7.
5. In the console tree, right-click **Templates Management** and then click **Import Templates from a file**.
6. Select the template configuration file **iastemplates.xml** that you copied from the source server and then click **Open**.
7. In the console tree, right-click **NPS** and then click **Import Configuration**.
8. Select the configuration file **file.xml** or **ias.txt** that you copied from the source server and then click **Open**.
9. Verify that a message appears indicating the import was successful.
10. Configure SQL accounting if required using the **sql.txt** file and the data collection worksheet. To configure SQL accounting, follow steps 11 to 13.
11. In the NPS console tree, click **Accounting** and then click **Change SQL Server Logging Properties** in the details pane.
12. Modify the properties on the **Settings** tab if required, and then click **Configure** to enter detailed settings.
13. Using information recorded in the **sql.txt** file, enter the required settings on the **Connection** and **Advanced** tabs, and then click **OK**.

See Also

Migrate Network Policy Server to Windows Server 2012

NPS Server Migration: Preparing to Migrate
NPS Server Migration: Verifying the Migration
NPS Server Migration: Post-migration Tasks
NPS Server Migration: Appendix A - Data Collection Worksheet

NPS Server Migration: Verifying the Migration

After the migration of your Network Policy Server (NPS) server is complete, you can perform some tasks to verify that the migration was successful.

Verifying NPS Migration

To verify the functionality of NPS on the destination server, confirm that the service is running, that the correct configuration was migrated, and that client computers can authenticate successfully.

To verify NPS migration

1. To verify that the NPS service is running on the destination server, type the following command at an elevated command prompt on the destination server and then press ENTER.

   ```
   sc query ias
   ```

   In the command output, verify that **RUNNING** is displayed next to **STATE**.
2. To verify that the source NPS configuration has been migrated to the destination server, type the following command at an elevated command prompt on the destination server and then press ENTER:

   ```
   netsh nps show config
   ```

   Verify that the destination server is not using default NPS settings. For example, default settings display a single policy under **Connection request policy configuration** with the name **Use Windows authentication for all users**.
3. To verify that the NPS console on the destination server displays the correct settings, type the following command at an elevated command prompt on the destination server and then press ENTER:

   ```
   nps.msc
   ```

   a. The NPS console will open. In the console tree, click **Accounting**, click **Change SQL Server Logging Properties**, click **Configure**, and verify that the correct settings are displayed on the **Connection** and **Advanced** tabs.

b. In the NPS console tree, click **Policies** and then click **Connection Request Policies**, **Network Policies**, and **Health Policies**. For each type of policy, verify that the correct policies are displayed.

c. In the NPS console tree, click **RADIUS Clients and Servers** and then click **RADIUS Clients and Remote RADIUS Server Groups**. Verify that the correct RADIUS clients and remote RADIUS server groups are displayed.

d. In the NPS console tree, click **Network Access Protection**, and then click **System Health Validators** and **Remediation Server Groups**. Verify that the correct Network Access Protection (NAP) related settings are displayed.

e. In the NPS console tree, click **Templates Management**. If the source server was running Windows Server 2008 R2, verify that the correct templates settings are displayed.

f. In the NPS console tree, right-click **NPS**, click **Properties**, and then click the **Ports** tab. Verify that the correct **Authentication** and **Accounting** ports are displayed.

4. To verify the configuration of authentication methods, you must manually review settings in connection request policy and network policy. Certificate based EAP methods require that the proper certificate is chosen, and might require that you provision a computer certificate on the destination server.

Verifying authentication methods

a. If you use certificate based EAP methods, your destination server might already be provisioned with a suitable certificate through autoenrollment. You might also be required to manually enroll the destination server with a computer certificate. For an overview of certificate requirements for network authentication, see Network access authentication and certificates (http://go.microsoft.com/fwlink/?LinkId=169625).

b. To view certificates associated with EAP methods, click **Start**, click **Run**, type **nps.msc**, and press ENTER.

c. In the NPS console tree, open **Policies** and then open the type of policy you are using to perform authentication. For example, if the option to **Override network policy authentication settings** is enabled on the **Settings** tab in a connection request policy, then authentication is performed in connection request policy. Otherwise, authentication is performed in network policy. Authentication can be configured in both types of policies.

d. For connection request policy, double-click the policy name and then click the **Settings** tab. For network policy, double-click the policy name and then click the **Constraints** tab.

e. Click **Authentication Methods**, and then under **EAP Types** click the name of the certificate-based authentication method. For example: **Microsoft: Protected EAP (PEAP)** or Microsoft: Smart Card or other certificate.

f. Click **Edit**, verify that the correct certificate is chosen next to **Certificate issued** or **Certificate issued to**, and then click **OK**.

Note

Client computers using certificate based authentication methods must trust the certification path for this certificate.

5. To verify that client computers can authenticate using the destination server, attempt to connect to the network using client VPN connection, an 802.1X connection, or another connection that requires successful RADIUS authentication for network access.

Verifying client connections

a. To verify that client computers are successfully connecting to the network, click **Start**, click **Run**, type **eventvwr.msc**, and then press ENTER.
b. In the event viewer console tree, open **Custom Views\Server Roles\Network Policy and Access Services**.
c. In the details pane, verify under **Event ID** that event number 6272 is displayed.
d. Events 6273 or 6274 indicate that client authentication attempts are unsuccessful.
e. If no events are displayed, client connection requests are unable to reach the destination server, or the server is not logging authentication attempts.

See Also

Migrate Network Policy Server to Windows Server 2012

NPS Server Migration: Preparing to Migrate

NPS Server Migration: Migrating the NPS Server

NPS Server Migration: Post-migration Tasks

NPS Server Migration: Appendix A - Data Collection Worksheet

NPS Server Migration: Post-migration Tasks

After all migration steps are complete and you have verified the migration of the Network Policy Server (NPS) role service, perform the following post-migration tasks.

Post migration tasks

After verifying NPS configuration is working on the destination server, the following steps need to be performed:

To decommission a source server using the same host and IP address

1. Remove the source server from your Active Directory domain.

2. Shut down the source server.
3. Rename the destination server from *tempNPS* to the name of the source server and configure the same static IP address as that used by the source server.
4. Perform verification steps in NPS Server Migration: Verifying the Migration with the updated host name and IP address configured on the destination server.

To decommission a source server using a different host and IP address

1. NPS server name/ IP address should be updated on Remote RADIUS servers and RADIUS clients. It requires manual update of the configurations on RADIUS clients and Network Access Servers (NAS). Please refer to your RADIUS client configuration guide for more information.
2. Perform verification steps in NPS Server Migration: Verifying the Migration.
3. When the destination server has been configured, tested, and verified, then the NPS role on the source server may be retired.

Restoring the role in the event of migration failure

If the destination server is deployed simultaneously with the source server using a different host name and IP address, then the migration can be reversed by changing RADIUS clients, remote RADIUS server groups, and network access device settings to use the source NPS server name and IP address. If the destination server is replacing the source server using the same host name and IP address, then the destination server will need to be renamed, the IP address must be updated, and the destination server must be removed from the domain to reverse the migration and bring the source server back online.

See Also

Migrate Network Policy Server to Windows Server 2012

NPS Server Migration: Preparing to Migrate

NPS Server Migration: Migrating the NPS Server

NPS Server Migration: Verifying the Migration

NPS Server Migration: Appendix A - Data Collection Worksheet

NPS Server Migration: Appendix A - Data Collection Worksheet

Migration data collection worksheet

You can use this migration data collection worksheet to collect data about your source server and help ensure that the destination server functions properly after the migration.

NPS data worksheet

#	Source server essential settings	Setting values
1	**Server name** At a command prompt, type the following command, and then press ENTER. `ipconfig /all` The host name of a server is the first part of the fully qualified domain name (FQDN). The FQDN is the full computer name, including both the host name and the primary DNS suffix, separated by dots (.). For example, the FQDN of a computer named *host* with a primary DNS suffix of *example.microsoft.com* is *host.example.microsoft.com*.	Computer host name: _____ FQDN: _____
2	**Authentication, authorization, and accounting (AAA) roles** Determine what types of network access requests are validated using the RADIUS protocol on the source server.	Check all that apply (√) ☐ Network Access Protection (NAP) ☐ RADIUS server for dial-Up or VPN connections ☐ RADIUS server for 802.1X wireless or wired connections
3	**Text logging** Record the path and settings used for text logging. By default, local file accounting logs are stored in **%windir%\system32\LogFiles**.	Local file logging directory: _____ Format: _____ Create a new log file: _____
4	**SQL settings**	Application Name:

#	Source server essential settings	Setting values
	Manually record any customized SQL data link properties.	_____ Auto Translate: _____ Connect Timeout: _____ Current Language: _____ Data Source: _____ Extended Properties: _____ General Timeout: _____ Initial Catalog: _____ Initial File Name: _____ Integrated Security: _____ Locale Identifier: _____ Network Address: _____ Network Library: _____ Packet Size: _____ Password: _____ Persistent Security Info: _____ Replication server name connect option: _____ Tag with column collation when possible: _____ Use Encryption for Data: _____ Use Procedure for Prepare:

#	Source server essential settings	Setting values
		_____ User ID: _____ Workstation ID: _____

See Also

Migrate Network Policy Server to Windows Server 2012

NPS Server Migration: Preparing to Migrate

NPS Server Migration: Migrating the NPS Server

NPS Server Migration: Verifying the Migration

NPS Server Migration: Post-migration Tasks

Migrate Print and Document Services to Windows Server 2012

Overview

This article provides guidance to migrate a print server running Windows Server® 2003, Windows Server 2003 R2, Windows Server 2008, or Windows Server 2008 R2 operating systems to a server running the Windows Server 2012 operating system with the Print and Document Services role installed. This includes cross-architecture and stand-alone migrations, as well as configurations for a server in a cluster. This document provides step-by-step instructions for migrating from old hardware to new hardware and consolidating print servers.

Print and Document Services enables print server tools and configures the server to act as a print server. Print and Document Services is not dependent on any other features or roles. However, some specific network configurations, clients, and hardware may require you to install additional features or enable certain services.

This guide provides you with instructions for migrating an existing print server to a server that is running Windows Server 2012. This guide does not contain instructions for migration when the source server is running multiple roles. If your server is running multiple roles, we recommend that you design a custom migration procedure specific to your server environment, based on the information provided in other role migration guides. Migration guides for additional roles are available at Migrate Roles and Features to Windows Server 2012.

Caution

If your source server is running multiple roles, some migration steps in this guide, such as those for computer name and IP configuration, can cause other roles that are running on the source server to fail.

To manage the migration process, use one of the following:

- The Printer Migration Wizard, which you access through Print Management, a snap-in in Microsoft Management Console (MMC).
- The Printbrm.exe command-line tool.

You can perform the migration locally or remotely, and from either a client computer or server.

Important

As a best practice, run the Printer Migration Wizard or Printbrm.exe from a computer running Windows Server 2012 or Windows 8 to ensure that you are using the newest version of the migration tools that have the latest updates and features. You can run these tools either locally on the server or remotely from any other computer running Windows Server 2012 or Windows 8.

Remember that if you are running printbrm over the network to remote servers, the **Print$** share must exist on both the source and target servers and the Remote Registry Service must be running.

For more information about installing and using these tools, see Access the migration tools.

Notes

The Print Management snap-in is not available in Windows Server 2003. However, it is available in Windows Vista Enterprise and Windows Vista Ultimate, which enables you to migrate from Windows Server 2003. It is also available in Windows Server 2008, Windows Server 2008 R2 and Windows Server 2012. For more information about migrating from Windows Server 2003, see Preparing to Migrate.

The Print Management snap-in and the Printbrm.exe command-line tool are not available for the Server Core installation option. To migrate from a print server running on a Server Core installation, use a server running the Printer Migration Wizard, Windows Vista Enterprise, or Windows Vista Ultimate. For more information about migrating from a server running a Server Core installation, see Preparing to Migrate.

There is no equivalent to Print and Document Services for Windows® client operating systems.

You can migrate Print and Document Services from the destination server or from any client with the following:

- The Printer Migration Wizard (provided that the client is running one of the supported operating systems listed in the Supported operating systems matrix).
- Remote access to the destination server.
- Access to the printer settings file created when you back up the source server.

Note

All commands in this guide are case-insensitive unless specifically noted.

About this guide

This guide is designed as a step-by-step tutorial for migrating print servers.

Target audience

This document is intended for IT administrators and other knowledge workers who are responsible for the operation and deployment of print servers in a managed environment.

What this guide does not provide

This document does not provide guidance for the following:

- Upgrading roles on the same computer
- Migrating printer configurations during client installations of Windows
- Migrating settings for a server that is not being used as a print server
- Recovering server information that was not properly saved prior to migration for in-place upgrades
- Instructions for migrating more than Print and Document Services

Supported migration scenarios

You must have access to the Printer Migration Wizard to migrate print servers. For more information about supported scenarios and limitations, see Preparing to Migrate.

Supported operating systems

The following table outlines the supported operating systems for migration covered in this guide.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 with Service Pack 2	Windows Server 2012, including full, MinShell, and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2003 R2	Windows Server 2012, including full, MinShell, and Server Core installation options	x64-based
x86- or x64-based	Windows Server 2008, both full and Server Core	Windows Server 2012, including full,	x64-based

Source server processor	Source server operating system	Destination server operating system	Destination server processor
	installation options	MinShell, and Server Core installation options	
x64-based	Windows Server 2008 R2	Windows Server 2012, including full, MinShell, and Server Core installation options	x64-based
x64-based	Server Core installation option of Windows Server 2008 R2	Windows Server 2012, including full, MinShell, and Server Core installation options	x64-based
X64-based	Windows Server 2012	Windows Server 2012, including full, MinShell, and Server Core installation options	x64-based
X64-based	Server Core and MinShell installation options of Windows Server 2012	Windows Server 2012, including full, MinShell, and Server Core installation options	x64-based

The versions of operating systems shown in the preceding table are the oldest combinations of operating systems and service packs that are supported. Newer service packs, if available, are supported.

The Foundation, Standard, Enterprise, and Datacenter editions of Windows Server are supported on full, Server Core, and MinShell installation options, as either source or destination servers.

All versions of Windows Server 2012 are x64-based. Migrating to an x86-based server is not supported.

Migrations between physical operating systems and virtual operating systems are supported.

Note
Both x86-based and x64-based migrations are supported for Windows Server 2003 and Windows Server 2008. All editions of Windows Server 2012 are x64-based.

You might prefer the migration process, rather than an upgrade, even when the hardware is native x64-based. An example would be a case where there is increased use of the server and

there is a server role split (in which the source server has more than one server role)—and you decide to separate the roles onto several additional x64-based servers. In this case, migration of individual server roles to other servers might be the best solution.

The server administrator can choose which parts of an existing installation to migrate, but along with the server role, this usually includes configuration, data, system identity, and operating system settings.

Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shares from a computer that is running Windows Server 2008 in the French language to a computer that is running Windows Server 2012 in the German language.

Note

The system UI language is the language of the localized installation package that was used to set up the Windows operating system.

Cross-architecture migration (such as migrating from an x86-based server to an x64-based server) is supported. The source server must have print queue drivers installed for both the source and destination server architectures. If a print queue does not have a driver for the destination server architecture, then it will not be migrated. Similarly, verify that the destination server contains drivers for each supported architecture.

Supported role configurations

The Supported operating systems matrix provides a complete listing of the supported migration scenarios.

Some migration scenarios require additional preparation. For more information about these scenarios, see Appendix B - Additional Destination Server Scenarios.

Supported role services and features

The Printer Migration Wizard migrates:

- Print queues.
- Shared printer settings.
- Printer drivers in use by the print spooler.
- Any security settings specific to the installed printer.

Migrating from x86-based to x64-based v3 printer drivers

There are several things you must consider when managing migrations using v3 print drivers. The first is that a print queue cannot function without the native printer driver for the server architecture (x86 or x64) on which it exists. Since Windows Server 2012 is a 64-bit only operating system, it is important that you have 64-bit drivers installed for all of your printers if you are

migrating from a 32-bit system. The most difficult transition is from 32-bit to 64-bit servers in an organization with 32-bit clients since it is common to have third-party 32-bit printer drivers that do not have 64-bit equivalents. During the print configuration restoration for cross-platform scenarios, if the backup file does not contain driver binaries for the platform of the target server, the drivers will be installed from the target server's driver store, if available.

As a best practice, when migrating from x86-based to x64-based v3 drivers:

1. Verify that x64-based versions of the drivers are available.
2. If you are unable to verify their availability, back up the source server before the migration.
3. Install the x64-based drivers on the source server so that you can determine if there any problems or conflicts before the migration process.
4. If there are conflicts or problems on the destination server after the migration, roll back the migration. For more information, see "Roll back migration on the source server" in Post-Migration Tasks.

Unsupported scenarios

The Printer Migration Wizard does not migrate the following:

- Other services or settings that specific printers may rely on, such as Line Printer Remote (LPR), Internet Printer Protocol (IPP), or Web Services on Devices (WSD). You must enable or install these features before restoring the source print server configuration. For additional information, see "Roll back migration on the source server" in Post-Migration Tasks.
- Local bus printers (LPT and USB), although they are shown during backup. For additional information, see Appendix B - Additional Destination Server Scenarios.
- Plug and play printers. However, plug and play printer drivers will be migrated. For additional information, see Appendix B - Additional Destination Server Scenarios.
- Any print jobs currently in the printer queue.
- Any system or print administrators, or permissions. If you want to retain the same system or print administrators on the destination server as on the source server, you will need to manually add these administrators to the destination server.

Print and Document Services migration overview

While the original server is still running, use the Printer Migration Wizard or the Printbrm.exe command-line tool to export or back up the print information (such as settings, queues, and drivers) in a printer settings file. Then, import or restore this backup image to a destination server running Windows Server 2012 that has been configured to run as a print server.

Some migration scenarios require additional preparation. For more information about these scenarios, see Appendix B - Additional Destination Server Scenarios.

To migrate printers from a server running Windows Server 2003 or a Server Core installation to a server running Windows Server 2008 R2, you must use a computer running the Printer Migration Wizard to remotely manage the server running Windows Server 2003 or a Server Core installation. Using this computer, you can store the printer settings file (containing information

about the printers you want to migrate, such as settings, queues, and drivers) from the server running Windows Server 2003 or a Server Core installation to a file share. You can then use the Printer Migration Wizard to migrate the printers from the file share to the server running Windows Server 2008 R2.

For more information about accessing the Printer Migration Wizard, see Preparing to Migrate.

Note
The **Printing-Server Core** role must be installed on a server running a Server Core installation from which you want to migrate.

Migrate print servers (overview)

The following list provides an overview of the steps to migrating the print servers.

- Access the migration tools
- Prepare the destination server
- Prepare the source server
- Back up the source server
- Restoration
- Verify the migration
- Post-migration

Impact of migration

The objective of the migration process is that the destination server is able to perform the same functions as the source server did, without client computers on the network being aware that the migration has taken place. The following sections describe the impact of migration.

Impact of migration on the source server

The source server is not impacted by print server migration until the destination server takes over as the active server (typically when the name or IP address of the source server is assigned to the destination server). At that point, the source server no longer services print requests that target the print server.

Impact of migration on other computers in the enterprise

If the destination server replaces the source server in the network (same name or IP address), then there should be no impact to other computers in the enterprise.

If the destination server has a different name or IP address than the source server, then all clients with existing print connections must delete and recreate those print connections so that they target the destination server.

Permissions required to complete migration

Administrative permissions are required on both the source print server and the destination print server.

Permissions required to complete migration on other computers in the enterprise

If the destination server replaces the source server in the network, then no permissions are required on other computers in the enterprise. If the destination server has a different name or IP address, then the permissions required on other computers may vary depending on Group Policy settings, Windows Update access, and driver availability.

Estimated duration

The time required to migrate a print server will vary from server to server, depending on the following:

- The number of queues being migrated.
- The number of individual drivers needed for the queue.
- The size of a given driver, in terms of its file size and the number of files.
- The configuration of the server.

Migrating a single printer queue with a typical x86-based and x64-based driver can range from five seconds to several minutes, depending on the factors listed above. Because of this range, a typical migration can take anywhere from less than an hour to several hours.

See Also

Preparing to Migrate

Migrating the Print and Document Services Role

Verifying the Migration

Post-Migration Tasks

Appendix A - Printbrm.exe Command-Line Tool Details

Appendix B - Additional Destination Server Scenarios

Appendix C - Printbrm Event IDs

Preparing to Migrate

Access the migration tools

The Printer Migration Wizard and the Printbrm.exe command-line tool support all migrations to Windows Server 2012.

Note

Although the Printer Migration Wizard supports migrations from servers running Windows Server 2003 or a Server Core installation, it cannot run on these servers directly.

To access the Printer Migration Wizard

Open the Print Management snap-in on computers running operating systems previous to Windows 8

1. If necessary, enable the **Administrative Tools** menu, which is hidden by default on Windows-based client operating systems.
   a. Right-click **Start**, and then click **Properties**. The **Start Menu and Taskbar Properties** dialog box opens.
   b. On the **Start Menu** tab, click **Customize**. The **Customize Start Menu** dialog box opens.
   c. Under **System Administrative Tools**, select **Display on the All Programs menu** or **Display on the All Programs menu and the Start menu**.
2. In the **Administrative Tools** menu, click **Print Management**.

Tip

Alternatively, you can click **Start**, and type **printmanagement.msc** in the search text box.

Open the Print Management snap-in on computers running Windows Server 2012

- Open Server Manager, click **Tools**, and then click **Print Management**.

Open the Print Management snap-in on computers running Windows 8

- From the Start screen, type **printmanagement.msc** and click **printmanagement** when it appears in the search results.

Tip

If you run the Print Management Console remotely (for example, from a client computer), simply add the print server to your console (right-click **Print Servers**, click **Add/Remove Servers**) and continue the migration from there.

Start the Printer Migration wizard

- In Print Management, right-click **Print Management** and click **Migrate Printers**.

Important
The Print Management snap-in filter settings will not be migrated and need to be saved independently of the printer migration.

To access the Printbrm.exe command-line tool

1. To open a Command Prompt window, click **Start**, click **All Programs**, click **Accessories**, right-click **Command Prompt**, and then click **Run as administrator**.

   To open a command Prompt window on a computer running Windows 8 or Windows Server 2012, right-click the start charm and click Command Prompt (Admin)
2. Type:

   ```
   %WINDIR%\System32\Spool\Tools\Printbrm.exe
   ```

To view the complete syntax for this command, at a command prompt, type:

```
Printbrm.exe /?
```

For a listing of the available syntax for the **Printbrm.exe** command, see Appendix A - Printbrm.exe Command-Line Tool Details.

Prepare the destination server

The second step in the migration process is to prepare the destination server.

Hardware requirements for the destination server

There are no specific hardware requirements for being a print server beyond those for the version of the server operating system you are using.

The amount of disk space needed for a migration is dependent on the number of print drivers to be migrated and the size of the drivers. Because print drivers can vary widely in size, the amount of disk space can range from one megabyte to several gigabytes.

Software requirements for the destination server

Verify that hard drive space is sufficient on the destination server for the backup.

No additional software is needed other than the necessary drivers required for the printers to be hosted. Migrate these drivers from the source server.

For cross-architecture migrations, verify that the destination server contains drivers for each supported architecture.

Installing the Print and Document Services role on the destination server

You must install the Print and Document Services role on the destination server before you begin the migration process. For more information on installing this and other server roles, see the Print and Document Services overview.

Preparing for cross-architecture migrations

If you are migrating from the x86-based architecture of Windows Server 2003 or Windows Server 2008 to the x64-based architecture of Windows Server 2012, you should install x64-based drivers on the source server before creating the backup file. The migration process copies all installed drivers from the source server to the destination server. It recreates the printer queues on the destination server if the printer settings file contains the x64-based drivers.

Verify that each print queue on the source server has a driver installed for the operating system on the destination server before creating the printer settings file. For example, if you are migrating an x86-based source print server to an x64-based destination print server, verify that each print queue has an x64-based driver installed before you create the printer settings file. Any print queue that does not have a cross-architecture driver installed will not be migrated to the destination server.

To install cross-architecture drivers for a printer, you can use:

- The **Add Printer Driver Wizard**, which is available in the Print Management snap-in.
- The **Printer Properties** dialog box, which is available through the Printers folder in the Control Panel.

As a best practice, you need to install a driver with the same name as the native architecture. To add the x86-based driver to the x64-based destination server, use the x86-based client to remotely open the x64-based server using Windows Explorer and navigate to the remote printer folder and add the driver. To install an x64-based driver on the x86-based source server, use the x64-based client to remotely open the x86-based server using Windows Explorer and navigate to the remote printer folder and add the driver.

Tip

In many cases, it can take you a long time to update all the print drivers for all your print queues. To save time, you may want to update just the most used print queues first, and gradually update the others when you have time. To save time, you can set the existing print queues to the **Generic/Text Only** driver for migration and later switch them to the OEM driver at your convenience. Most printers allow basic printing using the Generic Text driver. For more information, see Cross-Architecture print server migrations: Speeding up the migration process at the Microsoft TechNet Blogs web site.

Preparing for additional scenarios

In the following instances, installing a feature on your destination server may require additional preparation before you migrate to it:

- The server hosts Line Printer Remote (LPR) printers.
- The server offers Internet Printing Protocol (IPP) printer connections.
- The server hosts Web Services on Devices (WSD) printers.
- The server is in a server cluster.
- The server hosts plug and play printers.

For more information on these scenarios, see Appendix B - Additional Destination Server Scenarios.

Prepare the source server

Simple system-to-system migrations require no preparation for the source server. However, additional preparation is required for cross-architecture migrations. If performing the migration as quickly as possible is a priority, remove unused drivers, ports, and queues before starting the migration to improve its speed after verifying with users that the items to remove are no longer in use. In general, however, minimize changes to the source server environment to ensure you can roll back to this environment if necessary.

Caution

If your source server is running multiple roles, renaming the source server or changing its IP address can cause other roles that are running on the source server to fail.

Notes

You should delete native print drivers that are not currently associated with a print queue because these drivers increase the size of the printer settings file unnecessarily. The print spooler will not allow a native print driver that is currently associated with a print queue to be deleted.

The Print Spooler service will use non-native drivers. It routes these drivers to the Print Server service when a non-native client connects to a print queue and has to download a driver. You should remove any unused drivers and print queues.

Do not delete a non-native driver with a corresponding native print driver that is associated with a print queue. In this instance, the Print Spooler service will not prevent the non-native driver from being deleted. If the non-native driver's architecture matches the destination server's architecture, then you must block the driver's deletion. Cross-architecture drivers will never appear to be loaded by the Print Spooler service. Administrators should only delete them after confirming the drivers are no longer needed.

To install cross-architecture drivers using the Print Management snap-in on computers running Windows Vista and Windows Server 2008

1. Open the Print Management snap-in. Click **Start**, click **Administrative Tools**, and then click **Print Management**.
2. In the Print Management tree, under **Print Servers**, click the print server you want.
3. Under the print server, right-click **Drivers** and then select **Add Driver** to open the **Add**

Printer Driver Wizard.

4. Follow the steps as indicated by the wizard.

To install cross-architecture drivers by using only the Printer Properties dialog box on computers running Windows XP and Windows Server 2003

1. Click **Start**, click **Control Panel**, and double-click **Printers**.
2. Select **Printer**. Right-click **Sharing**.
3. Click **Additional Drivers** and select **Processor** from the list.
4. Follow the instructions in the dialog boxes to install the correct driver. Only install the driver associated with the printer you are administering.

Note

You can only add a cross-architecture driver if you have already installed a native architecture version of the same driver.

See Also

Migrate Print and Document Services to Windows Server 2012

Migrating the Print and Document Services Role

Verifying the Migration

Post-Migration Tasks

Appendix A - Printbrm.exe Command-Line Tool Details

Appendix B - Additional Destination Server Scenarios

Appendix C - Printbrm Event IDs

Migrating the Print and Document Services Role

Back up the source server

The fourth step in the migration process is to back up your source server data to a printer settings file using either the Printer Migration Wizard or the **Printbrm.exe** command-line tool in preparation for exporting printer queues, print drivers, and printer settings.

Important

As a best practice, run the Printer Migration Wizard or Printbrm.exe from a computer running Windows Server 2012 or Windows 8 to ensure that you are using the newest version of the migration tools that have the latest updates and features. You can run

these tools either locally on the server or remotely from any other computer running Windows Server 2012 or Windows 8.

Remember that if you are running printbrm over the network to remote servers, the **Print$** share must exist on both the source and target servers and the Remote Registry Service must be running.

To back up the source server using the Printer Migration Wizard

1. Open the Print Management snap-in.
2. Do one of the following:
   - In the **Print Management** window, right-click **Print Management**, and then click **Migrate Printers** to open the Printer Migration Wizard. Make sure that **Export printer queues and printer drivers to a file** is selected, and then click **Next**. In the **Select a print server** window, select the print server to be migrated, and then click **Next**.
   - In the **Print Management** tree, under **Print Servers**, right-click the print server that contains the printer queues to migrate, and then click **Export printers to a file** to open the Printer Migration Wizard.
3. Review the list of items to be exported, and then click **Next**.
4. In the **Export printer data to** box, enter the path to the printer settings file to use, or browse to the location where you want to store the file. Click **Next** to export the printer-specific information for the server to this file.
5. Verify that the printer settings file is stored on a resource that will be available to the destination server. Optimally, store it on a network share. Click **Finish**.

The backup file that you create by using either the Printer Migration Wizard or the Printbrm.exe tool inherits the permissions allowed by your user credentials. Only you can access the file if you saved the file directly to a share during the backup file creation process. You must either change the file permissions on the **Security** tab of the file's **Properties** dialog box, or you must perform any restorations or migrations by using that file yourself. If you create the backup file on the computer from which you are running the migration and later copy the file to a share, then file access permissions are inherited from the destination folder.

To back up the source server using the Printbrm.exe command-line tool

1. Open an administrator Command Prompt window.
2. Perform a remote print backup. To do this, type the following command in the **%WINDIR%\System32\Spool\Tools** folder at the command prompt, in which *Source Computer1* is the Universal Naming Convention (UNC) name of the source computer, and *Printer1 Settings* is the name of the printer settings file to back up.

   ```
   Printbrm -s \\<Source Computer1> -b -f <Printer1
   Settings>.printerExport
   ```

   Notes

   The Printer Migration Wizard and the Printbrm.exe command-line tool only

support a printer settings file that is created by the migration tool you are using. For example, .cab file backups that were created by using the Printer Migration Wizard are not supported. To view the complete syntax for the **Printbrm.exe** command, type **Printbrm.exe /?** in a Command Prompt session.

Only TCP/IP, WSD, and LPR ports will be migrated. The Printer Migration Wizard will not migrate printers attached through USB, LPT, or other local ports. For more information about these scenarios and migrating Plug and Play printers, see Appendix B - Additional Destination Server Scenarios.

Cross-architecture migrations

For cross-architecture migrations, verify that each print queue has a driver installed on the source server that is compatible with the operating system on the destination server before creating the printer settings file on the source server. For example, if you are migrating an x86-based source print server to an x64-based destination print server, verify that each print queue has an x64-based driver installed before you create the printer settings file. Any print queue that does not have a cross-architecture driver installed will not be migrated to the destination server.

Restoration

The fifth step in the migration process is to restore the printers to the destination server, using the printer settings file you created.

Before beginning the migration process, verify that you installed the Print and Document Services role on the destination server as part of your preparation.

To restore printers to the destination server using the Printer Migration Wizard

1. On the source server, stop the Print Spooler service for all printers so you can preserve all print jobs prior to the migration.
   a. Open Computer Management. Click **Start**, click **Control Panel**, double-click **Administrative Tools**, and then click **Computer Management**.
   b. In the console tree, expand **Services and Applications**.
   c. In the console tree, under **Services and Applications**, click **Services**.
   d. In the details pane, do one of the following to stop the service:
      i. Right-click **Print Spooler** and select **Stop**.
      ii. Double-click **Print Spooler**. On the **General** tab, under **Service Status**, click **Stop**.
2. From the computer that is running the Printer Migration Wizard, on the **Administrative Tools** menu, click **Print Management**.
3. Do one of the following:
   - Right-click **Print Management**, and then click **Migrate Printers** to open the Printer Migration Wizard. Select **Import printer queues and printer drivers from a file,** and then click **Next**.

- In the **Print Management** tree, under **Print Servers**, right-click the destination print server, and then click **Import printers from a file** to open the Printer Migration Wizard.

4. Specify the printer settings file created in the **Back up the source server** section, and then click **Next**.
5. Review the list of items to be imported, and then click **Next**.
6. In the **Import Mode** list, indicate whether you want to keep or overwrite existing printers.

   If the printer settings file contains a printer already on the destination server, the printer is not restored, and the existing printer on the destination server is not changed.
7. In the **List in the directory** list, indicate which printers to list on the destination server.
8. Optionally, indicate whether you want to convert LPR ports to standard port monitors when you migrate.
9. Click **Next** to import the printers.
10. Click **Finish**.

    Note

    It is recommended that you review the Application events that have a PrintBRM source to determine whether any additional actions are needed. The restored printers are shared in the same manner in which they were shared previously.
11. To view details of the migration, click **Open Event Viewer**. For more information, see "Verify the Migration" in Verifying the Migration. If you identify Error 30 in the Event Viewer, see "Troubleshooting" and "Migrating cross-platform driver language monitors" in Post-Migration Tasks for instructions on resolving the error.

To restore printers to the destination server using the Printbrm.exe command-line tool

1. Open an administrator Command Prompt window.
2. Type the following command in the **%WINDIR%\System32\Spool\Tools** folder at the command prompt, in which *Source Computer1* is the UNC name of the source computer, and *Printer1 Settings* is the name of the printer settings file to restore.

```
Printbrm -s \\<Source Computer1> -r -f <Printer1
Settings>.printerExport
```

See Also

Migrate Print and Document Services to Windows Server 2012

Preparing to Migrate

Verifying the Migration

Post-Migration Tasks

Appendix A - Printbrm.exe Command-Line Tool Details

Appendix B - Additional Destination Server Scenarios

Appendix C - Printbrm Event IDs

Verifying the Migration

Verify the migration

The sixth step in the migration process is to verify that the migration was successful by testing and validating the new print server. The Printer Migration Wizard provides detailed logging of migration events in the Event Viewer.

To verify destination server configuration

1. View event log messages about the migration.
   - If you are managing Print Services migration from a remote client computer, you can view event messages in **Custom Views/Administrative Events** in Event Viewer on the Windows-based client computer.
   - If you are managing migration from the destination server and Print and Document Services is not yet installed, then migration-related events are logged in **Custom Views/Administrative Events** in Event Viewer on the destination server.
   - If you are managing migration from the destination server and Print and Document Services is installed, events are logged to a different location. See the events located at **Applications and Services Logs/Microsoft/PrintBRM/Admin**

     To view details after closing the Printer Migration Wizard, right-click the **Start** charm, and then click **Event Viewer**. In the Event Viewer pane, under **Custom Views**, click **Server Roles**, and then click **Print and Document Services**. In the center pane, click the printer migration event to view details.
2. To verify that each printer queue was migrated to the new server:
   - Manually check the destination server for each printer migrated from the source server.
   - Verify that the printer associated with each printer queue is online.

   To check each printer queue's online status in the Print Management snap-in, under **Print Servers**, click **Printers**. A list of all migrated printers appears in the center pane, listing the printer queue status for each printer. Clients will be unable to print to printers that were not restored successfully, and any connections to a missing printer queue will be invalidated.
3. Check the printer queue settings.
   - Confirm that a printer queue's special settings, permissions, or drivers were preserved during the migration.
   - Check the properties for each queue on the destination server and verify that any special settings are still applicable.
   - If the driver installs any non-standard settings that have been altered as a result of the migration, verify those as well.

   Note

The migration process only preserves printer queue permissions. Other permissions on the source server, such as system permissions (for example, user accounts) and custom permissions, are not migrated using this process.

4. Make any necessary changes, such as adding a port monitor or a new driver.

Rename the destination server to the name of the source server

Temporarily rename the destination server. For example, you can name the destination server the same as the source server with **_NEW** appended to the source server name. After verifying that printers are restored to the destination server, rename the source server (for example, by appending **_OLD** to the source server name), and then use the source server's pre-migration name as the new name of the destination server.

Important

Validating existing printer connections from client computers can only be completed after the destination server name is the same as the pre-migration source server name. Print connections to the server in the period between renaming the source server and renaming the destination server will fail. All migration steps should be complete on the destination server before renaming to ensure that the downtime occurs only between the renaming of the source server and the final renaming of the destination server.

If you are using Active Directory Domain Services (AD DS) to publish printers, do the following to ensure that AD DS does not contain multiple instances of the same printer.

When you restore printers to the destination server, do not publish printers to AD DS. This prevents duplicate printers from being displayed by AD DS before the destination server configuration is verified.

On the source server, you must unpublish printers before renaming the source server. To do this, select all printers in the Print Management snap-in, right-click the selected printers, and then click **Remove from Directory**. This prevents printers from being published twice to AD DS when the source server is renamed.

After renaming the destination server to the source server's original name, you can publish all printers on the destination server to AD DS. To do this, select all printers in the Print Management snap-in, right-click the selected printers, and then click **List in directory**.

To verify configuration of other computers in the enterprise

In most cases, a new print server will not affect other computers in the enterprise. Existing client connections may be corrupted if you make a change to any of the following print server properties:

- The print server name
- The printer name
- The print share name
- The share permissions
- The printer's availability to the server

Print a test page to each printer queue from a client (or set of clients) that had an existing connection to the source server to verify that other computers are not affected by the new print server. In a cross-architecture environment, test each supported architecture.

Print a test job from a client with an existing connection

From a client computer that is configured to print to the source print server, use the existing print queue to print a page to the new server. If you cannot print a test page:

- Determine if one or more of the print server properties listed above have been changed.
- Check whether the destination server is available to the client on the network. Create a new connection to a printer on the destination server to verify that the client and server are communicating normally.

See Also

Migrate Print and Document Services to Windows Server 2012

Preparing to Migrate

Migrating the Print and Document Services Role

Post-Migration Tasks

Appendix A - Printbrm.exe Command-Line Tool Details

Appendix B - Additional Destination Server Scenarios

Appendix C - Printbrm Event IDs

Post-Migration Tasks

Post-migration

The final step in the migration process is determined by whether the migration was successful or unsuccessful.

No post-migration tasks are necessary beyond the standard migration process. If the server state (new settings, drivers, queues, and so on) changed, create and archive a new backup of the server state for recovery purposes.

Success

The next section provides tasks that should be completed after you have successfully migrated the source printer server.

Retire the source server

After taking the source server offline while backing it up, users are unable to print until the migration to the destination server is complete. To minimize the impact, leave the source server

in service while you complete the migration and testing of the destination server. By leaving it in service, you can also add both the source and destination servers to the Print Management snap-in to simplify verifying the restoration.

Once you have validated the installation, rename the source and destination servers and take the source server offline.

1. On the source server, restart the print spooler for all printers so it can finish spooling delayed print jobs. When it finishes, verify that there are no new print jobs.
2. Rename source and destination servers as directed in Verifying the Migration.
3. Follow your enterprise's normal policy for server decommissioning and retirement until retirement of the source server is complete.

Note

If the destination server has already been published in Active Directory Domain Services (AD DS), then the source server must be unpublished.

Failure

The next section provides tasks that should be completed if your migration of the source printer server did not succeede.

Restoring the role in the event of migration failure

Restoring the source server lets you deploy the settings to a new system or use the source server while determining the cause of the failure.

Rollback requirements

Warning

Rollback can only be completed if retirement of the source server has not been started. After you start retiring a source server—that is, you delete any print queues, close any print connections, reformat any drivers, or remove any hardware from the source server—you cannot roll back migration. After you start retiring the source server, the only method of rolling back migration is to restart the Print Services migration process from the beginning.

Estimated time to complete rollback

Rolling back migration involves renaming the source server to its pre-migration name, and renaming the destination server to either its original name, or another name that is not the same as the pre-migration name of the source server. Renaming the source and destination servers can be completed in a few minutes.

Roll back migration on the source server

Rename the source server to its original, pre-migration name. You might have to rename the destination server to a temporary name first.

Roll back migration on the destination server

Rename the destination server, either to its original name, or to another name that is not the same as the original name of the source server.

Troubleshooting

The following sections can help you troubleshoot any migration issues.

Log file locations

Printer migration events are included in the Application log, which is located at %*SystemRoot*%\System32\Winevt\Logs\Application.evtx and can be viewed using Event Viewer. A custom view for Printer Migration Events is available in Event Viewer.

Note

If the Printer Migration Wizard fails, you are directed to Event Viewer to view error messages. If you cannot find an error that explains the failure in Event Viewer, restore the backup by using the Printbrm.exe command-line tool. Error reporting from Printbrm.exe can often provide more detail than what is available in the event log.

Migrating cross-platform driver language monitors

When a cross-architecture migration includes the migration of printer language monitors, an error will occur during the process of restoring the printers to the destination server using the Backup Restore Migration (Printbrm.exe) command-line tool. The reason for the error is that language monitor driver architecture must always be the same as the source server architecture. Therefore, when migrating from x86-based architecture to x64-based platforms, language monitor migration cannot be successful. An error posted to the event log will state that the source architecture is not the same as that of the destination server.

You can recover from the printer restore error on the destination server by manually installing (or reinstalling) the appropriate standard driver for the migrated printer(s) running on that architecture.

Mitigating a failure in the Print Spooler service

If you encounter a failure in the Print Spooler service during print server migration, you can work around the failure. Using policy settings, you can isolate print drivers in separate processes so that print driver failures will not cause the Print Spooler service to fail—which allows the restoration to continue.

To turn on print driver isolation using Group Policy

1. Open the Group Policy Management Console. Right-click a Group Policy Object with the necessary scope, and then click **Edit**.
2. In the console tree under **Computer Configuration**, expand the **Administrative Templates** folder, and then expand the **Printers** folder.
3. Double-click **Execute print drivers in isolated processes**.
4. Click **Enabled**, and then click **OK**.
5. Double-click **Override print driver execution compatibility setting reported by print driver**.
6. Click **Enabled**, and then click **OK**.
7. At a command prompt, type **gpupdate /force** to reapply Group Policy settings.

Additional references

- Install, Deploy, and Migrate to Windows Server 2012
- Windows PowerShell Blog (http://go.microsoft.com/fwlink/?LinkId=128557)

See Also

Migrate Print and Document Services to Windows Server 2012

Preparing to Migrate

Migrating the Print and Document Services Role

Verifying the Migration

Appendix A - Printbrm.exe Command-Line Tool Details

Appendix B - Additional Destination Server Scenarios

Appendix C - Printbrm Event IDs

Appendix A - Printbrm.exe Command-Line Tool Details

Printbrm.exe command-line tool syntax

The following table lists the available printbrm parameters:

Syntax	Description
-s <server name>	Specifies the destination server.

Syntax	Description
-b	Backs up the server to the specified file.
-r	Restores the configuration in the file to the server.
-q	Queries the server or the backup file.
-f <file name>	Specifies the backup file.
-d <directory name>	Unpacks the backup file to the directory (with -r), or repacks a backup file from the directory (with -b).
-o force	Forces overwriting of existing objects.
-p all:org	Publishes all printers in the directory, or publishes printers that were published originally.
-nobin	Omits binary files from the backup.
-lpr2tcp	Converts LPR ports to standard TCP/IP ports on restore.
-c <file name>	Uses the specified configuration file.
-noacl	Removes ACLs from print queues on restore.
-?	Displays Help.

Printbrm enhancements

Printbrm.exe in Windows Server 2012 has several enhancements and improvements, including the following:

- **Supports both v3 and v4 print drivers**

  Windows Server 2012 supports both driver types, so with Printbrm you have full flexibility to back up, restore and configure the drivers you need to support your environment.

- **Supports backup CAB files greater than 2 GB**

  You should use the latest PrintBRM version when performing a migration or restoration. You will avoid CAB file size issues if you use the latest version of Printbrm to migrate and restore your Windows Server 2008 R2 (or previous) servers.

- **General improvements for reporting and error handling conditions during the backup and restore processes**

  These conditions are primarily logged in the Event Log under **Custom Views\Printer Migration Events**.

Printbrm usage scenarios

There are many ways Printprm can be used to make migrating your printers easier and more flexible.

Using the configuration file

You can use a configuration file to customize your printbrm migration for the following purposes:

- Replace printer drivers during a restore operation.

  For example, you might want to import your printers to a new print server, but use new v4 printer drivers.
- Backup / Restore dependent files from third-party Language Monitors
- Backup / Restore dependent files from third-party party Print Processors

For example, to replace your printer drivers, you can backup your printers using the `-nobin` parameter, and then restore the printers using the `-c <file name>` parameter to specify a configuration file with a DriverMap section.

To use a configuration file to specify updated print drivers

1. Backup you printers using the `-nobin` parameter. For example:

   ```
   Printbrm.exe -b -nobin -s \\myoldprintserver -f printers.printerExport
   ```
2. On the new print server computer, manually install the updated printer drivers.
3. Create a BrmConfig.xml configuration file to map the old drivers to the new drivers. For example:

   ```
   <BrmConfig>
   <PLUGINS>
   </PLUGINS>

   <LanguageMonitors>
   </LanguageMonitors>

   <DriverMap>
   <DRV old="OldDriverName1" new="NewDriverName1"/>
   <DRV old="OldDrverName2" new="NewDriverName2"/>
   </DriverMap>

   </BrmConfig>
   ```
4. Restore the printers specifying your configuration file using your configuration file. For

example: `PrintBrm.exe –r –c BrmConfig.xml –f printers.printerExport –o force`

5. Check your installed printers to verify they are installed with the updated printer drivers.

Important

Remember that if you are running printbrm over the network to remote servers, the **Print$** share must exist on both the source and target servers and the Remote Registry Service must be running.

Selectively restoring your printers

After you export the printers from your source server, you can selectively restore the printers and their related objects using the `-d` parameter. You can follow a general procedure to accomplish this:

To selectively restore your printers

1. Export the printer objects from the source server.
2. Restore to a temporary folder using the `-d` parameter.
3. Manually edit the files in the temporary folder.

   Note

   More information about the files created by a printbrm backup is described later.
4. Backup the temporary folder using the `-d` parameter.
5. Import the modified backup file to the target server.

A backup or export operation using PrintBRM produces a compressed file that is used for the restore or import operation. The following XML files are part of the export file in addition to the individual printer driver and configuration files:

- **BrmDrivers** – contains a list of every driver installed on the computer and the driver files for each driver
- **BrmForms** – contains a list of all the installed forms
- **BrmLMons** – will usually contain either Windows NT x86 or Windows x64 as the architecture and a list of port monitors and the port monitor files installed
- **BrmPorts** – contains a list of all the installed printer ports
- **BrmPrinters** – contains a list of all printers that have been installed
- **BrmSpoolerAttrib** – contains information about the spooler directory path and also indicates whether or not the source server was a cluster server

Moving printers to a different domain

If you move printers to a different domain, you will want to prevent the restoration of the print queue's ACLs. Use the `-NoACL` parameter to do this. If you use this parameter, the restored print queues will inherit the permissions of the target print server.

See Also

Migrate Print and Document Services to Windows Server 2012

Preparing to Migrate

Migrating the Print and Document Services Role

Verifying the Migration

Post-Migration Tasks

Appendix B - Additional Destination Server Scenarios

Appendix C - Printbrm Event IDs

Appendix B - Additional Destination Server Scenarios

In some instances, your destination server may require additional preparation before you migrate to it.

If your server hosts Line Printer Remote (LPR) printers

To enable the hosting of LPR printers, install the LPR Port Monitor feature on the server:

1. Open Server Manager.
2. In the **Server Manager** dashboard, click **Add roles and features**. The **Add Roles and Features Wizard** opens.
3. Click **Next** on the **Before you begin** page.
4. Ensure **Role-based or feature-based installation** is select on the **Select installation type** page, and click **Next**.
5. Ensure your destination server is selected on the **Select destination server** page and click **Next**.
6. On the **Select server roles** page, click **Next**.
7. On the **Select Features** page, click **LPR Port Monitor**, click **Next**, and then follow the instructions to complete the installation.

Important

The LPD and LPR Services are deprecated starting with Windows Server 2012. Eventually, they will be completely removed from the product, but they are still available in this release. You should begin planning now to employ alternate methods for any applications, code, or usage that depend on these features. For more information about features or functionalities that have either been removed from the product in the current

release or are planned for potential removal in subsequent releases, see Features Removed or Deprecated in Windows Server 2012.

If your server offers Internet Printing Protocol (IPP) printer connections

To enable the Internet Printing Protocol (IPP):

- When installing the Print and Document Services role, select **Internet Printing**.

This automatically configures IIS and any other necessary features to support IPP printer hosting.

If your server hosts Web Services on Devices (WSD) printers

To enable WSD printing support:

1. Start the **Network and Sharing Center** from Control Panel and, click **Change advanced sharing settings** and click **Turn on network discovery**.
2. In Computer Management, Services, start the **Function Discovery Provider Host** service.
3. In Computer Management, Services, ensure the **Device Association Service** is started.

The server will be then able to identify and communicate with WSD-enabled printers.

If your print server is a highly available virtual machine

- To create a highly available print environment using Hyper-V and Failover Clustering, see High Availability Printing Overview. For more information about Windows Server 2012 Failover Clustering, see What's New in Failover Clustering in Windows Server 2012 and Clustering and High-Availability.
- Continue with the restoration process on the Printer Server virtual machine on the primary node.

If your server hosts local bus printers (LPT and USB)

The migration of local bus printers (LPT and USB) is not supported, although these printers are shown during backup. After the migration is complete:

1. Share the local bus printers again on the destination server.
2. Verify that each printer's name has not changed.
3. Test the printers to ensure that the shared connections still work.

If your server hosts plug and play printers

The migration of plug-and-play printers is not supported. To migrate plug and play printers:

1. Plug the printer into the destination server. The plug-and-play printer drivers will be installed automatically.
2. Enable printer sharing for the print queues.

See Also

Migrate Print and Document Services to Windows Server 2012

Preparing to Migrate

Migrating the Print and Document Services Role

Verifying the Migration

Post-Migration Tasks

Appendix A - Printbrm.exe Command-Line Tool Details

Appendix C - Printbrm Event IDs

Appendix C - Printbrm Event IDs

Printbrm Event IDs

The following Printbrm events are logged in the **Applications and Services Logs/Microsoft/PrintBRM/Admin** event log:

Event ID	Description
1	Printbrm.exe (the Printer Migration Wizard or the command-line tool) is beginning a backup of print queues. No user action is required.
2	Printbrm.exe (the Printer Migration Wizard or the command-line tool) is beginning a restore of print queues. No user action is required.
3	Printbrm.exe (the Printer Migration Wizard or the command-line tool) replaced driver map settings %1 with %2 for queue %3. No user action is required.
4	Printer queue %1 will be restored without a separator page. Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to create a separator file for this queue. Error: %2.
5	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not find a configuration file and is using the default settings. No user action is required.

Event ID	Description
6	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully restored print queue %1. No user action is required.
7	Printbrm.exe (the Printer Migration Wizard or the command-line tool) restored queue %1, but failed to restore printer driver settings. Error %2. This can occur if the driver on the destination server is newer than the driver in the migration file. Open the printer Properties dialog box on the destination computer and manually specify the appropriate printer settings.
8	While attempting to publish the printer to the Active Directory directory service, Printbrm.exe (the Printer Migration Wizard or the command-line tool) may have failed to adjust publishing settings for %1. This can occur if Printbrm.exe cannot access Active Directory. Manually publish the printer using the printer Properties dialog box.
9	Printer queue %1 already exists on the destination computer. Printbrm.exe (the Printer Migration Wizard or the command-line tool) will update the printer settings to match the settings on the source computer. No user action is required.
10	Printer queue %1 already exists on the destination computer and will not be changed because Printbrm.exe (the Printer Migration Wizard or the command-line tool) was run in 'Keep Existing' mode. No user action is required.
11	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore print queue %1 because port %2 is unknown. Printbrm.exe will attempt to restore the print queue on port FILE: instead. This can occur if the backup file contains incomplete data about the port, or if the port or port settings are

Event ID	Description
	incompatible with the version of Windows installed on the destination computer. Recreate the affected port on the destination computer and then change the print queue to use the new port.
12	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully installed printer queue %1 on port FILE:. No user action is required.
13	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore printer queue %1 on port FILE:. Error %2. This can occur if the backup file contains incomplete data about the port, or if the port or port settings are incompatible with the version of Windows installed on the destination computer. Recreate the affected port on the destination computer.
14	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully restored %1. No user action is required.
15	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore print processor %1 while restoring print queues from a file. Error: %2. Examine the Windows error returned by this event to determine the cause.
16	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not back up a dependent file for a language monitor. Dependent file: %1. Error: %2. This can occur if the file was deleted or moved.
17	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully restored language monitor %1. No user action is required.
18	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not restore language monitor %1 while restoring print queues from a file. Check the print queue

Event ID	Description
	backup file and examine the Windows error returned by this event to determine the cause.
19	The language monitors in the backup file are for a different processor architecture than the destination computer. Printbrm.exe (the Printer Migration Wizard or the command-line tool) will not migrate any language monitors. Source architecture: %1. Destination architecture: %2.
20	Printbrm.exe (the Printer Migration Wizard or the command-line tool) restored a driver for a different processor architecture than that of the destination computer. Printbrm.exe will attempt to locate and install a native version of driver %1 on destination %2. Try to print to the print queue, and if necessary, manually install a native version of the driver.
21	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully installed driver %1 for the processor architecture of the destination computer. No user action is required.
22	The driver in the backup file is for a different processor architecture than the destination computer, and Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not locate and install a native version of the driver. Driver: %1. Destination architecture: %2. Error: %3. Install a native version of the driver on the destination computer and then retry importing the print queues.
23	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully installed driver %1 (%2) from a cabinet (.cab) file. No user action is required.
24	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not restore driver %1 (%2) because it was backed up on the source computer without its binary files. Printbrm.exe could not install the driver from the

Event ID	Description
	local driver cabinet (.cab) file on the destination computer. Error: %3. This can occur if the user did not save the driver binary files while backing up the print queue, or when restoring a print queue on a destination computer that uses a different processor architecture than the source computer. Install the driver manually on the destination computer or back up the print queue with its binary files. Then, retry importing the print queues.
25	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully installed driver %1 (%2) from files. No user action is required.
26	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not restore driver %1 (%2) from files. Error reported: %3. This can occur if the driver requires a file that Printbrm.exe did not back up or if the user does not have permission to install drivers on the destination computer.
27	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to remove driver temporary folder %1. Error %2. Manually delete the temporary files and folder.
28	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to copy %1 to %2. Error %3 occurred while it was restoring print queues from a file. This can occur if the user does not have proper permissions on the destination computer, if the backup file is corrupted, or if the system is unstable. Retry exporting the printer queues on the source computer.
29	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not convert Line Printer Remote (LPR) port %1 to a standard TCP/IP printer port. Cannot get device settings. Error %2. This can occur if the LPR port was

Event ID	Description
	incorrectly configured. Manually create a standard TCP/IP printer port for the printer.
30	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore the Line Printer Remote (LPR) printer port %1 because the length of the port name is too long. len >= MAX_PORTNAME_LEN: Error %2. This can occur if the backup file contains incomplete data about the port, or if the port or port settings are incompatible with the version of Windows installed on the destination computer. Recreate the affected port on the destination computer.
31	Port %1 already exists. Printbrm.exe (the Printer Migration Wizard or the command-line tool) will skip restoring this port. No user action is required.
32	Printbrm.exe (the Printer Migration Wizard or the command-line tool) port install status: %1. No user action is required.
33	Printbrm.exe (the Printer Migration Wizard or the command-line tool) is backing up printer forms. No user action is required.
34	Printbrm.exe (the Printer Migration Wizard or the command-line tool) is restoring printer forms. No user action is required.
35	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully saved %1 user forms. No user action is required.
36	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not find any user forms to restore. No user action is required.
37	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not restore one or more printer forms from the backup file. This can occur if the forms already exist on the destination computer or if the user does not have permissions to create forms on the destination computer.

Event ID	Description
38	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not load .xml file %1 while restoring print queues from a file. Error: %2. This can occur if the user does not have proper permissions on the destination computer, if the backup file is corrupted, or if the system is unstable. Retry exporting the printer queues on the source computer.
39	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not find the printer driver migration plug-in DLL %1 for print processor %2. Printbrm.exe will skip this file and attempt to migrate the driver, but the printer might not work. This can occur due to a problem with the migration plug-in provided by the printer driver. Try to print to the affected printer and install a newer version of the driver, if necessary.
40	Printbrm.exe (the Printer Migration Wizard or the command-line tool) changed the spooler folder to: %1. No user action is required.
41	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully restored the print queue, but could not change spooler folder to %1. This path does not exist, so Printbrm.exe used the default spooler folder location (%WINDIR%\System32\Spool\Printers). This can occur if the configuration of the source computer is different from the destination computer. No user action is required unless you want to use a custom spooler folder location.
42	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully restored the print queue, but could not change the location of the spooler folder. Error %1. Printbrm.exe used the default spooler folder location (%WINDIR%\System32\Spool\Printers). This

Event ID	Description
	can occur if the configuration of the source computer is different from the destination computer. No user action is required unless you want to use a custom spooler folder location.
43	Printbrm.exe (the Printer Migration Wizard or the command-line tool) changed the spooler log level to: %1. No user action is required.
44	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully restored the print queue, but could not change the spooler log level. Error reported: %1. No user action is required unless you want to change the default spooler log level.
45	Printbrm.exe (the Printer Migration Wizard or the command-line tool) is backing up printer objects on server: %1. No user action is required.
46	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to place a file in cabinet (CAB) file %1 while backing up print queues. Error reported: %2. This can occur if the user does not have permission to create a file in the destination location, or if there is insufficient disk space or system resources.
47	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to create destination file %1 while backing up print queues. Error reported: %2. This can occur if the user does not have permission to create a file in the destination location, or if there is insufficient disk space or system resources.
48	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not load printer driver migration plug-in DLL %1. Error: %2. Printbrm.exe will attempt to migrate the driver, but the printer might not work because of missing files. This can occur due to a problem with the migration plug-in provided by the printer driver. Try to print to the affected printer

Event ID	Description
	and install a newer version of the driver, if necessary.
49	The printer driver migration plug-in %1 is incompatible with this version of Windows. Printbrm.exe (the Printer Migration Wizard or the command-line tool) will attempt to migrate the driver, but the printer might not work because of missing files. This can occur due to a problem with the migration plug-in provided by the printer driver. Try to print to the affected printer and install a newer version of the driver, if necessary.
50	The printer driver migration plug-in %1 ran with error: %2. Printbrm.exe (the Printer Migration Wizard or the command-line tool) will attempt to migrate the driver, but the printer might not work because of missing files. This can occur due to a problem with the migration plug-in provided by the printer driver. Try to print to the affected printer and install a newer version of the driver, if necessary.
51	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully ran printer driver migration plug-in %1. No user action is required.
52	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not restore print processor %1 because it is already installed on the destination computer. No user action is required.
53	While Printbrm.exe (the Printer Migration Wizard or the command-line tool) was restoring a print queue from backup, the following service control function %1 on computer %2 returned error %3. This can occur if Printbrm.exe cannot locate all files for a print processor, cannot stop the print spooler or the Cluster service, or if either of these services become unresponsive during installation of the print processor. If

Event ID	Description
	printing fails, manually install the print processor on the destination computer and examine the Print Spooler service and/or the Cluster service.
54	The service %1 is now running on server %2. No user action is required.
55	The service %1 is now stopped on server %2. No user action is required.
56	While restoring a print queue from backup, Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to stop service %1 on computer %2. Error %3. Try printing to the affected printer to determine if there is a problem.
57	While restoring a print queue from backup, Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to start service %1 on computer %2. Error %3. Try printing to the affected printer to determine if there is a problem.
58	Printbrm.exe (the Printer Migration Wizard or the command-line tool) did not overwrite print processor file %1 because the file on the destination system has an identical (or newer) time stamp. No user action is required.
59	Printbrm.exe (the Printer Migration Wizard or the command-line tool) copied print processor file %1 to %2. No user action is required.
60	While restoring a print queue from backup, Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to copy print processor file %1 to %2. Error %3. This can occur if Printbrm.exe cannot locate all files for a print queue, or if the Print Spooler service cannot be stopped during installation of the print processor. If printing fails, manually install the print processor on the destination computer.
61	The device %1 %2 supports more than one

Event ID	Description
	port. Printbrm.exe (the Printer Migration Wizard or the command-line tool) will convert the first port on the device from a Line Printer Remote (LPR) port to a standard TCP/IP port. The port on the device might need to be reconfigured.
62	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore the Line Printer Remote (LPR) port %1 because the LPR port monitor is not installed on the destination computer. Install the LPR port monitor on the destination computer.
63	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore the Line Printer Remote (LPR) port %1 while restoring print queues from a previously created backup file. Error code: %2. This can occur if the backup file contains incomplete data about the port, or if the port or port settings are incompatible with the version of Windows installed on the destination computer. Recreate the affected port on the destination computer.
64	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore the standard TCP/IP printer port %1. Error: %2. This can occur if the backup file contains incomplete data about the port, or if the port or port settings are incompatible with the version of Windows installed on the destination computer. Recreate the affected port on the destination computer.
65	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore printer driver settings (%1, %2). Error %3. This can occur if the driver on the destination server is newer than the driver in the migration file. Open the printer Properties dialog box on the destination computer and manually specify the appropriate printer settings.
66	Printbrm.exe (the Printer Migration Wizard or

Event ID	Description
	the command-line tool) cannot apply new settings for printer queue %1 because port %2 that is saved in the cabinet (.cab) file does not exist on the destination server. This can occur if Printbrm.exe failed to migrate the port. Manually install the appropriate port, and then specify the appropriate printer settings or retry importing the print queue.
67	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not open registry key %1. Error code %2. This can occur if an important Windows resource (such as the registry) is unavailable, if the Component Object Model (COM) cannot be initialized, or if Printbrm.exe cannot allocate memory. Examine the Windows error returned by this event to determine the cause.
68	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not create a cabinet (CAB) file at this location: %1. Error reported: %2. This can occur if the user does not have permission to create a file in the destination location, or if there is insufficient disk space or system resources.
69	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore the standard TCP/IP printer port %2 because the length of the host IP address %1 is too long to be restored on operating systems older than Windows Vista.
70	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to save separator file %1 because the file or path does not exist. This can occur when a separator page cannot be found during the backup process on the source computer.
71	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not access the print$ share on %1. Error reported: %2. This

Event ID	Description
	can occur if the destination computer does not have any shared printers.
72	Printbrm.exe (the Printer Migration Wizard or the command-line tool) was unable to access the Remote Registry service on %1. Error reported: %2. This can occur if the Remote Registry service is not started or if the computer is behind a firewall.
73	Printbrm.exe (the Printer Migration Wizard or the command-line tool) reset the availability information for the print queue because the StartTime or UntilTime values were invalid. Error %1. The printer will be always available until you use the Advanced tab of the printer's Properties dialog box to specify the correct availability.
74	Printbrm.exe (the Printer Migration Wizard or the command-line tool) successfully installed driver %1 (%2) from a driver package. No user action is required.
75	Printbrm.exe (the Printer Migration Wizard or the command-line tool) could not restore driver %1 (%2) from a driver package. Error: %3.
76	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to backup driver %1 (%2). The backup process will continue, skipping this driver. Error: %3.
77	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to backup port %1. The backup process will continue, skipping this port. Error: %2.
78	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to back up print processor %1 (%2). Error: %3.
79	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to back up print processor files for architecture %1. Error: %2.

Event ID	Description
80	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to back up print queue %1. The backup process will continue, skipping this queue. Error: %2.
81	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore print queue %1. The restore process will continue, skipping this queue. Error: %2.
82	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to create a new separator file folder while restoring print queue %1. This can occur if the user does not have permission to create a file in the destination location, or if there is insufficient disk space or system resources. The print queue was not restored successfully. Error: %2.
83	Printbrm.exe (the Printer Migration Wizard or the command-line tool) failed to restore the WSD printer port described by the remote URL or global identifier %1. Error: %2. This can occur if the backup file contains incomplete data about the port, or if the port or port settings are incompatible with the version of Windows installed on the destination computer. Recreate the affected port on the destination computer.

See Also

Migrate Print and Document Services to Windows Server 2012

Migrating the Print and Document Services Role

Verifying the Migration

Post-Migration Tasks

Appendix A - Printbrm.exe Command-Line Tool Details

Appendix B - Additional Destination Server Scenarios

Migrate Remote Access to Windows Server 2012

Routing and Remote Access Service (RRAS) was a role service in Windows Server operating systems prior to Windows Server 2012 that enabled you to use a computer as an IPv4 or IPv6 router, as an IPv4 network address translation (NAT) router, or as a remote access server that hosted dial-up or virtual private network (VPN) connections from remote clients. Now, that feature has been combined with DirectAccess to make up the Remote Access server role in Windows Server 2012 This guide describes how to migrate a server that is hosting the Routing and Remote Access service (in Windows Server 2008 R2 and other down-level versions) to a computer that is running Windows Server 2012.

Note

Your detailed feedback is very important, and helps us to make Windows Server Migration Guides as reliable, complete, and easy to use as possible. Please take a moment to rate this topic, and then add comments that support your rating. Describe what you liked, did not like, or want to see in future versions of the topic. To submit additional suggestions about how to improve Migration guides or utilities, post on the Windows Server Migration forum.

About this guide

Migration documentation and tools ease the migration of server role settings and data from an existing server to a destination server that is running Windows Server 2012. By using the tools that are described in this guide, you can simplify the migration process, reduce migration time, increase the accuracy of the migration process, and help to eliminate possible conflicts that might otherwise occur during the migration process. For more information about installing and using the migration tools on the source and destination servers, see Install, Use, and Remove Windows Server Migration Tools.

Target audience

This document is intended for information technology (IT) administrators, IT professionals, and other knowledge workers who are responsible for the operation and deployment of the Remote Access servers in a managed environment. Some scripting knowledge may be required to perform some of the migration steps that are contained in this guide.

What this guide does not provide

This guide does not describe the architecture or detailed functionality of the Remote Access role. The following scenarios are not supported in this migration guide:

- Any process for an in-place upgrade, in which the new operating system is installed on the existing server hardware by using the **Upgrade** option during setup

- Clustering and multisite scenarios
- Migrating more than one server role

  If your server is running multiple roles, it is recommended that you design a custom migration procedure that is specific to your server environment and based on the information that is provided in this and other role migration guides.

Supported migration scenarios

This guide provides instructions for migrating an existing server to a server that is running Windows Server 2012.

Caution

This guide does not contain instructions for migration when the source server is running multiple roles. If your source server is running multiple roles, some migration steps in this guide, such as those for migrating user accounts and network interface names, can cause other roles that are running on the source server to fail.

Supported operating systems

This guide provides instructions for migrating data and settings from an existing server that is being replaced by a new physical or virtual 64-bit server with a clean-installed operating system, as described in the following table.

Source server processor	Source server operating system	Destination server operating system	Destination server processor
x86- or x64-based	Windows Server 2003 with Service Pack 2	win8_Server_2	x64-based
x86- or x64-based	Windows Server 2003 R2	win8_Server_2	x64-based
x86- or x64-based	Windows Server 2008, full installation option only	win8_Server_2	x64-based
x64-based	Windows Server 2008 R2, full installation option only	win8_Server_2	x64-based
x64-based	Win8_Server_2	win8_Server_2	x64-based

- The versions of operating systems shown in the preceding table are the earliest combinations of operating systems and service packs that are supported. Newer service packs are supported.
- Migration with the destination server already having DirectAccess configured is not supported.
- The Foundation, Standard, Enterprise, and Datacenter editions of the Windows Server operating system are supported as either source or destination servers. This includes

migrating across editions. For example, you can migrate from a server running Windows Server 2003 Standard to a server running Windows Server 2012.

- Migrations between physical operating systems and virtual operating systems are supported.
- Migration from a source server to a destination server that is running an operating system in a different system UI language (that is, the installed language) than the source server is not supported. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shared resources from a computer that is running Windows Server 2008 R2 in the French system UI language to a computer that is running Windows Server 2012 in the German system UI language.

  Note

  The system UI language is the language of the localized installation package that was used to set up the Windows operating system.
- Both x86- and x64-based migrations are supported for Windows Server 2003 and Windows Server 2008. All editions of Windows Server 2008 R2 and win8_server_2 are x64-based.

Supported role configurations

The following is a broad list of the migration scenarios that are supported for Remote Access. All settings under these scenarios are migrated.

- **DirectAccess (supported in Windows Server 2012 to Windows Server 2012 migration only)**
- **VPN server**
- **Dial-up server**
- **Network address translation (NAT)**
- **Routing**, with the following optional components:
  - DHCP Relay Agent
  - Routing Information Protocol (RIP)
  - Internet Group Management Protocol (IGMP)

In addition to the above scenarios, migration also automatically adjusts configuration of the destination server to account for features that are no longer supported and to support features that are new in Windows Server 2012 and not supported on earlier versions of Windows.

Migration dependencies

If a local or remote NPS server that is used for authentication, accounting, or policy management must also be migrated, then migrate the NPS service before migrating Remote Access. For more information, see Migrate Network Policy Server to Windows Server 2012.

If you are upgrading from Windows 2008 R2 DirectAccess to Windows Server 2012, ensure that all the DirectAccess configuration settings have been applied on the Windows 2008 R2 server. It is possible to save settings through the console but not apply them. Before upgrading, ensure that the saved settings have also been applied.

Migration components that are not supported in all operating system versions

The following Remote Access components are not supported by all operating systems:

Component	UI Dialog/Settings	Action
New Components, not available on Windows Server 2003, Windows Server 2008, and Windows Server 2008 R2		
Specifying adapter to obtain DNS/WINS addresses	RAS Properties – IPv4 Tab: Adapter	This component is not supported on Windows Server 2003. The default value should be used for this setting on the target computer.
SSTP	SSTP ports	SSTP is not supported on Windows Server 2003. SSTP ports should be enabled on the target computer. The number depends on the default value for the SKU on the target computer
IPv6	1. RAS Properties – General Tab: IPv6 router check-box and corresponding radio-buttons, IPv6 Remote access server 1. RAS Properties – IPv6 Tab: All settings 1. Demand-Dial (VPN/PPPoE) properties – Networking tab – TCP/IPv6 1. Demand-dial (VPN/PPPoE) – IPv6 filters for connection initiation 1. IPv6 - Router	• IPv6 is not supported on Windows Server 2003. It should be disabled on the target computer if DirectAccess is not deployed (legacy VPN). Under the General tab of RRAS properties, IPv6 Router and IPv6 Remote access server should not be selected. • The adapter setting under the IPv6 tab of RAS properties was introduced in Windows Server 2008 R2 for IKEv2. It was not present in Windows Server 2008. During migration this setting should be set to the

		default value on the target computer for Windows Server 2008 and be “as is” for Windows Server 2008 R2 and Windows Server 2012
IP Filters under RA Logging and Policies	Remote Access Logging & Policies – IP Filters	Windows Server 2003 and Windows Server 2008 do not have an option to create IP filters under Remote Access Logging and Policies. Hence there would be no filters to migrate.
Automatically obtaining IPv6 address	Demand-Dial (VPN/PPPoE) properties - Networking tab - IPv6 Properties – ‘Obtain IP address automatically’ radio-button	This setting is not present in Windows Server 2008. The value of this setting should be set to default on the target computer.
IKEv2	1. IKEv2 ports 1. RAS Properties – Security Tab: computer certificate authentication for IKEv2 1. RAS Properties – IKEv2 Tab: All settings	• IKEv2 is not supported on Windows Server 2003 and Windows Server 2008. IKEv2 ports should be enabled on the target computer. The number would depend on the default value for the SKU on the target computer. • Default values should be used for all IKEv2 settings on the target computer.
SSTP Cert. Selection	RAS Properties – Security Tab: ‘Use HTTP’ check-box, drop-down to select certificate, crypto binding settings	This component is not supported on Windows Server 2003 and Windows Server 2008. Default values should be used for all these settings on the target computer.
VPN Accounting	Remote Access Logging & Policies – Accounting: Logging failure action settings under ‘SQL Server Logging Properties’ and ‘Log File	These are not present in Windows Server 2008. The default value should be used on the target computer.

	Properties'	
Deprecated Features: Not available on Windows Server 2008, Windows Server 2008 R2, or Win8_Server_2		
SPAP, MS-CHAP, EAP-MD5 protocols and related settings	VPN/PPPoE Demand-dial interface properties - Security tab	SPAP, EAP-MD5 and MS-CHAP settings are not supported on Windows Server 2008 R2, or win8_Server_2, and will not be migrated.
Local Area Connection interface configuration under Routing	Routing – General – Local Area Connection properties – Configuration tab	This tab provides settings to configure how an IP address should be obtained for this interface. It is only present on Windows Server 2003 and will not be migrated.
RAS Firewall (integrated with NAT)	1. NAT – Interface – NAT/Basic Firewall Tab 1. NAT – Interface – ICMP Tab	Windows Server 2003 supported RAS firewall functionality which was removed in Windows Server 2008. These settings will not be migrated.
Weak Encryption settings		Weak encryption is supported on Windows Server 2003 but on Windows Server 2008 and Windows Server 2008 R2, it can only be enabled through the registry. During migration from Windows Server 2003 the registry settings will not be created automatically. For Windows Server 2008 and higher versions, if these registry settings happen to be present already, they will be migrated.

Migration components that are not automatically migrated

The following Remote Access elements and settings are not migrated by the Windows PowerShell cmdlets that are supplied with the Windows Server Migration Tools. Instead, you must manually configure the element or setting on the new RRAS server as described in **Completing the required manual migration steps** in this guide.

Important
Perform the manual configuration of these elements only when directed later in this guide.

- **SSL certificate bindings**. SSL Certificate binding and crypto-binding settings for SSTP are migrated as follows:
  a. The migration Wizard looks for a source certificate on the destination computer. If one is found, SSTP uses that certificate.
  b. If a source certificate is not found, the migration wizard will look for a valid certificate with the same trusted root as the source certificate.
  c. If still no certificate is found, then the SSTP configuration on the destination computer is 'Default'.
  d. If self-signed certificates are being used (valid for win8_Server_2), they will be automatically created on the destination computer.
- **User accounts on the local RRAS server**. If you use domain-based user and group accounts, and both the old and new RRAS servers are part of the same domain, no migration of the accounts is required. Local user accounts can be used if **Windows Authentication** is configured on the RRAS source server.
- **Only routing/VPN/DirectAccess when all are installed**. If your Remote Access server configuration includes all of the available services, then all services must be migrated together. Migrating only one of the services to the destination server is not supported.
- **A local or remote server that is running Network Policy Server (NPS)** that provides authentication, accounting, and policy management. This guide does not include the steps that are required to migrate a server that is running NPS. To migrate a server that is running NPS, use Migrate Network Policy Server to Windows Server 2012. NPS migration should be performed when directed later in this guide.

  Note
  If you are not using a server that is running NPS, the default Remote Access policies and accounting settings that are automatically created while configuring RRAS are not migrated.
- **Dial-up based demand-dial connections**. The destination server might have different modems, and there are many demand-dial settings that are specific to the modem or ISDN device that is selected.
- **Certificates** used for authenticating IKEv2, SSTP, and L2TP/IPsec connections.
- **SSL Certificate Binding** for SSTP when the **Use HTTP** check box is not selected.

- **IKEv2 VPN connections that use IPv6 network adapters**. IKEv2 is supported on RRAS servers that are running Windows Server 2008 R2 only. In the RRAS Microsoft Management Console (MMC) on Windows Server 2008 R2, you can specify the network interface used to acquire IPV6 DHCP and DNS addresses that are used for IKEv2 VPN clients. If you migrate RRAS from Windows Server 2008 R2 to another server running Windows Server 2008 R2, the setting is migrated. However, if you migrate from a previous version of Windows, there is no setting to migrate, and the default value of **Allow RAS to select the adapter** is used.
- **Weak encryption**. In Windows Server 2003, weak encryption is enabled, but on later versions of Windows it is disabled by default. You can enable weak encryption only by modifying the registry. During migration from Windows Server 2003 the required registry settings are not created on the new server by the migration process, and they must manually be configured. For later versions of Windows, if these registry settings are present, they are migrated.
- **Admin DLLs and Security DLLs** and their corresponding registry keys. These DLLs are available in both 32-bit and 64-bit versions, and they do not work in a 32-bit to 64-bit migration.
- **Custom DLLs used for dialing a demand-dial connection**. These DLLs are available in both 32-bit and 64-bit versions, and they do not work in a 32-bit to 64-bit migration. Any corresponding registry settings also are not migrated.
- **Connection Manager profiles**. The Connection Manager Administration Kit is used to create VPN and dial-up remote access profiles. Profiles created are stored under specific folders on the RRAS server. Profiles that are created on a 32-bit version of Windows do not work on computers that are running a 64-bit version of Windows, and vice versa. For more information about connection profiles, see Connection Manager Administration Kit (http://go.microsoft.com/fwlink/?linkid=55986).
- **The Group Forwarded Fragments setting on NAT**. This setting is enabled if the RRAS server is deployed behind a NAT router running on the Windows operating system. This is required for L2TP/IPsec connections that are using computer certificate authentication to succeed. We recommend that you enable this value to work around a known issue in RRAS.
- The **Log additional Routing and Remote Access information (used for debugging)** setting in the **Routing and Remote Access Properties** dialog box on the **Logging** tab.

Overview of the Routing and Remote Access service migration process

The pre-migration process involves the manual collection of data, followed by running procedures on the destination and source servers. The migration process includes source and destination server procedures that use the **Export** and **Import** cmdlets to automatically collect, store, and migrate server role settings. Post-migration procedures include verifying that the destination server successfully replaced the source server and then retiring or repurposing the source server. If the verification procedure indicates that the migration failed, troubleshooting begins. If troubleshooting fails, rollback instructions are provided to return the network to the use of the original source server.

Impact of migration

During migration, the Remote Access server is not available to accept incoming connections or to route traffic.

- New remote clients cannot connect to the server by using dial-up, VPN or DirectAccess connections. Existing connections on the server are disconnected. If you have multiple remote access servers, then the loss of availability of this server results in a reduction in capacity until the new server is operational again. For demand-dial connections, you must provide alternate connectivity between offices or reconfigure the connections to point to an alternate server.
- Routing and NAT functionalities are not available. If the functionality is required during the migration, an alternate router can be deployed until the new destination server is available.

Post-migration impacts include the following:

- If you plan to reuse the name of the source server as the name of the destination server, the name can be configured on the destination server after the source server is disconnected from the network. Otherwise, there is a name conflict that can affect availability. If you plan to run both servers, then the destination server must be given a unique name.
- A VPN server can be directly connected to the Internet, or it can be placed on a perimeter network that is behind a firewall or NAT router. If the IP address or DNS name of the destination server changes as part of the migration, or after the migration is completed, then the mappings in the firewall or NAT device must be reconfigured to point to the correct address or name. You must also update any intranet or Internet DNS servers with the new name and IP address. Also, remember to provide information about any server name or IP address changes to your users so that they can connect to the correct server. If you use connection profiles that are created by using the Connection Manager Administration Kit, then deploy a new profile with the updated server address information.

Note

For DirectAccess, the source computer and the destination computer must have the same IP addresses and interface names.

We recommend that you advertise the expected date and time of the migration so that users can plan accordingly, and make other arrangements as needed.

Permissions required to complete migration

The following permissions are required on the source server and the destination Remote Access servers:

- Domain user rights are required join the new server to the domain.
- Local administrative rights are required to install and manage the Remote Access role.
- Equivalent administrator permissions for DirectAccess GPOs as were configured on the source computer.
- Write permissions are required to the migration store location. For more information, see Remote Access: Prepare to Migrate in this guide.

Estimated duration

The migration can take two to three hours, including testing.

See Also

Remote Access: Prepare to Migrate

Remote Access: Migrate Remote Access

Remote Access: Verify the Migration

Remote Access: Post-migration Tasks

Remote Access: Prepare to Migrate

Perform the following steps before you begin migrating Remote Access from your x86-based or x64-based source server to an x64-based destination server that is running Windows Server® 2012.

- Prepare your destination server
- Prepare your source server
- Install the migration tools

Membership in the local **Administrators** group or equivalent is the minimum required to complete these procedures. If User Account Control (UAC) is enabled, you might have to run the following steps by using the **Run as administrator** option.

Prepare your destination server

Complete the following procedures to prepare the destination server for the migration of Remote Access.

Hardware requirements for the destination server

Your destination server should have the same number or more network adapters as your source server. You can have more network adapters on the destination server than the source server, but the migration fails if you have fewer.

Important

The names of the network adapters on the destination server must be the same as those on the source server, and they must have the same intention (for example, Internet facing versus intranet facing). Most Remote Access server components have interface-specific settings and configuration. Having the same number of interfaces, with the same names and intent, helps ensure that the settings are migrated to the right interface. This is critical to a successful migration. If there are more adapters on the destination server than on the source server, you must still have a one-to-one match between the names and

intention of the network adapters on the source server and those on the destination server.

Note

DirectAccess configuration can only be migrated from a computer running Windows Server 2012 to another computer running Windows 2012. Migration from Windows Server 2008 R2 DirectAccess to Windows Server 2012 DirectAccess is not supported.

Prepare the destination server for migration

To prepare the destination server

1. Install Windows Server 2012 on the destination server.
2. Whether or not you intend to migrate the source server name to the destination server, give the destination server a temporary computer name at this time.
3. If you store the user accounts for remote access users locally on the Remote Access server instead of in Active Directory, and if you use the Challenge Handshake Authentication Protocol (CHAP) for authentication, then you must perform the following additional steps before migrating Remote Access:
    a. To enable the use of CHAP authentication, you must manually configure a local security policy setting that enables passwords to be stored by using a reversible encryption algorithm.

        Security

        We recommend that you do not use CHAP for authentication, and that you do not enable the setting to store passwords with reversible encryption. These options are not considered secure, and they are provided only for backwards compatibility. Use them only if your environment requires the use of CHAP.

        i. On the destination server, in the **Start** screen, click **Administrative tools**, and then click **Local Security Policy**.
        ii. In the navigation tree, expand **Account Policies**, and then select **Password Policy**.
        iii. In the details pane, double-click **Store passwords using reversible encryption**, click **Enabled**, and then click **OK**.

    b. Migrate the local users and groups from the source server to the destination server. Do this separately and before you begin migrating Remote Access.
4. If the source server that is being replaced is joined to a domain, join the destination server to the same domain.
5. In the dashboard of the Server Manager console click **Add roles and Features**.
6. Click **Next** until you reach the **Select Server Roles** screen.
7. On the **Select Server Roles** screen, select **Remote Access**. Click **Add Required Features**, and then click **Next**.

8. On the **Select features** screen, click **Windows Server Migration Tools**. Click **Next** until you reach the **Select Role Services** screen.
9. In the **Select Role Services** screen, select **Routing** and then click **Next**.
10. On the **Confirm Installation selections** screen, click **Install**.
11. On the **Installation progress** screen, verify that the installation was successful, and then click **Close**.
12. Server roles were introduced in Windows Server 2008, and they are also used in Windows Server 2012. Remote Access is a role service that consists of the Routing service and the DirectAccess and VPN service. In Windows Server 2003, the Routing and VPN services were not separate. If the source server is running Windows Server 2003, ensure that both the Routing service and the DirectAccess and VPN service are installed on the destination server. If the source server is running Windows Server 2008, Windows Server 2008 R2 or Windows Server 2012, ensure that the destination server has the same Remote Access services installed as has the source server. If the source server has the routing service and the DirectAccess and VPN service, then you must install all these components on the destination server.
13. If DirectAccess is being migrated, the IP-HTTPS and Network Location certificate must be imported to the destination server. Note that if self-signed certificates were being used on the source server, this step is not required.
14. If DirectAccess is being migrated, run the Windows PowerShell cmdlet **install-remoteaccess –prerequisite** to ensure that the destination server meets all the requirements for DirectAccess.

The destination server is now prepared for migration.

Prepare your source server

Important

Before you begin migration, as a best practice, we recommend that you perform a backup of the source server. If the migration fails, and the recovery steps to restore the source server also fail, this backup can be critical for the quick restoration of service.

Back up your source server

- For information about backing up Windows Server 2003, see Backing up and restoring data in the Windows Server Technical Library (http://go.microsoft.com/fwlink/?linkid=163718).
- For information about backing up Windows Server 2008 or Windows Server 2008 R2, see Backup and Recovery in the Windows Server Technical Library (http://go.microsoft.com/fwlink/?linkid=163719).

Install the migration tools

Windows Server Migration Tools in Windows Server 2012 allows an administrator to migrate some server roles, features, operating system settings, shares, and other data from computers

that are running certain editions of Windows Server 2003, Windows Server 2008, or Windows Server 2008 R2 to computers that are running Windows Server 2012.

Install Windows Server Migration Tools on the source and the destination servers. Complete installation, configuration, and removal instructions for Windows Server Migration Tools are available in Install, Use, and Remove Windows Server Migration Tools.

Important

Before you run the **Import-SmigServerSetting**, **Export-SmigServerSetting**, or **Get-SmigServerFeature** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with domain users or groups who are members of local groups on the source server.

Before you run the **Send-SmigServerData** or **Receive-SmigServerData** cmdlets, verify that during migration, both source and destination servers can contact the domain controller that is associated with those domain users who have rights to files or shares that are being migrated.

See Also

Migrate Remote Access to Windows Server 2012

Remote Access: Migrate Remote Access

Remote Access: Verify the Migration

Remote Access: Post-migration Tasks

Remote Access: Migrate Remote Access

Complete the following procedures to migrate the Routing and Remote Access service from a source server to a destination server.

- Migrating Remote Access from the source server
- Migrating Remote Access to the destination server
- Completing the required manual migration steps

Membership in the local **Administrators** group or equivalent is the minimum required to complete these procedures. If User Account Control (UAC) is enabled, you might have to run the following steps by using the **Run as administrator** option. For more information, see Run a program with administrative credentials in the Windows Server TechCenter (http://go.microsoft.com/fwlink/?LinkId=131210).

Migrating Remote Access from the source server

Follow these steps to capture the configuration of Remote Access on the source server.

To capture Remote Access configuration: Windows Server 2003, Windows Server 2008, Windows Server 2008 R2

1. On the source server, open a Windows PowerShell session with elevated user rights.
2. Load Windows Server Migration Tools into your Windows PowerShell session.

   Note
   If you opened the current Windows PowerShell session by using the Windows Server Migration Tools shortcut on the **Start** menu, skip this step and go to the next step. You should only load the Windows Server Migration Tools snap-in in a Windows PowerShell session that was opened by using another method, and into which the snap-in has not already been loaded.

   To load Windows Server Migration Tools, run the following cmdlet:

   ```
   Add-PSSnapin Microsoft.Windows.ServerManager.Migration
   ```

3. Remote Access can be running on the source server while you are capturing its configuration. However, if you made configuration changes to Remote Access that require a service restart, then you must stop Remote Access before starting the migration. Use the following PowerShell command to stop the service:

   ```
   stop-service remoteaccess -force
   ```

   Note
   You must use the `-force` parameter because Remote Access has dependent services.

   To verify that the service is stopped, run the following command:

   ```
   get-service remoteaccess
   ```

4. On the source server, from Windows PowerShell, collect the settings from the source server by running the **Export-SmigServerSetting** cmdlet as an administrator. The following is the syntax for the cmdlet:

   ```
   Export-SmigServerSetting -featureID NPAS-RRAS -User All -
   Group -path StorePath -verbose
   ```

   The **Export-SmigServerSetting** cmdlet parameters can collect all Routing and Remote Access service settings on the source server in a single file (Svrmig.mig). Before you run this command, review the following:

   - When you run the **Export-SmigServerSetting** command, you are prompted to provide a password to encrypt the migration store data. You must provide this same password when you later import from the migration store. Make sure you provide a strong password to encrypt the migration data and that the location of the migration data file is secure.
   - The *StorePath* variable that is provided as the value of the **path** parameter can be an empty or nonempty folder. The actual data file that is placed in the folder (Svrmig.mig) is created by the **Export-SmigServerSetting** cmdlet. Do not specify a file name. If a migration data file already exists and you want to rerun the **Export-**

SmigServerSetting cmdlet, you must first move the Svrmig.mig file from that location and store it elsewhere, rename it, or delete it.

- If the path is not a shared location that the destination server can access, you must manually copy the migration store to the destination server or to a location that the destination server can access.
- Migrating users and groups can be combined with the cmdlets that are used to migrate Remote Access. The **-Users** and **-Group** parameters can be used in the **Export-SmigServerSetting** command to migrate the user and group accounts that are present locally on the Remote Access source server. If you are using an Active Directory domain or RADIUS for authentication, then these parameters are not needed.

  The **-Users** command supports the following parameters:

  - **All**. All user accounts on the source server are included in the migration output file.
  - **Enabled**. Only enabled user accounts on the source server are included in the migration output file.
  - **Disabled**. Only disabled user accounts on the source server are included in the migration output file.

  To prevent migrating any user accounts, do not include the **-Users** parameter in the command.

  The **-Group** command takes no additional parameters. If it is present, then all groups defined locally on the source server are included in the migration output file.

Note
The process described in the Local User and Group Migration Guide does not migrate some settings, including those under the **Dial-in** tab. We recommend that you thoroughly review the Local User and Group Migration Guide to understand which settings are migrated and which are not.

To capture the Routing and Remote Access service configuration: Windows Server 2012

1. From the Start screen, click **Windows Server Migration Tools**.
2.

Note
This step is only required if Routing/VPN is configured on the source computer.

Remote Access can be running on the source server while you are capturing its configuration. However, if you made configuration changes to Remote Access that require a service restart, then you must stop Remote Access before starting the migration. Use the following PowerShell command to stop the service:

```
stop-service remoteaccess -force
```

Note

You must use the `-force` parameter because Remote Access has dependent services.

To verify that the service is stopped, run the following command:

```
get-service remoteaccess
```

3. Before you start to capture the configuration, you must stop the Remote Access Management service. Use the following PowerShell command to stop the service:

   ```
   stop-service Ramgmtsvc
   ```

   To verify that the service is stopped, run the following command:

   ```
   get-service Ramgmtsvc
   ```

   Once the export and migration are complete, you can restart the Remote Access Management service:

   ```
   start-service Ramgmtsvc
   ```

4. On the source server, from Windows PowerShell, collect the settings from the source server by running the **Export-SmigServerSetting** cmdlet as an administrator. The following is the syntax for the cmdlet:

   ```
   Export-SmigServerSetting -featureID DirectAccess-VPN [-User
   All] -Group -path StorePath
   ```

   The **Export-SmigServerSetting** cmdlet parameters can collect all Remote Access settings on the source server in a single file (Svrmig.mig). Before you run this command, review the following:

   - When you run the **Export-SmigServerSetting** command, you are prompted to provide a password to encrypt the migration store data. You must provide this same password when you later import from the migration store. Make sure you provide a strong password to encrypt the migration data and that the location of the migration data file is secure.
   - The *StorePath* variable that is provided as the value of the **path** parameter can be an empty or nonempty folder. The actual data file that is placed in the folder (Svrmig.mig) is created by the **Export-SmigServerSetting** cmdlet. Do not specify a file name. If a migration data file already exists and you want to rerun the **Export-SmigServerSetting** cmdlet, you must first move the Svrmig.mig file from that location and store it elsewhere, rename it, or delete it.
   - If the path is not a shared location that the destination server can access, you must manually copy the migration store to the destination server or to a location that the destination server can access.
   - Migrating users and groups can be combined with the cmdlets that are used to migrate Remote Access. The **-Users** and **-Group** parameters can be used in the **Export-SmigServerSetting** command to migrate the user and group accounts that are present locally on the Remote Access source server. If you are using an Active Directory domain or RADIUS for authentication, then these parameters are not needed.

The **-Users** command supports the following parameters:

- **All**. All user accounts on the source server are included in the migration output file.
- **Enabled**. Only enabled user accounts on the source server are included in the migration output file.
- **Disabled**. Only disabled user accounts on the source server are included in the migration output file.

To prevent migrating any user accounts, do not include the **-Users** parameter in the command.

The **-Group** command takes no additional parameters. If it is present, then all groups defined locally on the source server are included in the migration output file.

Note

The process described in the Local User and Group Migration Guide does not migrate some settings, including those under the **Dial-in** tab. We recommend that you thoroughly review the Local User and Group Migration Guide to understand which settings are migrated and which are not.

Migrating Remote Access to the destination server

Return to the destination server, and use the following procedure to complete the migration:

To import the Routing and Remote Access service configuration to the destination server

1. Before you use the **Import-SmigServerSetting** cmdlet to import the Routing and Remote Access service settings, be aware of the following condition:
   - If you chose to migrate the users and groups on the source computer, you need to specify the **-User** and **-Group** parameters in the **Import-SmigServerSetting** cmdlet on the destination server.
2. On the destination Server that is running Windows Server 2012, from the Start screen, click **Windows Server Migration Tools**.
3. On the destination server, from Windows PowerShell, run the following command, where *StorePath* is the folder that contains the Svrmig.mig file that you exported from the source server. Do not include the name of the file in the path.

```
Import-SmigServerSetting -featureID DirectAccess-VPN [-User
All] -Group -path StorePath –Force
```

For more information about running the **Import-SmigServerSetting** cmdlet, see the “Using Windows Server Migration Tools” section in the Windows Server Migration Tools Install, Use, and Remove Windows Server Migration Tools guide.

4.

Note

This step is only required if Routing/VPN is configured on the source computer.

Before starting the Remote Access service, you must manually stop the RASMAN service. Run the following command in the Windows PowerShell Command Prompt window:

```
Stop-service -force rasman
```

5.

Note

This step is only required if Routing/VPN is configured on the source computer.

Then run the following command in the Windows PowerShell Command Prompt window to start the Routing and Remote Access service:

```
Start-Service RemoteAccess
```

If a failure occurs while running the **Import-SmigServerSetting** cmdlet, review the Setupact.log, Setuperr.log, and ServerMigration.log files under %localappdata%\SvrMig\Log. Information about how the Remote Access components migrated is included in the Servermigration.log file.

After the script completes, review the following section and adjust any remaining settings that require manual configuration.

Completing the required manual migration steps

Certain settings cannot be migrated by the Windows PowerShell scripts, and they must be configured manually on the destination server. Review the following configuration options, and apply those that are relevant to your environment.

DirectAccess

To ensure that the destination server meets all DirectAccess requirements, run the following Windows PowerShell cmdlet: **Install-remoteaccess –prerequisite**.

Dial-up demand-dial connections

Because of the differences in modem hardware that might exist between the source and destination servers, dial-up connections are not migrated. Use the Demand-Dial Interface Wizard in the Remote Access MMC snap-in.

To create a dial-up demand-dial connection

1. If you are using Server Manager, in Tools click Routing and Remote Access.
2. Right-click the server in the tree, and then click **Configure and Enable Routing and Remote Access**.

3. Follow the steps in the wizard to configure the connection.

Certificates for IKEv2, SSTP, and L2TP/IPsec connections

Certificates can be exported from the source server and imported to the destination server by using the Certificates MMC snap-in.

Routing and Remote Access service policies and accounting settings

If you are not using a local or remote server to run NPS, then default remote access policies and accounting settings are automatically created on the destination server when Remote Access is configured.

To migrate NPS settings, refer to Migrate Network Policy Server to Windows Server 2012.

PEAP, smart card, and other certificate settings on Network Policy Server

If you also migrated a local or remote server running NPS to support the Remote Access server that you are migrating, we recommend that you verify that the server that is running NPS has the correct certificate configuration. Specifically, confirm that any certificates that are associated with Protected Extensible Authentication Protocol (PEAP) and the **Smart card or other certificate** authentication settings are set properly. You can find these settings on the server that is running NPS, in the NPS MMC snap-in under **Connection Request Policies** or **Network Policies** (depending on where the authentication protocols are configured). If no certificates are present, or if the certificates are not configured correctly, perform the following steps:

To reconfigure PEAP or smart card certificates

1. Remove the **PEAP** or **SmartCard or other certificate** methods from the list of authentication methods.
2. Add the method back to the list.
3. Reconfigure the certificate for the specified method.

Weak encryption settings

In Windows Server 2003 weak encryption is enabled, but on later versions of Windows it is disabled by default. You can enable weak encryption only by modifying the registry. During migration from Windows Server 2003, the required registry settings are not created on the new server by the migration process, and they must manually be configured. For later versions of Windows, if these registry settings are present, they are migrated. For more information about the registry entries that Remote Access adds, see “Registry entries that Routing and Remote Access adds in Windows Server 2008”, article 947054 in the Microsoft Knowledge Base (http://go.microsoft.com/fwlink/?linkid=159112). The description of the settings for the weak

encryption settings are at the end of the article, and they are named **AllowPPTPWeakCrypto** and **AllowL2TPWeakCrypto**.

Security

Weak encryption includes the use of 40-bit or 56-bit encryption in PPTP, and the use of MD5 or DES for L2TP/IPsec. By default, these weak algorithms are disabled, and we recommend that you do not use them unless they are required.

Connection Manager profiles

Profiles that are created by the Connection Manager Administration Kit (CMAK) can only be created on a computer with the same 32-bit or 64-bit architecture as the client computer on which they are to be run. If your source server is 64-bit, and you have created 64-bit profiles on that source server, you can copy them from the **%PROGRAMFILES%\CMAK\Profiles** folder to the appropriate folder on the destination server.

If the source server is 32-bit, you must use a computer running a 32-bit version of Windows to create and manage the profiles. You can set up a computer running a 32-bit version of Windows 7 or Windows 8, and then install CMAK on it to manage the profiles for your 32-bit client computers. For more information, see Connection Manager Administration Kit in the Windows Server Technical Library (http://go.microsoft.com/fwlink/?linkid=136440).

Group forwarded fragments

The **Group Forwarded Fragments** setting on NAT is enabled if the Remote Access server is deployed behind a NAT device that runs Windows. This is required for L2TP/IPsec connections that are using computer certificate authentication to succeed. We recommend that you enable this setting. Group Forwarded Fragments can be enabled for IPv4 on the Windows NAT computer by running the following command at the command prompt:

```
netsh int ipv4 set global groupforwardfragments=enabled
```

RAS administration and security DLLs

Administration DLLs and security DLLs and their corresponding registry keys are not migrated. This is because they are available in both 32-bit and 64-bit versions, and they do not work in a 32-bit to 64-bit migration. If the source and destination computers are 64-bit, the administration and security DLLs can be reused. For more information, refer to the following topics:

RAS Administration DLL (http://go.microsoft.com/fwlink/?linkid=163778)

RAS Security DLL (http://go.microsoft.com/fwlink/?linkid=163779)

See Also

Migrate Remote Access to Windows Server 2012

Remote Access: Prepare to Migrate

Remote Access: Verify the Migration
Remote Access: Post-migration Tasks

Remote Access: Verify the Migration

After all the migration steps are completed, you can use the following procedure to verify that the migration of Remote Access was successful. If the migration failed, you can return to the previous valid configuration by following the roll-back steps in Remote Access: Post-migration Tasks.

Verifying the destination server configuration

Membership in the local **Administrators** group or equivalent is the minimum required to complete these procedures. If User Account Control (UAC) is enabled, then you might have to run the following steps by using the **Run as administrator** option.

We recommend that you check the configuration of the destination Remote Access server, from the service start-up to the detailed configuration of individual components. The following sections provide a list of items to check. Depending on which Remote Access components are enabled on your server, only some of these checks might be necessary.

Installation state of Remote Access

The first verification step is to confirm that the Remote Access feature installed successfully.

To verify that Remote Access installed on the destination server

1. Click **Windows Server Migration Tools** on the **Start** screen.
2. View the installation status of the Routing and Remote Access service by running the following command:

   ```
   Get-WindowsFeature RemoteAccess
   ```

   The check box on the left of the **Remote Access** feature name is selected if the service is installed on the destination server. If it is not installed, the check box is clear.

Status of Remote Access Service

Verify that the Remote Access service is running.

To verify that the Routing and Remote Access service is running on the destination server

1. Click **Windows Server Migration Tools** on the **Start** screen.
2. View the service status of the Routing and Remote Access service by running the following command:

```
Get-service RemoteAccess
```

3. Examine the **Status** column. It should read **Running**.

Remote access Operations Status

Verify the operations status of the deployment.

To verify the Remote Access operations status

1. In **Server Manager** click **Tools** and then click **Remote Access Management**.
2. Click **OPERATIONS STATUS** to navigate to **Operations Status** in the **Remote Access Management Console**. **Operations Status** lists the server operational status and that of all its components.

DirectAccess configuration

Verify the operations status of the deployment.

To verify the DirectAccess configuration settings

1. In **Server Manager** click **Tools** and then click **Remote Access Management**.
2. Click **CONFIGURATION** to navigate to Configuration tab in **Remote Access Management console**. Step through each of the wizards to ensure that the configuration has been migrated successfully.

VPN configuration

Confirm the configuration settings for the Remote Access server and ports.

To verify the Remote Access configuration settings

1. Start Server Manager.
2. In **Tools**, click **Routing and Remote Access**.
3. Right-click the Remote Access server node, and then click **Properties**.

   On each tab, confirm that the destination server is configured the same as the source server, and then click **OK**.
4. In the navigation pane, select **Ports**.

   Confirm that any modem or ISDN devices that are attached to the computer are included in the list.
5. In the navigation pane, right-click **Remote Access Logging and Policies**, and then click **Launch NPS**. In the Network Policy Server navigation pane, select **Network Policies**.

   Confirm that the NPS policies that are currently configured are those required for your environment. If you migrated them from an NPS source server to an NPS destination server, confirm that you are connected to the destination server and that the policies

migrated successfully.

Dial-up configuration

You must confirm that the correct phone lines are attached to the modems or ISDN ports on the destination server.

Demand-dial VPN configuration

Examine all of your demand-dial VPN connections to ensure that they migrated with the correct settings.

To verify the settings for a demand-dial VPN connection

1. Start Server Manager.
2. Click **Routing and Remote Access**, and then select **Network Interfaces**.
3. In the details pane, right-click a demand-dial interface, and then click **Properties**.

   On each tab, confirm that the connection is configured the same as the source server, and then click **OK**.

Router settings

Confirm that the router components installed, and verify that each is configured correctly. The available routing components include:

- **IPv4:** Static Routes, DHCP Relay Agent, IGMP, NAT, and RIPv2
- **IPv6:** Static Routes and DHCPv6 Relay Agent

To verify the routing components

1. Start Server Manager.
2. Click **Routing and Remote Access**.
3. Expand **IPv4**. Examine the list of installed routing components, and ensure that the components required for your deployment are installed.
4. Expand **IPv6** and follow the same process as the previous step.
5. In the navigation pane, under **IPv4**, click **General**.

   The details pane identifies the interfaces that are configured to route packets for each version of IP. Confirm that the list contains the expected interfaces, including configured demand-dial interfaces.
6. In the navigation pane, under **IPv6**, click **General** and follow the same process as the previous step.
7. In the details pane for **General** under **IPv4**, right-click each interface and select **Properties**.

   On each tab confirm that the interface is configured as required for its routing role on the

server.

8. Follow the same process as described in the previous step for the interfaces listed on the under **IPv6** / **General**.
9. Under **IPv4** select **Static Routes** and confirm that the routes to destination networks are correctly configured with the associated interface and destination gateway address.
10. Follow the same process as described in the previous step for the **Static Routes** under **IPv6**.
11. Under **IPv4**, select **NAT**. The details pane shows the interfaces that NAT is configured to use. Right-click each interface and click **Properties**.
    - Confirm that each interface is configured correctly for NAT. There should be at least two interfaces enabled for NAT, one configured as the **Private** interface, and one configured as the **Public** interface.
    - If NAT is responsible for providing IPv4 addresses to clients on the private network, then on **NAT Properties** page, on the **Address Assignment** tab, select the **Automatically assign IP addresses by using the DHCP allocator** check box and enter the address information to be used.
    - If your ISP has provided a pool of addresses to be used by the NAT public interface, ensure that they are configured correctly. The addresses are under **NAT**, on the **Properties** page for the interface, on the **Address Pool** tab. If the addresses that were migrated are not applicable to the target computer, modify the list to use the correct addresses.
    - For each interface under **NAT**, on the interface's **Properties** page on the **Services and Ports** tab, examine the port mappings for services that must be routed to a specific server IP address. Confirm that each service that is to be mapped has the correct address pool entry, private IP address, and port settings configured.
12. Under **IPv4**, select each enabled routing protocol. The details pane shows the interfaces on which the selected routing protocol is enabled. Right-click each interface, and then click **Properties**.

    Confirm that each interface is configured correctly for the selected routing protocol. For example, under **IPv4**/**NAT**, there should be at least two interfaces, one configured as the **Private** interface, and one configured as the **Public** interface.
13. Under **IPv6**, select each enabled routing protocol and follow the same process described in the previous step.
14. Under **IPv4**, right-click each routing protocol, and then select **Properties** to examine the global configuration for that routing protocol.

    Confirm that each protocol is configured correctly for your environment. For example, ensure that the **DHCP Relay Agent** has a list of DHCP server addresses to which it can forward DHCP requests from clients.
15. Under **IPv6**, select each enabled routing protocol and follow the same process described in the previous step.

User and Group accounts

If you migrated the user and group accounts by using the Local User and Group Migration Guide (http://go.microsoft.com/fwlink/?linkid=163774), follow the procedures in its verification section to confirm that the required users and group were migrated successfully.

If you instead used the **-user** and **-group** parameters on the **Import-SmigServerSetting** command, you can manually verify the accounts by using the **Local Users and Groups** MMC snap-in to examine the user and group accounts and confirm that the properties for the accounts are set properly.

Final checks

- If your computer is configured to host VPN/DirectAccess connections, test each type of supported connection to confirm that users can connect.
- If your server is configured to host dial-up connections, verify that client computers can successfully dial-in and connect to the server by using the modems that are installed.
- If your server is configured as an IPv4 or IPv6 router, verify that clients on each attached network can connect through the router to computers on all of the other attached networks. If you use the **ping** command for this test, ensure that Windows Firewall on the router and the client computers is configured to allow ICMP Echo Request and ICMP Echo Reply messages.

See Also

Migrate Remote Access to Windows Server 2012

Remote Access: Prepare to Migrate

Remote Access: Migrate Remote Access

Remote Access: Post-migration Tasks

Remote Access: Post-migration Tasks

Perform the following post-migration tasks to complete your migration:

- Completing the migration
- Configuring firewall rules for VPN
- Configuring firewall rules for DirectAccess
- Restoring Remote Access in the event of migration failure
- Retiring Remote Access on your source server
- Troubleshooting cmdlet-based migration

Note

The post-migration tasks for the source server are optional, depending on your migration scenario.

Completing the migration

Migration is complete as soon as verification efforts demonstrate that the destination server has replaced the source server in serving the network.

If your verification efforts indicate that the migration failed, follow the steps in Restoring Remote Access in the event of migration failure to return to the previous valid configuration. The following list identifies some settings that must be manually configured after the migration is complete.

- **Server name**: If the source server is to be decommissioned, the same name is available to be configured on the destination server. If the IP addresses of the intranet and internet interfaces have changed, it is important to ensure that the intranet and internet DNS servers are updated with the new name and IP addresses. These configuration details also need to be deployed to your users by updating their connection profiles so that they can connect to the correct server.
- **Perimeter network firewall or NAT**: If the destination server is in a perimeter network, the firewall or NAT through which the server is accessed from the Internet must be configured with the new IP address of the RRAS server. Refer to the documentation for your Firewall or NAT router for the relevant configuration instructions.

Configuring firewall rules for VPN

Firewall rules that permit VPN network traffic are included with the Windows Firewall with Advanced Security, but they are disabled by default. The rules that enable the required inbound network traffic must be enabled on the destination server. If there were any other rules explicitly configured on the source server to support RRAS or its roles, they should be configured on the destination server also. If you use non-Microsoft firewall software, refer to the documentation that is provided by the vendor for instructions about how to configure the appropriate rules.

When you migrate RRAS from the source server to the destination server, the firewall rules are not automatically enabled. You must use Windows Firewall with Advanced Security or a Group Policy setting to enable the rules that correspond to the types of RRAS network traffic that must enter your server. The following is a list of rules that you should enable, depending on which RRAS protocols you use:

- Routing and Remote Access (GRE-In)
- Routing and Remote Access (L2TP-In)
- Routing and Remote Access (PPTP-In)
- Secure Socket Tunneling Protocol (SSTP-In)

If you change the default firewall behavior to blocking all traffic that does not match an allow rule then you must enable the outbound allow rules in addition to the inbound rules.

Configuring firewall rules for DirectAccess

When using additional firewalls in your deployment, apply the following Internet-facing firewall exceptions for Remote Access traffic when the Remote Access server is on the IPv4 Internet:

- Teredo traffic—User Datagram Protocol (UDP) destination port 3544 inbound, and UDP source port 3544 outbound.
- 6to4 traffic—IP Protocol 41 inbound and outbound.
- IP-HTTPS—Transmission Control Protocol (TCP) destination port 443, and TCP source port 443 outbound. When the Remote Access server has a single network adapter, and the network location server is on the Remote Access server, then TCP port 62000 is also required.

  Note
  This exemption must be configured on the Remote Access server, while all the other exemptions have to be configured on the edge firewall.

Note
For Teredo and 6to4 traffic, these exceptions should be applied for both of the Internet-facing consecutive public IPv4 addresses on the Remote Access server. For IP-HTTPS the exceptions need only be applied to the address where the public name of the server resolves.

When using additional firewalls, apply the following Internet-facing firewall exceptions for Remote Access traffic when the Remote Access server is on the IPv6 Internet:

- IP Protocol 50
- UDP destination port 500 inbound, and UDP source port 500 outbound.
- Internet Control Message Protocol for IPv6 (ICMPv6) traffic inbound and outbound – for Teredo implementations only.

When using additional firewalls, apply the following internal network firewall exceptions for Remote Access traffic:

- ISATAP—Protocol 41 inbound and outbound
- TCP/UDP for all IPv4/IPv6 traffic
- ICMP for all IPv4/IPv6 traffic

Restoring Remote Access in the event of migration failure

If migration of Remote Access to the destination server fails, you can put the source server back into operation by following these steps:

- If the source server has not been repurposed and the computer name and IP address have not been migrated from the source to the destination server, simply connect the source server to the network and start the Routing and Remote Access service to allow users to connect again.
- If the computer name and IP address have been migrated from the source server to the destination server, rename the destination server to a temporary name and change its IP address to a different IP address. Set the source server computer name and IP address to

the values that were used before the migration, and restart the Routing and Remote Access service on the source server.

- If the previous steps are not valid options, such as when the source server has been repurposed or is otherwise unavailable, use the backup files that were created from the source server, as described in Remote Access: Prepare to Migrate. You can use the backup files any time after migration to restore the original Routing and Remote Access source server if a catastrophic failure occurs.

Estimated time to complete a rollback

You should be able to complete a rollback in one to two hours.

Retiring Remote Access on your source server

After you verify that the migration is complete, the source server can be disconnected from the network and removed from service. Stop RRAS before you remove the computer from the network and turn it off. You can keep this computer as a backup server in the event that you want to revert to your previous Routing and Remote Access configuration.

Troubleshooting cmdlet-based migration

The Windows Server Migration Tools deployment log file is located at %*windir*%\Logs\SmigDeploy.log. Additional Windows Server Migration Tools log files are created at the following locations.

- %*windir*%\Logs\ServerMigration.log
- On Windows Server 2008, Windows Server 2008 R2 and Windows Server 2012: %*localappdata*%\SvrMig\Log
- On Windows Server 2003: %*userprofile*%\Local Settings\Application Data\SvrMig\Log

If migration log files cannot be created in the preceding locations, **ServerMigration.log** and **SmigDeploy.log** are created in %*temp*%, and other logs are created in %*windir*%\System32.

If you are migrating Remote Access and a failure occurs while running the **Import-SmigServerSetting** cmdlet, review the Setupact.log, Setuperr.log, and ServerMigration.log files under %localappdata%\SvrMig\Log. Information about how the Remote Access components migrated is included in the Servermigration.log file.

If a migration cmdlet fails, and the Windows PowerShell session closes unexpectedly with an access violation error message, look for a message similar to the following example in the %*localappdata*%\SvrMig\Logs\setuperr.log file.

```
FatalError [0x090001] PANTHR Exception (code 0xC0000005: ACCESS_VIOLATION) occurred at
0x000007FEEDE9E050 in C:\Windows\system32\migwiz\unbcl.dll (+000000000008E050).  Minidump
attached (317793 bytes).
```

This failure occurs when the server cannot contact domain controllers that are associated with domain users or groups who are members of local groups, or who have rights to files or shares

that are being migrated. When this happens, each domain user or group is displayed in the GUI as an unresolved security identifier (SID). An example of a SID is **S-1-5-21-1579938362-1064596589-3161144252-1006**.

To prevent this problem, verify that required domain controllers or global catalog servers are running, and that network connectivity allows communication between both source and destination servers and required domain controllers or global catalog servers. Then, run the cmdlets again.

If connections between either the source or destination servers and the domain controllers or global catalog servers cannot be restored, do the following.

1. Before you run **Export-SmigServerSetting**, **Import-SmigServerSetting** or **Get-SmigServerFeature** again, remove all unresolved domain users or groups who are members of local groups from the server on which you are running the cmdlet.
2. Before you run **Send-SmigServerData** or **Receive-SmigServerData** again, remove all unresolved domain users or groups who have user rights to files, folders, or shares on the migration source server.

Viewing the content of Windows Server Migration Tools result objects

All Windows Server Migration Tools cmdlets return results as objects. You can save result objects, and query them for more information about settings and data that were migrated. You can also use result objects as input for other Windows PowerShell commands and scripts.

Result object descriptions

The Windows Server Migration Tools **Import-SmigServerSetting** and **Export-SmigServerSetting** cmdlets return results in a list of **MigrationResult** objects. Each **MigrationResult** object contains information about the data or setting that the cmdlet processes, the result of the operation, and any related error or warning messages. The following table describes the properties of a **MigrationResult** object.

Property name	Type	Definition
ItemType	Enum	The type of item being migrated. Values include **General**, **WindowsFeatureInstallation**, **WindowsFeature**, and **OSSetting**.
ID	String	The ID of the migrated item. Examples of values include **Local User**, **Local Group**, and **DHCP**.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise,

Property name	Type	Definition
		False is displayed.
DetailsList	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.

Send-SmigServerData and **Receive-SmigServerData** cmdlets return results in a list of **MigrationDataResult** objects. Each **MigrationDataResult** object contains information about the data or share that the cmdlet processes, the result of the operation, any error or warning messages, and other related information. The following table describes the properties of a **MigrationDataResult** object.

Property name	Type	Definition
ItemType	Enum	The type of migrated item. Values include **File**, **Folder**, **Share**, and **Encrypted File**.
SourceLocation	String	The source location of the item, shown as a path name.
DestinationLocation	String	The destination location of the item, shown as a path name.
Success	Boolean	The value **True** is displayed if migration was successful; otherwise, **False** is displayed.
Size	Integer	The item size, in bytes.
ErrorDetails	List <MigrationResultDetails>	A list of **MigrationResultDetails** objects.
Error	Enum	Errors enumeration for errors that occurred.
WarningMessageList	List <String>	A list of warning messages.

The following table describes the properties of objects within the **MigrationResultDetails** object that are common to both **MigrationResult** and **MigrationDataResult** objects.

Property name	Type	Definition
FeatureId	String	The name of the migration setting that is related to the item. Examples of values

Property name	Type	Definition
		include **IPConfig** and **DNS**. This property is empty for data migration.
Messages	List <String>	A list of detailed event messages.
DetailCode	Integer	The error or warning code associated with each event message.
Severity	Enum	The severity of an event, if events occurred. Examples of values include **Information**, **Error**, and **Warning**.
Title	String	Title of the result object. Examples of values include NIC physical address for IP configuration, or user name for local user migration.

Examples

The following examples show how to store the list of the result objects in a variable, and then use the variable in a query to return the content of result objects after migration is complete.

To store a list of result objects as a variable for queries

1. To run a cmdlet and save the result in variable, type a command in the following format, and then press **Enter**.

   $*VariableName* **=** $(*Cmdlet*)

   The following is an example.

   $ImportResult = $(Import-SmigServerSetting -FeatureId DHCP -User all -Group -Path D:\rmt\DemoStore -force -Verbose)

   This command runs the **Import-SmigServerSetting** cmdlet with several parameters specified, and then saves result objects in the variable **ImportResult**.

2. After the **Import-SmigServerSetting** cmdlet has completed its operations, return the information contained in the result object by typing a command in the following format, and then pressing **Enter**.

   $*VariableName*

   In the following example, the variable is named **ImportResult**.

$*ImportResult*

This command returns information contained in the result objects that were returned by **Import-SmigServerSetting** in the example shown in step 1. The following is an example of the output that is displayed by calling the **ImportResult** variable.

```
            ItemType  ID                                Success
DetailsList
            --------  --                                -------
-----------
          OSSetting  Local User                          True
{Local User, Loc...
          OSSetting  Local Group                         True
{Local Group, Lo...
     WindowsFeature  DHCP                                True
{}
```

Each line of the preceding sample is a migration result for an item that was migrated by using the **Import-SmigServerSetting** cmdlet. The column heading names are properties of **MigrationResult** objects. You can incorporate these properties into another command to return greater detail about result objects, as shown by examples in step 3 and forward.

3. To display a specific property for all result objects in the list, type a command in the following format, and then press **Enter**.

   $<VariableName>| Select-Object -ExpandProperty <PropertyName>

   The following is an example.

   $importResult | Select-Object -ExpandProperty DetailsList

4. You can run more advanced queries to analyze result objects by using Windows PowerShell cmdlets. The following are examples.

   - The following command returns only those details of result objects that have the ID **Local User**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList

   - The following command returns only those details of result objects with an ID of **Local User** that have a message severity equal to **Warning**.

     $ImportResult | Where-Object { $_.ID -eq "Local User" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Severity -eq "Warning") {$_} }

   - The following command returns only the details of result objects with an ID of **Local User** that also have the title **Remote Desktop Users**.

     $ImportResult | Where-Object { $_.ID -eq "Local Group" } | Select-Object -ExpandProperty DetailsList | ForEach-Object { if ($_.Title -eq "Remote

DesktopUsers") {$_} }

More information about querying results

For more information about the cmdlets that are used in the preceding examples, see the following additional resources.

- Where-Object on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134853).
- Select-Object on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134858).
- ForEach-Object on the Microsoft Script Center Web site (http://www.microsoft.com/technet/scriptcenter/topics/msh/cmdlets/foreach-object.mspx)

For more information about Windows PowerShell scripting techniques, see What Can I Do With Windows PowerShell? - Scripting Techniques on the Microsoft Script Center Web site (http://go.microsoft.com/fwlink/?LinkId=134862).

See Also

Migrate Remote Access to Windows Server 2012

Remote Access: Prepare to Migrate

Remote Access: Migrate Remote Access

Remote Access: Verify the Migration

Migrate Windows Server Update Services to Windows Server 2012

This document describes the process to migrate an existing Windows Server Update Services (WSUS) 3.0 SP2 server role to a destination server that is running Windows Server 2012 or Windows Server 2012 R2. This document includes instructions for moving the updates, settings, target groups, and computers to the new server. By using the tools that are described in this document, you can simplify the migration process, reduce migration time, increase the accuracy of the migration process, and help eliminate possible conflicts that might otherwise occur during the migration process.

- Step 1: Plan for WSUS Migration
- Step 2: Prepare to Migrate WSUS
- Step 3: Migrate WSUS
- Step 4: Verify the WSUS Migration

Step 1: Plan for WSUS Migration

The first step in the migration of your Windows Server Update Services (WSUS) to Windows Server 2012 or Windows Server 2012 R2 is to understand the supported and unsupported scenarios and the supported operating systems for this migration. The following checklist describes the steps involved in planning for your WSUS migration.

Task	Description
1.1. Know supported operating systems	Review the list of supported source operating systems and WSUS versions.
1.2. Review supported migration scenarios	Review the list of supported migration scenarios.
1.3. Review migration scenarios that are not supported	Review the list of unsupported migration scenarios.

1.1. Know supported operating systems

Migration from the following operating systems is supported on Windows Server 2012 and Windows Server 2012 R2:

- Windows Server 2008 R2 running WSUS 3.0 SP2
- Windows Server 2008 (full installation option) running WSUS 3.0 SP2
- Windows Server 2003 SP2 running WSUS 3.0 SP2

1.2. Review supported migration scenarios

The following WSUS migration scenarios are supported:

- Windows Server 2012 Standard and Datacenter editions, and servers running Windows Server 2012 R2 can be used as source or destination servers.
- Windows Server 2012 Enterprise edition can be used as a source server.
- Migration between physical operating systems and virtual operating systems.
- Migration from a source server that is running SQL Server 2005 to a destination server that is running SQL Server 2008 R2 SP1.
- Migration from a source server that is running Windows Internal Database to a destination server that is running SQL Server 2008 R2 SP1.
- Migration from a domain to a workgroup or from a workgroup to a domain. However, if the source server is running SQL Server from a remote location, migration from the domain to a workgroup is not supported.
- The destination server must meet the Windows Server 2012 or Windows Server 2012 R2 WSUS role minimum system requirements for hardware and software.

Important

For more information about minimum system requirements and hardware capacity requirements for the WSUS server, see the Deploy Windows Server Update Services in Your Organization.

1.3. Review migration scenarios that are not supported

The following WSUS migration scenarios are not supported:

- Migration from an unsupported version of WSUS (prior to WSUS 3.0 SP2). Upgrade the existing WSUS server to a supported version before you migrate the WSUS server role to Windows Server 2012 or Windows Server 2012 R2.
- Migration from a Server Core installation option (WSUS 3.0 SP2 does not support a Server Core installation).
- Migration from a domain that is using SQL Server from a remote location to a workgroup.
- Migration from a source server that is running SQL Server to a destination server that is running Windows Internal Database.
- Migration from a source server that stores updates on Microsoft Update to a destination server that stores updates on a local WSUS server, and vice versa. Changing the configuration during the migration process is not supported.
- Migration from a source server to a destination server that is running an operating system in a different system UI language. The system UI language is the language of the localized installation package that was used to set up the Windows operating system. For example, you cannot use Windows Server Migration Tools to migrate roles, operating system settings, data, or shares from a computer that is running Windows Server 2008 in the French system UI language to a computer that is running Windows Server 2008 R2 in the German system UI language.

See also

- Step 2: Prepare to Migrate WSUS
- Migrate Windows Server Update Services to Windows Server 2012
- WSUS server role description

Step 2: Prepare to Migrate WSUS

The second step in the migration of your Windows Server Update Services (WSUS) server role involves preparing the destination and source servers. The following checklist describes the steps involved to prepare for your WSUS migration.

Task	Description
2.1. Prepare before you start the migration	Review the recommended guidelines before starting the migration process.
2.2. Prepare the destination server	Understand the steps that must be completed on the WSUS destination server before the migration.
2.3. Prepare the source server	Understand the steps that must be completed on the WSUS source server before the migration.

2.1. Prepare before you start the migration

This migration procedure assumes a working knowledge of deployment basics for Windows Server Update Services (WSUS) 3.0 SP2. For more information about WSUS deployment, see Deploy Windows Server Update Services in Your Organization. We recommend that you make the following decisions and preparations before you start the migration process:

Warning

Upgrade from any version of Windows Server that supports WSUS 3.2 to Windows Server® 2012 R2 requires that you first uninstall WSUS 3.2.

In Windows Server 2012, upgrading from any version of Windows Server with WSUS 3.2 installed is blocked during the installation process if WSUS 3.2 is detected, and you are prompted to first uninstall Windows Server Update Services prior to upgrading Windows Server 2012.

However, because of changes in Windows Server 2012 R2, when upgrading from any version of Windows Server and WSUS 3.2 to Windows Server 2012 R2, the installation is not blocked. Failure to uninstall WSUS 3.2 prior to performing a Windows Server 2012 R2 upgrade will cause the post installation tasks for WSUS in Windows Server 2012 R2 to fail. In this case, the only known corrective measure is to format the hard drive and reinstall Windows Server 2012 R2.

- Configure a location to store updates on the source server. Changing the content storage configuration as part of the migration process is not supported. You can store updates on the local WSUS server or on Microsoft Update. If you want the destination server to store updates in a different location than the source server, the new location must be configured on the source server before migration.
- Confirm that the destination server meets the minimum WSUS hardware requirements and database requirements. For more information about those requirements see Deploy Windows Server Update Services in Your Organization on Microsoft TechNet.

2.2. Prepare the destination server

Before migrating WSUS, set up a new Windows Server 2012 in your organization as the WSUS destination server and install WSUS server role on the destination server. After you have successfully installed the WSUS server role, the Configuration Wizard starts automatically. Close the Configuration Wizard. Do not try to sync the updates at this point, because you will copy the update binary files later in the migration process. The WSUS installation procedure assumes that updates for the new server come from Windows Update.

After this is complete, follow these guidelines:

- If you have decided to use the full installation of SQL Server as the WSUS database, install SQL Server 2008 R2 Standard or SQL Server 2008 R2 Enterprise.
- Download a graphical tool to manage your database on the destination server from Microsoft SQL Server Management Studio Express or Microsoft SQL Server 2008 R2 Management Studio Express.
- Open TCP port 7000 and make sure that it is not being used by other applications. This port is used by Send-SmigServerData and Receive-SmigServerData to perform the data transfer.
- If the destination server is not joined to the source server's domain, visually verify that the time, date, and time zone on the destination server are synchronized with the source server. Use the Windows Control Panel to update the date and time if it is necessary.

Important
For more information about minimum system requirements and hardware capacity requirements for the WSUS server, see the Deploy Windows Server Update Services in Your Organization.

2.3. Prepare the source server

Review and take action based on the following guidelines:

- Refer to Appendix A: Migration Data Collection Worksheet to collect data about the source server.
- Open TCP port 7000 and make sure that it is not being used by other applications. This port is used by Send-SmigServerData and Receive-SmigServerData to perform the data transfer.
- If you have changed the default behavior of Windows Firewall (or another firewall program) to block outgoing traffic on computers that are running Windows Server 2012, you must enable outgoing traffic on UDP port 7000.
- Download a graphical tool for managing your database on the source server at Microsoft SQL Server Management Studio Express.

See also

- Step 3: Migrate WSUS
- Step 1: Plan for WSUS Migration
- WSUS server role description

Step 3: Migrate WSUS

During the third step in the migration of your Windows Server Update Services (WSUS) server role, you will migrate binaries and security groups, back up the database, change the server identity, and apply security settings The following checklist describes the steps involved.

Task	Description
3.1. Migrate WSUS update binaries	Move WSUS update binaries from the source server to the destination server.
3.2. Migrate WSUS security groups	Migrate local users and groups manually or by using Windows Server Migration Tools.
3.3. Back up the WSUS database	Use SQL Server Management Studio to back up and restore the WSUS database, computer groups, update approvals, and WSUS settings.
3.4. Change the WSUS server identity	Change the WSUS server identity on the destination server. Performing this step ensures that there is no effect on clients that are managed by WSUS during the migration process.
3.5. Apply security settings	Configure security settings on the new server. This includes configuring security settings on the destination server that you were using on the source server.
3.6. Review additional considerations	Review some additional actions that you should take after the migration is complete.

3.1. Migrate WSUS update binaries

Use your preferred method to copy WSUS update binaries in the WSUS folder from the source server to the destination server (for example, Windows Server Migration Tools, Windows Explorer, Xcopy, or Robocopy). If you decide to use Windows Server Migration Tools to migrate WSUS update binaries to a destination server that is running Windows Server 2012, see Migrate WSUS Update Binaries from the Source Server to the Destination Server Using Windows Server Migration Tools.

Important
Migrating WSUS update binaries is unnecessary if update files are stored on Microsoft Update.

3.2. Migrate WSUS security groups

You have the option of manually migrate only the WSUS Administrators and WSUS Reporters local security groups. Or, you can use Windows Server Migration Tools to migrate all local users and groups (including the WSUS Administrators and WSUS Reporters local security groups) from the source server to the destination server.

Warning
The WSUS Server Migration Tools can be installed on the server using the Server Manager Add Features option.

Before you perform this procedure, verify that the destination server can resolve the names of domain users who are members of the local group during the import operation. If the source and destination servers are in different domains, the destination server must be able to contact a global catalog server for the forest in which the source domain user accounts are located. Use the following guidelines:

- **If the source server is a member of the domain and the destination server is a domain controller**: Imported local users are elevated to domain users, and imported local groups become Domain Local groups on the destination server.
- **If the source server is a domain controller, and the destination server is not**: Domain Local groups are migrated as local groups, and domain users are migrated as local users.

Use the following procedure to manually migrate users to the WSUS Administrators and WSUS Reporters local security groups.

To manually migrate local users and groups

1. Right-click in the Taskbar, click **Properties**, highlight **Toolbars**, and then click **Address**.
2. Type **lusrmgr.msc**, and then press ENTER.
3. In the console tree of the Local Users and Groups MMC snap-in, double-click **Users**.
4. Manually create a list of the local users.
5. In the console tree of the Local Users and Groups MMC snap-in, double-click **Groups**.
6. Manually add the users from the source server to the WSUS Administrators and WSUS Reporters groups.

Use the following procedure to use Windows Server Migration Tools to migrate users to the WSUS Administrators and WSUS Reporters local security groups.

To use Windows Server Migration Tools to migrate users

1. Open a Windows PowerShell session on the source server and on the destination server.
2. Type the command below and press ENTER:

   ```
   Add-PSSnapin Microsoft.Windows.ServerManager.Migration
   ```

3. In the Windows PowerShell session on the source server, type the following command to export local users and groups to a migration store:

   ```
   Export-SmigServerSetting -User <Enabled | Disabled | All> -Group -Path
   ```

```
<MigrationStorePath> -Verbose
```

MigrationStorePath represents the path of the location where you want to store migrated data. You can also use one of the following values with the **-User** parameter:

- **Enabled**: Export only enabled local users
- **Disabled**: Export only disabled local users
- **All**: Export enabled and disabled local users

4. Press ENTER.

Important

You are prompted to provide a password to encrypt the migration store. Remember this password, because you must provide the same password to import data from the migration store on the destination server. If the path is not a shared location to which the destination server has access, you must copy the migration store to the destination server manually, or to a location that this destination server can access as it runs the **Import-SmigServerSetting** cmdlet.

5. In the Windows PowerShell session on the destination server, type the following command to import local users and groups from the migration store that you created in Step 2:

```
Import-SmigServerSetting -User <Enabled | Disabled | All> -Group -Path
<MigrationStorePath> -Verbose
```

MigrationStorePath represents the path of the location from which you want to import migrated data. You can also use one of the following values with the **-User** parameter:

- **Enabled**: Export only enabled local users
- **Disabled**: Export only disabled local users
- **All**: Export enabled and disabled local users

6. Press ENTER.

Warning

After you enter the **Import-SmigServerSetting** cmdlet, you are prompted to provide the same password to decrypt the migration store that you created during the export process.

3.3. Back up the WSUS database

WSUS servers can be configured to use Windows Internal Database, the database software that is included with WSUS, or the full version of SQL Server. Regardless of which database option the source server is running, perform the following procedures to back up the WSUS database on the source server and restore the database to the destination server.

For an overview of backup and command-line syntax, see the following topics in SQL Server TechCenter:

- Backup Overview

- BACKUP (Transact-SQL)

For an overview of restore and command-line syntax, see the following topics in SQL Server TechCenter:

- Restore and Recovery Overview
- RESTORE (Transact-SQL)

Important
SQL Server Management Studio must be run with elevated administrator permissions throughout this procedure.

To back up the WSUS database on the source server

1. After you connect to the appropriate instance of the database in Object Explorer, click the server name to expand the server tree.

   Note
   If the source server is using Windows Internal Database, the query changes depending on which version of WSUS you are currently running. For WSUS 3.2, the query is: **\\.\pipe\mssql$microsoft##ssee\sql\query**, and for WSUS on Windows Server 2012, the query is: **\\.\pipe\Microsoft##WID\tsql\query**.
2. Expand **Databases**, and select the **SUSDB** database.
3. Right-click the database, point to **Tasks**, and then click **Back Up**. The **Back Up Database** dialog box appears.
4. In the **Database** list, verify the database name.
5. In the **Backup type** list, select **Full**.
6. Select **Copy Only Backup**. A copy-only backup is a SQL Server backup that is independent of the sequence of conventional SQL Server backups.
7. For **Backup component**, click **Database**.
8. Accept the default backup set name that is suggested in the **Name** text box, or enter a different name for the backup set.
9. Optionally, in the **Description** text box, enter a description of the backup set.
10. Specify when the backup set will expire and can be overwritten without explicitly skipping verification of the expiration data.
11. Choose the backup destination by clicking Disk.

    Important
    To remove a backup destination, select it and then click **Remove**. To view the contents of a backup destination, select it and then click **Contents**.
12. In the **Select a page** pane, click **Options** to view or select the advanced options.
13. On the **Overwrite Media** option, click one of the following:
    - **Back up to the existing media set** For this option, click **Append to the existing backup set** or **Overwrite all existing backup sets**. Optionally:
      - Click **Check media set name and backup set expiration** to cause the backup

operation to verify the date and time at which the media set and backup set expire.

- Enter a name in the **Media set name** text box. If no name is specified, a media set with a blank name is created. If you specify a media set name, the media (tape or disk) is checked to see whether the name matches the name that you enter here.

- **Back up to a new media set, and erase all existing backup sets** For this option, enter a name in the **New media set name** text box, and, optionally, describe the media set in the **New media set description** text box.

14. In the **Reliability** section, optionally select:
    - **Verify backup when finished.**
    - **Perform checksum before writing to media.**
    - **Continue on checksum error.**
15. SQL Server 2008 Enterprise support backup compression. By default, whether a backup is compressed depends on the value of the **Backup-compression default** server configuration option. Regardless of the current server-level default, you can compress the backup or prevent compression at this time. To compress the backup, select **Compress backup**, or to prevent compression, select **Do not compress backup**.

To restore the WSUS database backup on the destination server by using SQL Server Management Studio

1. After you connect to the appropriate instance of the database in Object Explorer, click the server name to expand the server tree.

   Important
   If the source server is using Windows Internal Database, the database name is: **\\.\pipe\Microsoft##WID\tsql\query**.
2. Click **New Query** and copy the following SQL command to drop the WSUS database

   USE masterGOALTER DATABASE SUSDB SET SINGLE_USER WITH ROLLBACK IMMEDIATEGODROP DATABASE SUSDBGO
3. Click **Execute**, to run the query
4. Run the following query:

   RESTORE DATABASE [SUSDB] FROM DISK = N'C:\SUSDB.bak' WITH FILE = 1, MOVE N'SUSDB' TO N'c:\Windows\WID\Data\susdb.mdf', MOVE N'SUSDB_log' TO N'c:\Windows\WID\Data\SUSDB_log.ldf', NOUNLOAD, STATS = 10

   Important
   Drive C: that is used in this example will vary according to the actual storage location for the files.
5. In the **Backup type** list, select **Full**.

   Warning
   Running the previous query will result in the following error message:

Msg 3605, Level 16, State 1, Line 5 Schema verification failed for database 'SUSDB'.Msg 3013, Level 16, State 1, Line 5RESTORE DATABASE is terminating abnormally

Disregard the error message and continue.

6. Open an elevated command prompt in Windows Server 2012, and run the following command:

   %programfiles%\update services\tools\wsusutil postinstall [sql parameter] [content parameter]

   Important
   For WID, do not specify the SQL parameter.

To restore the WSUS database backup on the destination server by using SQL Server Management Studio

1. After you connect to the appropriate instance of the database in **Object Explorer**, click the server name to expand the server tree.

   Important
   If the source server is using Windows Internal Database, the database name is \\.*pipe**Microsoft##WID**tsql**query*.

2. Click **New Query** and copy the following SQL command to drop the WSUS database (SUSDB):

```
USE master

GO

ALTER DATABASE SUSDB SET SINGLE_USER WITH ROLLBACK IMMEDIATE

GO

DROP DATABASE SUSDB

GO
```

3. Click **Execute** to run the query.
4. Run the following query:

```
RESTORE DATABASE [SUSDB] FROM  DISK = N'C:\SUSDB.bak' WITH  FILE = 1,  MOVE
N'SUSDB' TO N'c:\Windows\WID\Data\susdb.mdf',  MOVE N'SUSDB_log' TO
N'c:\Windows\WID\Data\SUSDB_log.ldf',  NOUNLOAD,  STATS = 10
```

   Important
   Drive C: that is mentioned in this example will vary according to the actual storage location for the files.

5. This will result in the following error message:

```
Msg 3605, Level 16, State 1, Line 5

Schema verification failed for database 'SUSDB'.

Msg 3013, Level 16, State 1, Line 5
```

```
RESTORE DATABASE is terminating abnormally
```

Disregard the error message and continue.

6. Open an elevated command prompt in Windows Server 2012, and run the following command:

```
%programfiles%\update services\tools\wsusutil postinstall [sql parameter] [content parameter]
```

Important
For WID, do not specify the SQL parameter.

When a database is restored to a different server, it contains a set of users and permissions, although there may be no corresponding user log on information, or the log on information may not be associated with the same users. This condition is known as having "orphaned users." See article 168644 in the Microsoft Knowledge Base for instructions about how to resolve orphaned users.

After you restore a SQL Server 2005 database to SQL Server 2008, the database becomes available immediately, and it is then automatically upgraded.

3.4. Change the WSUS server identity

The WSUS server identity on the destination server must be changed. Performing this step guarantees that WSUS-managed clients are not affected during the migration process. If the source server and the destination server run with the same identity, and a change is made to one of the servers, the communication between the client and server will fail.

To change the WSUS server identity

1. On the destination server, open a Windows PowerShell session with elevated user rights and run the following script:

```
$updateServer = get-wsusserver

$config = $updateServer.GetConfiguration()

$config.ServerId = [System.Guid]::NewGuid()

$config.Save()
```

2. As soon as the server identity is changed, run the following command to generate a new encryption key:

```
%ProgramFiles%\Update Services\Tools\wsusutil.exe postinstall
```

3.5. Apply security settings

Refer to the settings that you recorded in the Migration Data Collection Worksheet, and then complete the following tasks to apply the security settings that you were using on the source server to the destination server.

- **SMTP server settings**: If you are using an authenticating proxy or the email notification feature to an SMTP server that requires a password (or both), you must manually configure the proxy and email notification to the SMTP server, and enter the SMTP password on the new destination server if you are using email notification.
- **Code signing certificate**: If you are using an advanced management tool that exposes local update publishing (such as Microsoft System Center Essentials 2007 or Microsoft System Center Configuration Manager 2007), copy the code-signing certificate.

To initialize a trust relationship between the update server and its clients, use the following procedures to point the downstream servers to the new WSUS server, and point the WSUS clients to the new WSUS server.

Point the downstream servers to the new WSUS server

If you have downstream servers in your WSUS configuration, and if the server identity on the destination server was changed, perform the following procedure to point them to the new WSUS server.

To connect a downstream server to an upstream server

1. In the navigation pane of the downstream WSUS Server Administration console, click **Options**.
2. Click **Update Source and Proxy Server**, and then click the **Update Source** tab.
3. Select the **Synchronize from another Windows Server Update Services server** check box, and then type the server name and port number in the corresponding text boxes.
4. Repeat step 1 through step 3 for additional downstream servers, if applicable. The synchronization can take several minutes to several hours to finish.
5. To confirm that the downstream servers are synchronizing with the upstream server, in the WSUS Administration Console on the upstream WSUS server, click **Downstream Servers**. In the **Status** pane, confirm that the server's **Last Synchronization** date is after the date that the previous steps were completed.

Point the WSUS clients to the new WSUS server

If the server identity on the destination server was changed, use the following procedure to point the WSUS clients to the new WSUS destination server.

To point the WSUS clients to the new destination server

1. Open the Local Group Policy Editor, and in **Specify intranet Microsoft update service policy**, change the URL to reflect the new WSUS server.
2. Update the Group Policy settings that are used to point WSUS clients to the WSUS server by entering the FQDN of the new WSUS server. After you have updated the Group Policy settings, WSUS clients will synchronize with the new WSUS server.
3. To force the clients to detect the new destination server, open a command prompt, and run `wuauclt.exe /resetauthorization /detectnow`.

Note

To make sure that WSUS clients point to the new WSUS server immediately, you must force detection, which causes WSUS to update computer group membership. If you do not force a detection, it can take up to four hours for clients to point to the new WSUS server.

4. Depending on the number of clients, the initial synchronization can take several minutes to several hours to finish. To confirm that the synchronization is complete, in the WSUS Administration Console, expand **Computers**, and then click **All Computers**. In the **Status** pane, click **Any**, and then click **Refresh**. Confirm that the computers that you expect to see synchronizing to this WSUS server are listed. The **Last Contact Date** has to be refreshed with a post-migration time stamp.

   Tip

   To force a report that the **Last Contact Date** was updated, run `wuauclt.exe /resetauthorization /detectnow`, and then run `wuauclt.exe /reportnow`.

5. After the clients have synchronized, confirm that clients are installing approved updates based on your WSUS configuration settings. In the WSUS Administration Console, click **Reports**, and then click **Computer Tabular Status**. Select the **Report Options** that are applicable to the clients, and then click **Run Report**.
6. To make sure that no WSUS clients are still pointing at the old WSUS server, wait a week and then open the WSUS Administration Console on the old WSUS server. Expand **Computers**, and then click **All Computers**. In the **Status** pane, click **Any**, and then click **Refresh**. Sort on **Last Status Report**. There should be no clients that have a **Last Status Report** date after the date that the synchronization completed.

3.6. Review additional considerations

After the migration is complete, consider the following:

- It is important to have a backup plan for restoring the WSUS server role if there is a migration failure. You do not need to roll back the migration on the source server because the migration process makes no changes to it. You do not need to roll back the migration on the destination server because it is a new server.
- After you have confirmed that no WSUS clients are contacting the old WSUS server, you can uninstall WSUS from the source server. To perform this operation, see the section titled **Retire the WSUS role on the source server (optional)** in the Windows Server Update Services 3.0 SP2 Migration Guide topic: Post-migration Tasks for WSUS.

See also

- Step 4: Verify the WSUS Migration
- Step 2: Prepare to Migrate WSUS
- WSUS server role description

Step 4: Verify the WSUS Migration

The final step in the migration of your Windows Server Update Services (WSUS) server role is to verify that the migration was performed correctly and if the clients can obtain updates from the new WSUS server.

Task	Description
4.1. Verify the destination server configuration	Verify if the destination server is synchronized.
4.2. Verify client computer functionality	Verify if the clients are correctly obtaining updates from the new WSUS server.

4.1. Verify the destination server configuration

Perform the following procedure on the new WSUS destination server to verify that it is configured properly and functioning correctly before you point the WSUS clients and any downstream servers to the new WSUS server.

1. In Server Manager, click **Tools**, and then click **Windows Server Update Services**.
2. In the WSUS Administration Console, expand **Computers**, and verify that all the Computer Groups that existed on the source server are displayed.
3. Expand **Synchronizations**. In the **Actions** pane, click **Synchronize now**. After the synchronization is complete, (this may take several minutes), confirm that **Succeeded** is displayed in the **Results** column.

If the synchronization fails, click **Options**. Confirm that the **Update Source and Proxy Server** settings and password are correct. Confirm that the firewall access is configured correctly for the new server's environment. Make the necessary changes, and then run the synchronization again.

4.2. Verify client computer functionality

Select a client computer so that you can force a detection to verify that the client and server communication is functioning correctly. Use the ollowing procedure to perform this verification:

1. Open a command prompt and type `wuauclt.exe /detectnow` to force the detection.
2. After the detection is finished, open Windows Explorer and check the **%WinDir%\WindowsUpdate.log** to verify that the forced detection was successful.

See also

- Step 3: Migrate WSUS
- WSUS server role description

Migrating Clustered Services and Applications to Windows Server 2012

This guide provides step-by-step instructions for migrating clustered services and applications to a failover cluster running Windows Server 2012 by using the Migrate a Cluster Wizard. Not all clustered services and applications can be migrated using this method. This guide describes supported migration paths and provides instructions for migrating between two multi-node clusters or performing an in-place migration with only two servers. Instructions for migrating a highly available virtual machine to a new failover cluster, and for updating mount points after a clustered service migration, also are provided.

Operating system requirements for clustered roles and feature migrations

The Migrate a Cluster Wizard supports migration to a cluster running Windows Server 2012 from a cluster running any of the following operating systems:

- Windows Server 2008 with Service Pack 2 (SP2)
- Windows Server 2008 R2 with Service Pack 1 (SP1)
- Windows Server 2012

Migrations are supported between different editions of the operating system (for example, from Windows Server Enterprise to Windows Server Datacenter), between x86 and x64 processor architectures, and from a cluster running a core installation of Windows Server or Microsoft Hyper-V Server to a cluster running a full version of Windows Server.

The following migrations scenarios are not supported:

- Migrations from Windows Server 2003 or Windows Server 2003 R2 to Windows Server 2012 are not supported. You should first upgrade to Windows Server 2008 R2 SP1 or Windows Server 2008 SP2, and then migrate the resources to Windows Server 2012 using the steps in this guide.
- The Migrate a Cluster Wizard does not support migrations from a Windows Server 2012 failover cluster to a cluster with an earlier version of Windows Server.

Important

Before you perform a migration, you should install the latest updates for the operating systems on both the old failover cluster and the new failover cluster.

Target audience

This migration guide is designed for cluster administrators who want to migrate their existing clustered services and applications that are running on an existing failover cluster to a Windows Server 2012 failover cluster. The focus of the guide is the steps required to successfully migrate

the resources from one cluster to another by using the Migrate a Cluster Wizard in Failover Cluster Manager.

General knowledge of how to create a failover cluster, configure storage and networking, and deploy and manage the clustered roles and features is assumed.

It is also assumed that customers who will use the Migrate a Cluster Wizard to migrate highly available virtual machines have a basic knowledge of how to create, configure, and manage highly available Hyper-V virtual machines.

What this guide does not provide

The scenarios in this guide provide step-by-step instructions for using the Migrate a Cluster Wizard in Failover Cluster Manager to perform a standard migration of a clustered service or application to a Windows Server 2012 failover cluster. Although this guide identifies services and applications that require special handling during a wizard-based migration, the guide does not provide specific instructions for migrating individual clustered services and applications, including special requirements and dependent server roles and features. For information about migration requirements for specific server roles and features, see Migrate Roles and Features to Windows Server.

This guide does not provide instructions for migrating clustered services and applications by any means other than by using the Copy Cluster Roles Wizard.

Planning considerations for migrations between failover clusters

As you plan a migration to a failover cluster running Windows Server 2012, consider the following:

- Microsoft supports a failover cluster solution for Windows Server 2012 only if all the hardware devices are marked as "Certified for Windows Server 2012." In addition, the complete configuration (servers, network, and storage) must pass all tests in the Validate a Configuration Wizard, which is included in the Failover Cluster Manager snap-in. For more information, see Validate Hardware for a Failover Cluster.
- Hardware requirements are especially important if you plan to continue to use the same servers or storage for the new cluster that the old cluster used. When you plan the migration, you should check with your hardware vendor to ensure that the existing storage is certified for use with Windows Server 2012. For more information about hardware requirements, see Failover Clustering Hardware Requirements and Storage Options.
- The Migrate a Cluster Wizard assumes that the migrated role or feature will use the same storage that it used on the old cluster. If you plan to migrate to new storage, you must copy or move of data or folders (including shared folder settings) manually. The wizard also does not copy any mount point information used in the old cluster. For information about handling mount points during a migration, see Cluster Migrations Involving New Storage: Mount Points.

- Not all clustered services and features can be migrated to a Windows Server 2012 failover cluster by using the Migrate a Cluster Wizard. To find out which clustered services and applications can be migrated by using the Migrate a Cluster Wizard, and operating system requirements for the source failover cluster, see Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012.

Migration scenarios that use the Migrate a Cluster Wizard

When you use the Migrate a Cluster Wizard for your migration, you can choose from a variety of methods to perform the overall migration. This guide provides step-by-step instructions for the following two methods:

- **Create a separate failover cluster running Windows Server 2012 and then migrate to that cluster**. In this scenario, you migrate from a multi-node cluster running Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012. For more information, see Migration Between Two Multi-Node Clusters.
- **Perform an in-place migration involving only two servers**. In this scenario, you start with a two-node cluster that is running Windows Server 2008 R2 or Windows Server 2008, remove a server from the cluster, and perform a clean installation (not an upgrade) of Windows Server 2012 on that server. You use that server to create a new one-node failover cluster running Windows Server 2012. Then you migrate the clustered services and applications from the old cluster node to the new cluster. Finally, you evict the remaining node from the old cluster, perform a clean installation of Windows Server 2012 and add the Failover Clustering feature to that server, and then add the server to the new failover cluster. For more information, see In-Place Migration for a Two-Node Cluster.

This guide also provides step-by step instructions that describe how to migrate highly available virtual machines as part of a wizard-based migration. For requirements and process steps, see Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard.

Note

We recommend that you test your migration in a test lab environment before you migrate a clustered service or application in your production environment. To perform a successful migration, you need to understand the requirements and dependencies of the service or application and the supporting roles and features in Windows Server in addition to the processes that this migration guide describes.

In this guide

Additional References

Related references

What's New in Failover Clustering in Windows Server 2012

Failover Clustering Overview

Failover Clustering Hardware Requirements and Storage Options

Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012

This topic provides guidance for migrating specific clustered services and applications to a failover cluster running the Windows Server 2012 operating system by using the Migrate a Cluster Wizard in Failover Cluster Manager. The topic covers supported migration paths, provides an overview of wizard-based migration, and notes which clustered services and applications require special handling during migration.

Migration paths for specific migrations

The following table lists the operating system versions on a source failover cluster that can be migrated to a failover cluster running Windows Server 2012 for each clustered service or application. Migrations between failover clusters created with physical computers and failover clusters that are created from virtual machines (also known as a *guest cluster*) are supported.

Supported migrations for clustered roles and resources to a Windows Server 2012 failover cluster

Clustered role or resource	From Windows Server 2008 SP2	From Windows Server 2008 R2 SP1	From Windows Server 2012
Cluster Registry settings	Yes	Yes	Yes
Cluster Shared Volumes (CSV)	No	Yes	Yes
DFS Namespace (DFS-N)	Yes	Yes	Yes
DFS Replication (DFS-R)	No	Yes	Yes
DHCP Server	Yes	Yes	Yes
Distributed Network	No	No	Yes

Clustered role or resource	From Windows Server 2008 SP2	From Windows Server 2008 R2 SP1	From Windows Server 2012
Name (DNN)			
File Server	Yes	Yes	Yes
Scale-out File Server for application data	No	No	Yes
Generic Application	Yes	Yes	Yes
Generic Script	Yes	Yes	Yes
Generic Services	Yes	Yes	Yes
Virtual Machines	Yes	Yes	Yes
Hyper-V Replica Broker	No	No	Yes
IP addresses (IPV4, IPV6, IPv6 tunnel addresses)	Yes	Yes	Yes
iSCSI Target Server	No	Yes	Yes
Internet Storage Name Service (iSNS)	Yes	Yes	Yes
Message Queuing (MSMQ), MSMQ triggers	Yes	Yes	Yes
Microsoft Distributed Transaction Coordinator (MSDTC)	Yes	Yes	Yes
Network Name resources	Yes	Yes	Yes
NFS shares	Yes	Yes	Yes
Other Server	Yes	Yes	Yes
Physical Disk resource	Yes	Yes	Yes
WINS Server	Yes	Yes	Yes

Cluster roles that cannot be migrated

Some services and applications that can run in a failover cluster on Windows Server 2012 cannot be migrated by using the Migrate a Cluster Wizard—in some cases because they were not supported on earlier versions of clustering. The Migrate a Cluster Wizard in Windows Server 2012 cannot be used to migrate the following clustered roles:

- Microsoft SQL Server
- Microsoft Exchange Server
- Print Spooler – In Windows Server 2012, the print spooler is no longer a clustered resource. Instead, high availability is defined as a highly available virtual machine running on a single cluster node. The Print Server role is installed on a single virtual machine, which can be migrated to other nodes automatically or manually. For more information, see High Availability Printing Overview.
- DFS Replication (DFS-R) from Windows Server 2008 - Migration from Windows Server 2008 R2 or Windows Server 2012 is supported, but not from Windows Server 2008.
- Remote Desktop Connection Broker - In Windows Server 2012, the active/passive clustering model for the RD Connection Broker role service, used in earlier versions of Windows Server, is replaced by the Active/Active Broker feature, which eliminates the need for clustering and provides a fully active/active model. For more information, see the blog entry RD Connection Broker High Availability in Windows Server 2012.
- Volume Shadow Copy Service (VSS) tasks
- Task Scheduler tasks (Windows Server 2012 only)
- Cluster Aware Updating (CAU) settings (Windows Server 2012 only)

Roles restricted to a single instance per cluster

For the following roles, only one instance per failover cluster is supported:

- DHCP Server
- WINS Server
- iSCSI Target Server
- Hyper-V Replica Broker (Windows Server 2012 only)

For those roles, the Migrate a Cluster Wizard will not attempt to create a second role instance if one instance already exists on the target cluster.

Migrations for which the Migrate a Cluster Wizard performs most or all steps

For the following clustered services or applications, The Migrate a Cluster Wizard performs most or all steps for a migration to a Windows Server 2012 failover cluster:

- Distributed File System (DFS) Namespace
- Generic Application

- Generic Script
- Generic Service
- IPv4 Address, when migrating within the same subnet
- IPv6 Address or IPv6 Tunnel Address
- Internet Storage Name Service (iSNS)
- Network Name (other than the cluster name)

  If Kerberos authentication is enabled for the Network Name resource, the migration wizard prompts you for the password for the Cluster service account that is used by the old cluster.
- NFS
- Physical Disk (resource settings only; does not copy data to new storage)
- Windows Internet Name Service (WINS) (Extra steps might be required if you migrate to new storage, and you use a different drive letter on the path to the new database.)

For step-by-step instructions for performing a migration between two multimode failover clusters, see Migration Between Two Multi-Node Clusters. For step-by-step instructions for performing a stand-alone migration while upgrading a single failover cluster, see In-Place Migration for a Two-Node Cluster.

Migration within mixed environments

The Migrate a Cluster Wizard can migrate clustered resources within mixed environments. For example, the wizard accommodates the following differences in the source and destination environments:

- Migrate static IP addresses to a cluster using DHCP.
- Migrate IPv4 resources into an IPv6 environment.
- Migrate across routed subnets.
- Migrate a physical cluster to a guest (virtual) cluster (with the exception of Hyper-V clusters, which must run on physical computers).
- Migrate between different editions of the operating system (for example, from Windows Server Enterprise to Windows Server Datacenter), between x86 and x64 processor architectures, and from a cluster running Windows Server Core or Microsoft Hyper-V Server to a cluster running a full version of Windows Server.

During migration, the wizard allows you to address name conflicts between resource groups, resources, and share names and to address drive letter collisions. The wizard resolves the conflicts as part of the post-migration repair process.

Important

The Migrate a Cluster Wizard moves resources, not data. If you plan to migrate to new storage, you must move the data and folders yourself.

Additional steps for a wizard-based migration

Some additional steps typically are needed before or after you run the wizard, including the following:

- Install server roles and features that are needed in the new cluster. In most cases, you must install the role or feature on all nodes of the cluster.
- Copy or install any associated applications, services, or scripts on the new cluster (all nodes).
- If a migrated role or feature uses the same storage, take the services and storage offline on the old cluster and then make the storage available to the new cluster.
- If a migrated role or feature uses new storage, ensure that any data and folders are copied to new storage. Verify permissions on any shared subfolders that were migrated.
- If the new cluster is on a different subnet, provide static IP addresses.
- If the new cluster uses a different volume letter, update drive path locations for applications.
- Configure Task Manager tasks on the new cluster. (Windows Server 2012 only)
- For a virtual machine, install the latest integration services on the new virtual machine. Configure Volume Shadow Copy Service (VSS) backups. For a Windows Server 2012 migration, configure Hyper-V Replica settings.
- Configure Cluster Aware Updating (CAU). (Windows Server 2012 only)

Migration reports

The wizard provides both a Pre-Migration Report and a Post-Migration Report, which provide important information. We recommend that you review both reports while performing a migration:

- The Pre-Migration Report explains whether each resource that you plan to migrate is eligible for migration.
- The Post-Migration Report contains information about the success of the migration, and describes additional steps that might be needed before you bring the migrated resources online.

Note

Two resource groups are never migrated: **Cluster Core Resources Group** and **Available Storage Group**. You can ignore these resource groups in the migration reports.

Clustered role and feature migrations that require extra steps

This section provides guidance for migrating clustered roles and features that require additional steps before or after you run the Migrate a Cluster Wizard to perform a cluster migration.

- Clustered DFS Replication migrations
- Clustered DHCP migrations
- Clustered DTC migrations
- Clustered File Server and Scale-out File Server migrations

- Clustered FSRM migrations
- Clustered Message Queuing (MSMQ) migrations
- Other Server migrations involving resource types not built into failover clusters
- Clustered virtual machine migrations

Clustered DFS Replication migrations

Before you migrate clustered Distributed File System (DFS) Replication (also known as DFS-R or DFSR) to a cluster running Windows Server 2012, you must add the new cluster to the DFS replication group to which the old cluster belongs, and then wait until DFS Replication synchronizes the data to the new cluster. After data synchronization is complete, you can decommission the old cluster. For step-by-step guidance, see File and Storage Services: Prepare to Migrate and File and Storage Services: Post-Migration Tasks.

To migrate clustered instances of DFS Replication between clusters running Windows Server 2012

1. Obtain the name of the cluster to which you will migrate. In Active Directory, this is the name that is used for the computer account of the cluster itself (also called the cluster name object or CNO). Add this name to the replication group that you will migrate. For more information, see Add a member to a replication group.
2. Wait until DFS Replication finishes synchronizing the replicated data to the cluster to which you will migrate.
3. If you plan to decommission the cluster from which you migrated, remove its network name from the replication group. If necessary, destroy the cluster.

For more information about DFS Replication in Windows Server 2012, see DFS Namespaces and DFS Replication Overview.

Clustered DHCP migrations

When migrating clustered Dynamic Host Configuration Protocol (DHCP) to a cluster running Windows Server 2012, the Migrate a Cluster Wizard migrates resources and settings, but not the DHCP database. For information about how to migrate the DHCP database, see DHCP Server Migration: Migrating the DHCP Server Role. The information in the topic also applies to migrations from Windows Server 2008 R2 to Windows Server 2012. The topic includes information about migrating from a cluster.

Note

Although the migration of the clustered DHCP role is supported, in Windows Server 2012 there is the option to use DHCP failover. DHCP failover provides redundancy and load balancing without clustered DHCP. For more information, see Migrate to DHCP Failover and Understand and Deploy DHCP Failover.

Clustered DTC migrations

Before you begin the migration of clustered Distributed Transaction Coordinator (DTC) to a cluster running Windows Server 2012, you must make sure the list of transactions stored by DTC is empty. This is referred to as *draining the transaction logs*. If you do not drain the logs, the information in the logs (the transaction state information for unresolved transactions) will be lost during the migration. Unresolved transactions include **Active**, **In Doubt**, and **Cannot Notify** transactions.

To drain DTC transaction logs of unresolved transactions

1. Stop the application that creates transactions on the clustered instance of DTC that is being migrated.
2. On a node of the cluster that you are migrating from, click **Start**, point to **Administrative Tools**, and then click **Component Services**. (In Windows Server 2012, open **Component Services** directly from the Start screen.)
3. Expand **Component Services**, expand **Computers**, expand **My Computer**, expand **Distributed Transaction Coordinator**, and then expand **Clustered DTCs**.
4. Expand the clustered instance of DTC that you are migrating, and then click **Transaction List**.
5. View the transaction list to see if it is empty. If there are transactions listed, then either wait for them to be completed or right-click each transaction, click **Resolve**, and then select **Forget**, **Commit**, or **Abort**.

   For information about the effect of each of these options, see Transaction State Resolution After System Failure.

For additional information, see View Transaction Information.

Clustered File Server and Scale-out File Server migrations

You can migrate a clustered file server from Windows Server 2008 R2 or Windows Server 2008 to a failover cluster running Windows Server 2012 by using the Migrate a Cluster Wizard.

Clustered file server migrations

If you plan to migrate to new storage, keep in mind that if the migrated files and folder inherit permissions from their parents, during migration it is the inheritance setting that is migrated, not the inherited permissions. Therefore it is important to make sure that the parent folders on the source server and the destination server have the same permissions to maintain the permissions on migrated data that has inherited permissions. After the file server migration, it's important to verify the folder permissions after the migration. Sometimes folder permissions reset to Read-only during a file server migration.

You do not need to migrate the quorum resource. When you run the Create a Cluster Wizard in Windows Server 2012, the cluster software automatically chooses the quorum configuration that

provides the highest availability for your new failover cluster. You can change the quorum configuration on the new cluster if necessary for your specific environment.

Scale-out File Server migrations

The Scale-out File Server feature was introduced in Windows Server 2012. You can use the Migrate a Cluster Wizard to migrate a scale-out file server from one Windows Server 2012 failover cluster to another Windows Server 2012 failover cluster.

- The new failover cluster must use the same storage that the old cluster used.
- When you prepare for the migration, after you add the File Server role to each cluster node, you must configure the File Server role as the **Scale-out File Server for application data** role type. For more information, see Deploy Scale-Out File Server.

Clustered FSRM migrations

To migrate the File Server Resource Manager (FSRM) classification, storage reporting, and file management task configuration on a clustered file server running Windows Server 2008 R2 or Windows Server 2008 to a failover cluster running Windows Server 2012, you must export the configuration from one FSRM server node in the cluster and then import the configuration to another FSRM server. These steps must be performed locally on one node of the cluster. You then fail over the other nodes until this process is complete. For step-by-step instructions, see Migrate File and Storage Services to Windows Server 2012.

Important
When you migrate the configuration, FSRM requires that you use the same drive letters on both the source and destination servers.

Clustered Message Queuing (MSMQ) migrations

When you migrate a clustered instance of Message Queuing (also known as MSMQ) to a cluster running Windows Server 2012, it's important to take the following precautions to ensure that the data is preserved and you can bring the service online on the new cluster:

- Before you migrate, you should back up the data that is associated with clustered instances of Message Queuing. This ensures that you can restore service-specific Message Queuing data if it is accidentally deleted during migration. For more information about Message Queuing backup and restore, see Backing up and restoring messages.
- During the migration, it's important to make sure that the migration is complete before you delete either clustered instance of Message Queuing (old or new). Otherwise, service-specific data for Message Queuing might be deleted from the shared storage, which prevents the remaining Message Queuing resource from coming online. After the migration is complete and you are ready to delete a clustered instance of Message Queuing (old or new), first remove the disk resource from that clustered instance and take the disk offline. Then delete the clustered instance of Message Queuing.

Other Server migrations involving resource types not built into failover clusters

Before you use the Migrate a Cluster Wizard to migrate an application that uses a clustered resource type that is not built into failover clustering, be sure to add the resource type to the new cluster. You can then use the Migrate a Cluster Wizard to migrate your clustered application. In this situation, the Migrate a Cluster Wizard attempts a "best effort" migration.

To add a resource type to a failover cluster running Windows Server 2012

1. Open Failover Cluster Manager from the Start screen of any node in the cluster running Windows Server 2012.
2. If the cluster to which you want to migrate is not displayed, in the console tree, right-click **Failover Cluster Manager**, click **Connect to Cluster**, select the cluster that you want to migrate to, and then click **OK**.
3. In the console tree, right-click the cluster, and then click **Properties**.
4. Click the **Resource Types** tab, and then click **Add**.
5. Specify the following information for the resource type:
    - **Resource DLL path and file name**: The path and file name of the resource dynamic-link library (DLL) that the Cluster service should use when it communicates with your service or application.
    - **Resource type name**: The name that the Cluster service uses for the resource type. This name stays the same regardless of the regional and language options that are currently selected.
    - **Resource type display name**: The name that is displayed for the resource type. This name might vary when you make changes to regional and language options.

Clustered virtual machine migrations

You can use the Migrate a Cluster Wizard to migrate highly available virtual machines deployed using Hyper-V from a Windows Server 2008 R2 or Windows Server 2008 failover cluster to a cluster running Windows Server 2012. Using the wizard, you can migrate the Virtual Machine clustered role, select highly available virtual machines to migrate, and update virtual network settings for the virtual machines on the new cluster.

Migrating a highly available virtual machine requires some additional steps:

- You must merge or discard all shadow copies before you migrate the volume that contains the virtual machines. You should back up the volumes before you begin merging or discarding shadow copies.
- If you migrate one virtual machine that is stored on Cluster Shared Volume (CVS) volume, the Migrate a Cluster Wizard migrates all virtual machines on that volume. This restriction does not apply if you are migrating a Scale-out File Server cluster between Windows Server 2012 failover clusters. The Scale-out File Server cluster does not use CSV volumes, so you can migrate one virtual machine at a time.

- After you migrate the virtual machines, you must install the latest integration services on the new virtual machines.
- The wizard does not migrate Hyper-V Replica settings or Volume Shadow Copy Service (VSS) tasks. If you are using these with your virtual machines, you must configure them on the new cluster after the migration.

For step-by-step guidance, see Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard.

Additional references

- Migrating Clustered Services and Applications to Windows Server 2012
- Migration Forum
- What's New in Failover Clustering in Windows Server 2012
- Failover Clustering Overview
- Migrating Roles and Features in Windows Server
- Migrate File and Storage Services to Windows Server 2012
- Instructions for completing failover cluster migration scenarios:
  - Migration Between Two Multi-Node Clusters
  - In-Place Migration for a Two-Node Cluster
  - Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard
  - Cluster Migrations Involving New Storage: Mount Points
- High availability for Microsoft Exchange Server 2013:
  - High availability and site resilience documentation" section of High Availability and Site Resilience
  - Upgrade from Exchange 2010 to Exchange 2013
  - Exchange Server Supportability Matrix
- High availability for Microsoft SQL Server 2012:
  - Microsoft SQL Server AlwaysOn Solutions Guide for High Availability and Disaster Recovery (whitepaper)
  - SQL Server 2012 AlwaysOn: Multisite Failover Cluster Instance (whitepaper)
  - High Availability
  - Upgrade a SQL Server Failover Cluster
  - Upgrade a SQL Server Failover Cluster Instance (Deployment)
  - Supported Version and Edition Upgrades

Migration Between Two Multi-Node Clusters

This topic provides step-by-step instructions for migrating clustered services and applications from a multi-node failover cluster running Windows Server 2008, Windows Server 2008 R2, or

Windows Server 2012 to a multimode cluster running Windows Server 2012. (Alternatively, you can perform an in-place migration using a single two-node cluster. For more information, see In-Place Migration for a Two-Node Cluster.) If you plan to migrate highly available Hyper-V virtual machines (by migrating the clustered Virtual Machine role), see the Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard for additional instructions.

Important
Before you begin your migration, review Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012 to confirm that the clustered service or application can be migrated by using the Migrate a Cluster Wizard.

Overview of migration between two multi-node clusters

A migration between two multi-node clusters uses the Migrate a Cluster Wizard, and it has three phases:

1. **Install two or more new servers, run validation, and create a new cluster**. For this phase, while the old cluster continues to run, perform a clean installation of Windows Server 2012 and the Failover Clustering feature on at least two servers. Create the networks that the servers will use, and connect the storage. Make an appropriate number of logical unit numbers (LUNs) or disks accessible to the servers, and do not make those LUNs or disks accessible to any other servers. Next, run the complete set of cluster validation tests to confirm that the hardware and hardware settings can support a failover cluster. Finally, create the new cluster. At this point, you have two clusters.

   For more information, see Steps for creating a failover cluster, later in this topic.
2. **Migrate clustered services and applications to the new cluster, and determine how you will make any existing data available to the new cluster**. When the Migrate a Cluster Wizard completes, all the migrated resources will be offline. Leave them offline at this stage. If the new cluster will reuse old storage, plan how you will make the storage available to the new cluster, but leave the old cluster connected to the storage until you are ready to make the transition.

   For more information, see Steps for migrating clustered services and applications to a failover cluster running Windows Server 2012, later in this topic.
3. **Make the transition from the old cluster to the new cluster**. The first step in the transition is to take the clustered services and applications offline on the old cluster. If the new cluster will use old storage, follow your plan for making LUNs or disks inaccessible to the old cluster and accessible to the new cluster. If the new cluster will use new storage, copy the appropriate folders and data to the storage. Bring the clustered services and applications online on the new cluster. Then verify that failover is working and the clustered services and applications are available.

   For more information, see Steps for completing the transition from the old cluster to the new cluster, later in this topic.

Steps for creating a failover cluster

For information about how to create a Windows Server 2012 failover cluster, see Create a Failover Cluster. To prepare to migrate a clustered service or application to the new failover cluster, make the following preparations.

Preparation

Before you create the failover cluster, prepare storage, and install all required services, applications, and server roles.

1. Prepare storage:
    a. Make an appropriate number of LUNs or disks accessible to the servers, and do not make those LUNs or disks accessible to any other servers. If the new cluster will use old storage, for testing purposes, you can limit the number of LUNs or disks to one or two. If the new cluster will use new storage, make as many disks or LUNs accessible to the new server as you think the cluster will need.

    Note
    We recommend that you keep a small disk or LUN available (unused by clustered services and applications) throughout the life of the cluster, so that you can always run storage validation tests without taking your services and applications offline.

    b. On one of the servers that you plan to include in the cluster, open **Computer Management** from the **Start** screen, and then click **Disk Management** in the console tree. In Disk Management, confirm that the intended cluster disks are visible.
    c. Check the format of any exposed volume or LUN. We recommend that you use NTFS for the format. (For a disk witness, you must use NTFS.)
    d. If you are using new storage and your disk configuration uses mount points, review Cluster Migrations Involving New Storage: Mount Points to identify any additional steps you will need to perform.
2. Install services, applications, and server roles:
    - After you install the Failover Clustering feature on all nodes, install any needed services, applications, and server roles. For example, if you plan to migrate clustered Windows Internet Name Service (WINS) to the new cluster, install the WINS Server feature by using Server Manager.
    - If you plan to migrate highly available virtual machines, add the Hyper-V role to the server. You also must merge or discard all shadow copies on the volumes that contain the virtual machines. For step-by-step instructions for migrating highly available virtual machines, see Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard.
    - If you are migrating a Generic Application, Generic Script, or Generic Service resource, you must confirm that any associated application is compatible with Windows Server 2012. You also must confirm that any associated service exists in Windows Server 2012

and has the same name that it had in the old cluster. Test the application or service (separately, not as part of a cluster) to confirm that it runs as expected.

After you create the failover cluster

After you create the cluster, ensure that your firewall is configured appropriately. For example, if you are using Windows Firewall, and you will be sharing folders and files, use your preferred Windows Firewall interface to allow the exception for **Remote Volume Management**.

Steps for migrating clustered services and applications to a failover cluster running Windows Server 2012

Use the following instructions to migrate clustered services and applications from your old cluster to your new cluster. The Migrate a Cluster Wizard leaves most of the migrated resources offline so that you can perform additional steps before you bring them online.

Note
To migrate a clustered service or application by using the Migrate a Cluster Wizard, you must be a local administrator on the destination failover cluster and on the cluster or cluster node from which you are migrating.

To migrate data and clustered services or applications from an existing cluster to a new cluster

1. If the new cluster uses old storage, plan how you will make LUNs or disks inaccessible to the old cluster and accessible to the new cluster (but do not make changes yet). If you plan to use new storage with the migrated services or applications, before you run the Migrate a Cluster Wizard, make the storage is available to the new cluster – that is, that the volumes have been added to the new cluster and that they are online. This enables the wizard to update storage settings during migration.
2. From the **Start** screen or from **Server Manager** (**Tools**), open **Failover Cluster Manager**.
3. In the console tree, if the cluster that you created is not displayed, right-click **Failover Cluster Manager**, click **Connect to Cluster**, and then select the cluster that you want to configure.
4. In the console tree, expand the cluster that you created to see the items underneath it.
5. If the clustered servers are connected to a network that is not to be used for cluster communications (for example, a network intended only for iSCSI), then under **Networks**, right-click that network, click **Properties**, and then click **Do not allow cluster network communication on this network**. Click **OK**.
6. In the console tree, select the cluster.
7. Under **Configure**, click **Migrate services and applications**.

The Migrate a Cluster Wizard opens.

8. Read the Welcome page, and then click **Next**.
9. Specify the name or IP address of the cluster or cluster node from which you want to migrate services and applications, and then click **Next**.
10. The **Select Services and Applications** page lists the clustered services and applications that can be migrated from the old cluster. The list does not contain any service or application that is not eligible for migration. Click **View Report** for details. Then select each service and application that you want to migrate to the new cluster, and click **Next**.

    Important
    We recommend that you read the report, which explains whether each resource is eligible for migration. (The wizard also provides a report after it finishes, which describes any additional steps that might be needed before you bring the migrated resource groups online.)

    If storage is available on the new cluster, the **Specify Storage for Migration** page appears, giving you the option to migrate to new storage. If storage is not available on the new cluster, the wizard retains existing storage settings and does not display the page.

    Note
    Not all clustered roles can be migrated to new storage. For example, the wizard cannot be used to migrate highly available virtual machines (the Virtual Machine role) to new storage. For step-by-step instructions for migrating highly available virtual machines, see Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard.
11. If you want to use new storage for a service or application:
    a. On the **Specify Storage for Migration** page, select the cluster disk that you want to migrate to new storage, and then click **Select Storage**.
    b. In the **Select Storage for Resource Group** dialog box, under **Available Storage in New Cluster**, select the cluster disk that you want the service or application to use in the new cluster, and then click **OK**.
    c. Repeat these steps for each cluster disk that you want to migrate to new storage. Then click **Next**.

    Important
    The Migrate a Cluster Wizard does not move existing data and folders to the new storage. You must copy the folders and data manually.
12. Follow the instructions in the wizard to perform the migration. From the **Summary** page, we recommend that you read the Cluster Migration Report, which contains important information about any additional steps that you might need to complete before you bring the migrated services and applications online. For example, if you have not already installed needed applications on the new cluster node, you might need to install them.

After the wizard completes, most migrated resources will be offline. Leave them offline at this stage.

Steps for completing the transition from the old cluster to the new cluster

You must perform the following steps to complete the transition to the new cluster running Windows Server 2012. After you complete the transition, verify that failover is working correctly for the migrated services and applications and that the services are available.

To complete the transition from the old cluster to the new cluster

1. Prepare for clients to experience downtime, probably briefly.
2. On the old cluster, take each role and resource that was migrated offline.
3. Complete the transition for the storage:
   - If the new cluster will use old storage, follow your plan for making LUNs or disks inaccessible to the old cluster and accessible to the new cluster.
   - If the new cluster will use new storage, copy the appropriate folders and data to the storage. As needed for disk access on the old cluster, bring individual disk resources online on that cluster. (Keep other resources offline, to ensure that clients cannot change data on the disks in storage.) On the new cluster, use Disk Management to confirm that the appropriate LUNs or disks are visible to the new cluster and not visible to any other servers.
4. If the new cluster uses mount points, adjust the mount points as needed, and make each disk resource that uses a mount point dependent on the resource of the disk that hosts the mount point. For more information about mount points, see Cluster Migrations Involving New Storage: Mount Points.
5. Bring the migrated services or applications online on the new cluster.
6. If you migrated highly available virtual machines, install the latest integration services on each virtual machine. You might need to restart the virtual machine to complete the installation.

To verify that the migrated service or application is performing as expected and can fail over successfully

1. Verify that you can access the workload that was migrated. For example, can you connect to a highly available file server after it is migrated? Can you see the data that the server stores?
2. In the console tree of **Failover Cluster Manager**, click the failover cluster on which the service or application is running.
3. Expand **Services and Applications**, and then click a migrated service or application that you want to test.
4. Under **Actions** (on the right), click **Move this service or application to another node**, and then click an available choice of node. When prompted, confirm your choice.

You can observe the status changes in the center pane of the snap-in as the clustered service or application is moved.

5. If there are any issues with failover, review the following:
   - View events in Failover Cluster Manager. To do this, in the console tree, right-click **Cluster Events**, and then click **Query**. In the **Cluster Events Filter** dialog box, select the criteria for the events that you want to display, or to return to the default criteria, click the **Reset** button. Click **OK**. To sort events, click a heading, for example, **Level** or **Date and Time**.
   - Confirm that necessary services, applications, or server roles are installed on all nodes. Confirm that services or applications are compatible with Windows Server 2012 and run as expected.
   - If you used old storage for the new cluster, rerun the Validate a Cluster Configuration Wizard to confirm the validation results for all LUNs or disks in the storage.
   - If you migrated highly available virtual machines, verify the status of the virtual machines in Hyper-V Manager, and ensure that you can connect to the virtual machines by using Remote Desktop or Virtual Machine Connection.
   - Review migrated resource settings and dependencies. If you are using new storage that includes disks that use mount points, see Cluster Migrations Involving New Storage: Mount Points.
   - If you migrated one or more Network Name resources with the Kerberos protocol enabled, confirm that the following permissions change was made in **Active Directory Users and Computers** on a domain controller. In the computer accounts (computer objects) of your Kerberos protocol-enabled Network Name resources, **Full Control** must be assigned to the computer account for the failover cluster.

Note

The Migrate a Cluster Wizard does not migrate Volume Shadow Copy Service (VSS) tasks, Hyper-V Replica Broker settings, Task Scheduler tasks, and Cluster-Aware Updating (CAU) settings. If you were using any of these features on the old cluster, you will need to configure them on the new cluster.

Related references

In-Place Migration for a Two-Node Cluster

Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard

Cluster Migrations Involving New Storage: Mount Points

Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012

In-Place Migration for a Two-Node Cluster

This topic provides an overview and steps for upgrading an existing failover cluster to Windows Server 2012 when you have only two servers - that is, for performing an *in-place migration*.

Important

Before you begin the migration, confirm that the clustered service or application that you want to migrate can be migrated by using the Migrate a Cluster Wizard, as described in Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012, and note any preparation or follow-up steps that are required for the service type that is being migrated.

Note

For an alternative approach to failover cluster migration, see Migration Between Two Multi-Node Clusters.

Overview of an in-place migration for a two-node cluster

This migration uses the Migrate a Cluster Wizard, and it has four phases:

1. **Evict one node, install Windows Server 2012, and create a single-node failover cluster**. For this phase, allow one existing server to continue running Windows Server 2008 R2 or Windows Server 2008 and the Cluster service while you begin the migration process. Evict the other server from the old cluster, and then perform a clean installation of Windows Server 2012 and the Failover Clustering feature on it. On that server, run all tests that the Validate a Configuration Wizard will run. The wizard will recognize that this is a single node without storage and limit the tests that it runs. Tests that require two nodes (for example, tests that compare the nodes or that simulate failover) will not run.

   Note that the tests that you run at this stage do not provide complete information about whether the storage will work in a cluster running Windows Server 2012. As described later in this section, you will run the Validate a Configuration Wizard later with all tests included.

   For steps to complete this phase, see Steps for evicting a node and creating a new single-node Windows Server 2012 failover cluster, later in this topic.
2. **Migrate clustered services and applications to the new single-node cluster**. Run the Migrate a Cluster Wizard, but leave the migrated resources offline on the new cluster.

   For steps to complete this phase, see Steps for migrating clustered services and applications to the new cluster, later in this topic.
3. **Make existing data available to the new cluster, and bring the cluster online**. Confirm that the settings for the migrated services and applications are correct. Next, take the migrated services and applications in the old cluster offline. If the new cluster will use new storage, copy the folders and data to appropriate LUNs or disks in the new storage, and make sure that those LUNs or disks are visible to the new cluster (and not visible to any other servers). If the new cluster will use the old storage, make the appropriate disks or LUNs accessible to the new cluster. Bring the services and applications in the new cluster online, and make sure that the resources are functioning and can access the storage.

   For steps to complete this phase, see Steps for making existing data available to the new cluster and bringing it online, later in this topic.

4. **Add the second node to the new cluster**. Destroy the old cluster and, on that server, install Windows Server 2012 and the Failover Clustering feature. Connect that server to the networks and storage that are used by the new cluster. If the appropriate disks or LUNs are not already accessible to both servers, make them accessible. Run the Validate a Configuration Wizard, specifying both servers, and confirm that all tests pass. Finally, add the second server to the new cluster.

   For information about steps for this phase, see Steps for adding the second node to the new cluster, later in this topic.

Steps for evicting a node and creating a new single-node Windows Server 2012 failover cluster

You must complete the following steps to create a single-node Windows Server 2012 failover cluster:

Step 1: Evict one node from the old cluster, and perform a clean installation of Windows Server 2012

To begin, you must evict one node from the old cluster, and perform a clean installation of Windows Server 2012 on that node.

Before you evict a node from a cluster

- For each clustered service or application that you plan to migrate, verify that there are no special requirements or procedures for removing or evicting a node from the cluster. You can evict a node from a clustered file server or a cluster with the Hyper-V role with no special preparation. However, you might need to uncluster some services or applications before you evict a node.
- To prevent any loss of application data when the node is evicted, shut down all services and applications on the cluster before you evict the node.

To evict a node from a cluster

1. From the Start screen, open **Failover Cluster Manager**.
2. In the console tree, expand the cluster, expand **Nodes**, and then click the node that you want to evict to select it.
3. Right-click the node, click **More Actions**, and then click **Evict**.

Step 2: Create a single-node cluster and install other needed software

For information about how to create a Windows Server 2012 failover cluster, see Create a Failover Cluster. To prepare to migrate a clustered service or application to the new failover cluster, make the following preparations.

Preparation

Before you create the failover cluster, prepare storage, and install all required services, applications, and server roles.

1. Prepare storage:
    a. Make an appropriate number of LUNs or disks accessible to the server, and do not make those LUNs or disks accessible to any other servers. If the new cluster will use old storage, for testing purposes, you can limit the number of LUNs or disks to one or two. If the new cluster will use new storage, make as many disks or LUNs accessible to the new server as you think the cluster will need.

    Note
    We recommend that you keep a small disk or LUN available (unused by clustered services and applications) throughout the life of the cluster, so that you can always run storage validation tests without taking your services and applications offline.

    b. On the server, open **Computer Management** from the **Start** screen, and then click **Disk Management** in the console tree. In Disk Management, confirm that the intended cluster disks are visible.
    c. Check the format of any exposed volume or LUN. We recommend that you use NTFS for the format. (For a disk witness, you must use NTFS.)
    d. If you are using new storage and your disk configuration uses mount points, review Cluster Migrations Involving New Storage: Mount Points to identify any additional steps you will need to perform.
2. Install services, applications, and server roles:
    - After you install the Failover Clustering feature on the server, install any needed services, applications, and server roles. For example, if you plan to migrate clustered Windows Internet Name Service (WINS) to the new cluster, install the WINS Server feature by using Server Manager.
    - If you plan to migrate highly available virtual machines, add the Hyper-V role and install the latest Hyper-V integration components. You also must merge or discard all shadow copies on the volumes that contain the virtual machines. For step-by-step instructions for migrating highly available virtual machines, see Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard.
    - If you are migrating a Generic Application, Generic Script, or Generic Service resource, you must confirm that any associated application is compatible with Windows Server 2012. You also must confirm that any associated service exists in Windows Server 2012

and has the same name that it had in the old cluster. Test the application or service (separately, not as part of a cluster) to confirm that it runs as expected.

After you create the failover cluster

After you create the cluster, ensure that your firewall is configured appropriately. For example, if you are using Windows Firewall, and you will be sharing folders and files, use your preferred Windows Firewall interface to allow the exception for **Remote Volume Management**.

Steps for migrating clustered services and applications to the new cluster

Use the following instructions to migrate clustered services and applications from your old one-node cluster to your new one-node cluster. The Migrate a Cluster Wizard leaves most of the migrated resources offline so that you can perform additional steps before you bring them online.

Note

To migrate a clustered service or application by using the Migrate a Cluster Wizard, you must be a local administrator on the destination failover cluster and on the cluster or cluster node from which you are migrating.

To migrate clustered services and applications from the old cluster to the new cluster

1. If you want to migrate to new storage, before you run the Migrate a Cluster Wizard, ensure that the storage is available to the new cluster – that is, that the volumes have been added to the new cluster and that they are online.
2. From the **Start** screen or **Server Manager** (**Tools**), open **Failover Cluster Manager**.
3. In the console tree, if the cluster that you created is not displayed, right-click **Failover Cluster Manager**, click **Connect to Cluster**, and then select the cluster that you want to configure.
4. In the console tree, expand the cluster that you created to see the items underneath it.
5. If the clustered server is connected to a network that is not to be used for cluster communications (for example, a network intended only for iSCSI), then under **Networks**, right-click that network, click **Properties**, and then click **Do not allow cluster network communication on this network**. Click **OK**.
6. In the console tree, select the cluster.
7. Under **Configure**, click **Migrate services and applications**.
8. Read the first page of the Migrate a Cluster Wizard, and then click **Next**.
9. Specify the name or IP address of the cluster or cluster node from which you want to migrate services and applications, and then click **Next**.
10. The **Select Services and Applications** page lists the clustered services and applications that can be migrated from the old cluster. The list does not contain any service or application that is not eligible for migration. Click **View Report** for details. Then select each service and application that you want to migrate to the new cluster, and click

Next.

Important
We recommend that you read the report, which explains whether each resource is eligible for migration. (The wizard also provides a report after it finishes, which describes any additional steps that might be needed before you bring the migrated resource groups online.)

If storage is available on the new cluster, the **Specify Storage for Migration** page appears, giving you the option to migrate to new storage. If storage is not available on the new cluster, the wizard automatically retains existing storage settings and does not display the page.

11. If you want to use new storage for a service or application:
    a. On the **Specify Storage for Migration** page, select the cluster disk that you want to migrate to new storage, and then click **Select Storage**.
    b. In the **Select Storage for Resource Group** dialog box, under **Available Storage in New Cluster**, select the cluster disk that you want the service or application to use in the new cluster, and then click **OK**.
    c. Repeat these steps for each cluster disk that you want to migrate to new storage. Then click **Next**.

    Important
    The Migrate a Cluster Wizard does not move existing folders and data to the new storage. You must copy the folders and data manually.
12. Follow the instructions in the wizard to perform the migration. From the **Summary** page, we recommend that you read the Cluster Migration Report, which contains important information about any additional steps that you might need to complete before you bring the migrated services and applications online. For example, if you have not already installed needed applications on the new cluster node, you might need to install them.

When the wizard completes, most migrated resources will be offline. Leave them offline at this stage.

Steps for making existing data available to the new cluster and bringing it online

Use the following procedure to make existing data available to the new cluster and bring it online.

To make existing data available to the new cluster and bring it online

1. Confirm that the settings for the migrated services and applications appear correct.
2. Prepare for clients to experience downtime, probably briefly.
3. On the old cluster, take each clustered service or application that you migrated offline.
4. Complete the transition for the storage:
    - If the new cluster will use old storage, follow your plan for making LUNs or disks

inaccessible to the old cluster and accessible to the new cluster.

- If the new cluster will use new storage, copy the appropriate folders and data to the storage. As needed for disk access on the old cluster, bring individual disk resources online on that cluster. (Keep other resources offline to ensure that clients cannot change data on the disks in storage.) Then, on the new cluster node, use Disk Management to confirm that the appropriate LUNs or disks are visible to the new cluster and not visible to any other servers.

5. If the new cluster uses mount points, adjust the mount points as needed, and make each disk resource that uses a mount point dependent on the resource of the disk that hosts the mount point. For more information about mount points, see Cluster Migrations Involving New Storage: Mount Points.
6. Bring the migrated services or applications online on the new cluster.

Steps for adding the second node to the new cluster

Use the following instructions to prepare the second node and then add it to the new cluster. As part of this process, you will run validation tests that include both servers.

To add the second node to the new cluster

1. On the new cluster, confirm that the migrated services or applications are functioning and that clients can connect to them.
2. On the old cluster (the server that is running Windows Server 2008 R2 or Windows Server 2008), delete the migrated services and applications, and then destroy the old cluster:
   a. From the Start screen, open **Failover Cluster Manager**.
   b. Remove services and applications that were migrated. In Failover Cluster Manager, expand the cluster, and expand **Services and applications**. To delete a service or application, right-click the item, and click **Delete**.
   c. To destroy the cluster, right-click the cluster, click **More Actions**, and then click **Destroy Cluster**.
3. On the same server, perform a clean installation of Windows Server 2012.
4. Add the Failover Clustering feature in the same way that you added it to the other server, and install any needed services, applications, and server roles.
5. Connect the newly installed server to the same networks and storage that the existing failover cluster node is connected to.
6. Identify the disks or LUNs that are exposed to the new one-node failover cluster, and expose them to the newly installed server also.

   We recommend that you keep a small disk or LUN accessible to both nodes, and unused by clustered services and applications, throughout the life of the cluster. With this LUN, you can always run storage validation tests without taking your services and applications

offline.

7. On either server running Windows Server 2012, open Failover Cluster Manager from the Start screen.
8. Confirm that **Failover Cluster Manager** is selected, and then, in the center pane, under **Management**, click **Validate a Configuration**.

   Follow the instructions in the wizard, but this time, be sure to specify both servers (not just the existing cluster name) and specify that you want to run all tests. Then, run the tests. Because two nodes are now being tested, a more complete set of tests runs, which takes longer than testing one node.

   Important

   If any clustered service or application is using a disk when you start the wizard, the wizard asks whether to take that clustered service or application offline for testing. If you choose to take a clustered service or application offline, it remains offline until the tests finish.
9. The **Summary** page appears after the tests run. To view Help topics to help you interpret the results, click **More about cluster validation tests**.
10. While still on the **Summary** page, click **View Report** and read the test results.

    To view the results of the tests after you close the wizard, see

    <*SystemRoot*>\Cluster\Reports\Validation Report <*date and time*>.mht

    where <*SystemRoot*> is the folder in which the operating system is installed (for example, C:\Windows\).
11. As necessary, make changes in the configuration and rerun the tests.

    For more information about failover cluster validation tests, see Validate Hardware for a Failover Cluster.
12. If the new cluster is not displayed, in the console tree, right-click **Failover Cluster Manager**, click **Connect to a Cluster**, and then select the new cluster.
13. In the console tree, select the one-node cluster, and then in the **Actions** pane, click **Add Node**.
14. Follow the instructions in the wizard to specify the server that you want to add to the cluster. On the **Summary** page, click **View Report** to review the tasks that the wizard performed.
15. On the **Summary** page, click **View Report** if you want to review the tasks that the wizard performed. Or view the report after the wizard closes in the <*SystemRoot*>\Cluster\Reports\ folder.

    Note

    After you close the wizard, in the center pane, you might see a warning about "Node Majority." You will correct this issue in the next few steps.
16. In the console tree, expand **Storage**. Check to see if all the disks that you want to make available to the new cluster are shown, either in one of the clustered services or applications or in **Available Storage**.

In most cases, you need at least one disk in **Available Storage** for the next step (specifying a witness disk). If you need to add a disk, in the **Actions** pane, click **Add Disk** and follow the steps in the wizard.

Before you can add a disk to storage, it must be accessible from both nodes in the cluster. To be used for a witness disk, a disk can be a relatively small, but must be at least 512 MB.

17. In the console tree, right-click the new cluster, click **More Actions**, and then click **Configure Cluster Quorum Settings**.
18. Follow the instructions in the wizard to select the most appropriate quorum setting for your needs. In most cases, this is the **Node Majority** quorum configuration, which requires that you specify an appropriate disk (from **Available Storage**) for the witness disk. For more information about quorum settings in Windows Server 2012, see Configure and Manage the Quorum in a Windows Server 2012 Failover Cluster.
19. Expand **Services and Applications**, and then click a migrated service or application that you want to test.
20. To perform a basic test of failover for the migrated service or application, under **Actions** (on the right), click **Move this service or application to another node**, and then click an available choice of node. When prompted, confirm your choice.

    You can observe the status changes in the center pane of Failover Cluster Manager as the clustered service or application is moved. If there are any issues with failover, review the following:

    - View events in Failover Cluster Manager. To do this, in the console tree, right-click **Cluster Events**, and then click **Query**. In the **Cluster Events Filter** dialog box, select the criteria for the events that you want to display, or, to return to the default criteria, click the **Reset** button. Click **OK**. To sort events, click a heading, for example, **Level** or **Date and Time**.
    - Confirm that necessary services, applications, or server roles are installed on all nodes. Confirm that services or applications are compatible with Windows Server 2012 and run as expected.
    - Review migrated resource settings and dependencies. If you are using new storage that includes disks that use mount points, see Cluster Migrations Involving New Storage: Mount Points for more information.
    - If you migrated one or more Network Name resources with the Kerberos protocol enabled, confirm that the following permissions change was made in **Active Directory Users and Computers** on a domain controller. In the computer accounts (computer objects) of your Kerberos protocol-enabled Network Name resources, **Full Control** must be assigned to the computer account for the failover cluster.

Related references

Migration Between Two Multi-Node Clusters

Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard

Cluster Migrations Involving New Storage: Mount Points

Migration Paths for Migrating to a Failover Cluster Running Windows Server 2012

Migration of Highly Available Virtual Machines Using the Migrate a Cluster Wizard

This topic provides a process overview and step-by-step instructions for migrating a Hyper-V highly available virtual machine (HAVM) from a failover cluster running Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012 to a failover cluster running Windows Server 2012 by using the Migrate a Cluster Wizard in Failover Cluster Manager. This is accomplished by migrating the clustered Virtual Machine role from the old cluster to the new cluster. The migrated HAVMs use the same storage that they used in the old cluster. The wizard cannot migrate virtual machines to new storage.

You can use either of the two migration scenarios to migrate an HAVM: migrate between two multi-node clusters or perform an in-place migration.

Note

You can also use this method to migrate HAVMs to a failover cluster running Windows Server 2012 R2 from a failover cluster running Windows Server 2008 R2 with Service Pack 1 (SP1), Windows Server 2012, or Windows Server 2012 R2. In Windows Server 2012 R2, the name of the Migrate a Cluster Wizard was changed to **Copy Cluster Roles**, and the wizard is opened by using the **Copy Roles** action. For consistency with labeling in Windows Server 2012 R2, the items being migrated are referred to as clustered roles instead of clustered services and applications. However, the steps for performing the wizard-based migration are the same.

Supported operating systems

The Migrate a Cluster Wizard in Windows Server 2012 can migrate highly available virtual machines running on any of the following Windows Server operating system versions to Windows Server 2012:

- Windows Server 2008 with Server Pack 2 (SP2)
- Windows Server 2008 R2 with Service Pack 1 (SP1)
- Windows Server 2012

Before you migrate any clustered role or service, you should install the latest operating system updates on all nodes in the old and new clusters.

Overview of the migration process

To migrate a highly available virtual machine from one failover cluster to another, you use the Migrate a Cluster Wizard in Failover Cluster Manager to migrate the clustered Virtual Machine

role. After you select the Virtual Machine role, you select the role instances (virtual machines) that you want to migrate.

Note

Be aware that if you migrate one virtual machine that resides on a Cluster Shared Volume (CSV) volume, the wizard migrates all virtual machines on that volume. The wizard allows you to update virtual network settings on the virtual machines to the network settings on the new cluster. You cannot use the wizard to migrate virtual machines to new storage. This restriction does not apply if you are migrating a Scale-out File Server cluster. A Scale-out File Server cluster does not use CSV volumes, so you can migrate one virtual machine at a time.

To prepare the virtual machines for the migration, you must merge or discard all shadow copies on the volumes that contain the virtual machines. To prepare the new cluster for the migration, you must add the Hyper-V role to the cluster nodes and configure storage and virtual networks on the cluster.

After the migration, you will need to take the virtual machines offline on the old cluster, follow your plans to mask the volumes that contain the virtual machines to the old cluster and unmask the volumes to the new cluster, and then bring the virtual machines online on the new cluster. After you bring the virtual machines on the new cluster online, you must also install the latest integration services on the virtual machines.

To schedule local backups of the virtual machines, you will need to configure Volume Shadow Copy Service (VSS) tasks, which the wizard does not migrate. If you migrated from a Windows Server 2012 failover cluster, you will also need to configure Hyper-V Replica Manager settings and Cluster-Aware Updates (CAU) if you were using them; those settings also do not migrate.

Tip

For a step-by-step walk-through, with screenshots, of migrating a Hyper-V host cluster from Windows Server 2008 R2 to Windows Server 2012 by using the Migrate a Cluster Wizard, see the blog entry How to Move Highly Available (Clustered) VMs to Windows Server 2012 with the Cluster Migration Wizard on MSDN.

Impact of the migration

There will be a brief service interruption during the migration. To minimize the effects on users, schedule the migration during a maintenance window. We also recommend that you pretest and verify the migration before you migrate the virtual machines in your production environment.

Note

You cannot use live migration to migrate a highly available virtual machine to a new failover cluster.

Required permissions

To migrate a clustered service or application by using the Migrate a Cluster Wizard, you must be a local administrator on the destination failover cluster and on the cluster or cluster node from which you are migrating.

Prepare to migrate

While you prepare to migrate the virtual machines, the virtual machines can remain online and continue providing service.

To prepare virtual machine storage for migration

1. Before you begin working with shadow copies, you should back up all volumes that are attached to the virtual machine(s).
2. Merge or discard all shadow copies for the volumes that store the virtual machines.
3. Ensure that no virtual machines that you do not want to migrate share a CSV volume with virtual machines that you plan to migrate. If you migrate one virtual machine on a CSV volume, the Migrate a Cluster Wizard migrates all virtual machines on that volume.

To prepare the old failover cluster for the migration

- Install the latest operating system updates on each cluster node. A Windows Server 2008 failover cluster must be running Windows Server 2008 SP2 or later. A Windows Server 2008 R2 failover cluster must be running Windows Server 2008 R2 SP1 or later.

To prepare the new failover cluster for the migration

1. To create the new failover cluster in Windows Server 2012, use Failover Cluster Manager, or use the **New-Cluster** cmdlet in Windows PowerShell (for information, see New-Cluster). For a detailed description of the steps for preparing a new failover cluster, see Migration Between Two Multi-Node Clusters or In-Place Migration for a Two-Node Cluster.
2. Add the Hyper-V role to each cluster node.
3. Configure virtual switches in Hyper-V.
4. If you are migrating from a Windows Server 2008 or Windows Server 2008 R2 failover cluster, check with your hardware vendor to ensure that the existing storage is supported in Windows Server 2012.

   Note
   You do not need to configure storage for the virtual machines on the new cluster before you run the wizard. The Migrate a Cluster Wizard will migrate existing storage settings to the new cluster. After the wizard completes, you will mask the storage from the old cluster and then unmask the storage on the new cluster.
5. Install the latest operating system updates on each cluster node.

Migrate the highly available virtual machines to the new failover cluster

You will use the Migrate a Cluster Wizard in Failover Cluster Manager to migrate the virtual machines to the new failover cluster. This is done by migrating the clustered Virtual Machine role. After the virtual machines are migrated, you must make the storage available to the new cluster before you bring the virtual machines online.

Before you migrate the highly available virtual machines

- Ensure that the virtual switches are configured on host operating systems in the new cluster.
- Prepare for a brief service interruption on the workloads running on the virtual machines. Live migration is not supported during migration of a virtual machine to a new host cluster.

To migrate the virtual machines to a new failover cluster

1. Log on to any node in the new failover cluster with an Administrator account.
2. From the **Start** screen or **Server Manager (Tools)**, open **Failover Cluster Manager**.
3. In the console tree, expand **Failover Cluster Manager** in the console tree, and select the cluster that you want to migrate the virtual machines to.

   If the new cluster is not displayed, right-click **Failover Cluster Manager**, click **Connect to Cluster**, and then select the cluster to which you want to migrate the virtual machines.
4. With the destination cluster selected, click **Migrate Roles**.

   The Migrate a Cluster Wizard opens.
5. Review the instructions on the **Before You Begin** page, and click **Next**.
6. On the **Specify Old Cluster** page, enter the name or IP address of the source cluster, or use the **Browse** button to find the cluster, and then click **Next**.

   The wizard connects to the cluster and displays the roles and features that can be migrated. For virtual machines, each virtual machine “role” is listed under the cluster shared volume that stores the virtual machine.
7. On the **Select Services and Applications** page, click **View Reports**, and review the resources that can and cannot be migrated. Note that **Available Storage** and **Cluster Group** are never available for migration, and always have a Failed result. For all other resources, review any Warning results to identify any and resolve any issues that might prevent a successful migration. Then close the report.
8. On the **Select Services and Applications** page, select each highly available virtual machine that you want to migrate, and then click **Next**. If a volume stores more than one virtual machine, and you select any virtual machine on that volume, the wizard will migrate all virtual machines on that volume.
9. On the **Customize Virtual Machine Networks** page, optionally select a select virtual switch for the virtual machines to use on the destination host cluster. If you do not select

a virtual switch, the wizard retains the default switch that is selected automatically the first time the virtual machine starts on its new host.

10. On the **Configuration** page, review your settings. Then click **Next** to start the migration.
11. After the migration completes, review the Post-Migration Report to verify that the virtual machines were migrated. Then click **Finish**.
12. In Failover Cluster Manager, verify the status of the migrated virtual machines and the related resources:
    a. In the console tree, click the name of the new failover cluster, and then click **Roles**. You should see the migrated virtual machine (roles) in the **Roles** pane. The virtual machines will be turned off.
    b. Click a virtual machine to display the associated resources at the bottom of the window. For a newly migrated virtual machine, the resources have been registered but are not online.

    Before you can start the virtual machines, you must remap the storage to the new cluster and then bring the storage online.

To complete the migration

1. Prepare for clients to experience downtime, probably briefly.
2. Shut down the old cluster to ensure that no one will attempt to start the virtual machine during migration and no connections will be made from storage.

   Caution
   At no time should a virtual machine be running on both the old cluster and the new cluster. A virtual machine that runs on both the old cluster and the new cluster at the same time might become corrupted. You can run a virtual machine on the old cluster while you migrate it to a new cluster with no problems; the virtual machine on the new cluster is created in a Stopped state. However, to avoid corruption, it is important that you do not turn on the virtual machine on the new cluster until after you stop the virtual machine on the old cluster.
3. To complete the transition for the storage:
   a. Make the CSV volume that stores the virtual machines inaccessible to the old cluster, and then make them accessible to the new cluster.
   b. After you move the storage to the new cluster, in Disk Management, bring the CSV volume and Virtual Machine Configuration resource for each virtual machine online.
4. At this point, you should be able to start the virtual machines. To start the virtual machines in Failover Cluster Manager, display and select the virtual machine role, and then click **Start Role**.
5. Install the latest integration services on the new virtual machines. You might need to restart the virtual machine to complete the integration services update.

Note

The Migrate a Cluster Wizard does not migrate Volume Shadow Copy Service (VSS) tasks, Hyper-V Replica Broker settings, Task Scheduler settings, and Cluster Aware Updating (CAU) settings. If you were using any of these features on the old cluster, you will need to configure them on the new cluster.

Verify a successful migration

After you complete the migration, you should bring the virtual machine online, make sure the services that the virtual machine was providing on the old cluster are still available and working as expected, and test failover for the virtual machine on the new cluster. Verify that you can connect to the virtual machines by using Remote Desktop or Virtual Machine Connection. For detailed steps for verifying a successful role migration and testing failover, see Migration Between Two Multi-Node Clusters.

Related references

How to Move Highly Available (Clustered) VMs to Windows Server 2012 with the Cluster Migration Wizard

Migration Between Two Multi-Node Clusters

In-Place Migration for a Two-Node Cluster

Cluster Migrations Involving New Storage: Mount Points

This topic describes considerations for configuring mount points during a migration to a failover cluster running Windows Server 2012 R2 or Windows Server 2012 when the destination cluster will use new storage after the migration.

Caution

If you want to use new storage, you must copy or move the data or folders (including shared folder settings) during a migration. The wizard for migrating clustered resources does not copy data from one shared storage location to another.

The Migrate a Cluster Wizard does not migrate mount point information (that is, information about hard disk drives that do not use drive letters, but are mounted instead in a folder on another hard disk drive). However, the wizard can migrate Physical Disk Resource settings to and from disks that use mount points. The wizard also does not configure the necessary dependency between the resources for mounted disks and the resource for a host disk (the disk on which the other disks are mounted). You must configure those dependencies after the wizard completes.

When you work with new storage for your cluster migration, you have some flexibility in the order in which you complete the tasks. You must create the mount points, run the Migrate a Cluster Wizard, copy the data to the new storage, and confirm the disk letters and mount points for the

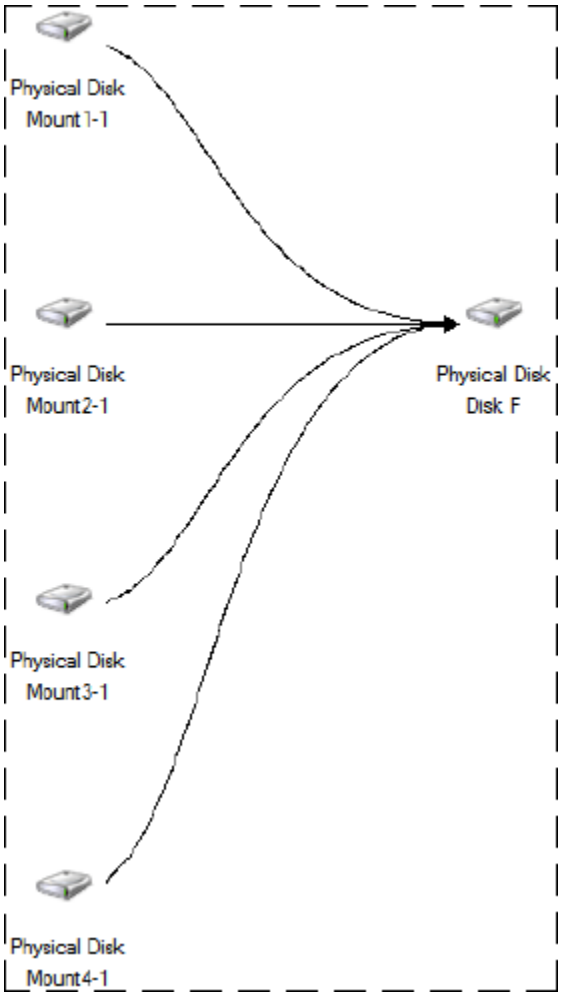
new storage. After completing those tasks, configure the disk resource dependencies in Failover Cluster Manager.

A useful way to keep track of disks in the new storage is to give them labels that indicate your intended mount point configuration. For example, in the new storage, when you are mounting a new disk in a folder called **Mount1-1** on another disk, you can also label the mounted disk as **Mount1-1**. (This assumes that the label **Mount1-1** is not already in use in the old storage.) When you run the Migrate a Cluster Wizard, and you need to specify that disk for a particular migrated resource, you can select the disk labeled **Mount1-1** from the list. After the wizard completes, you can return to Failover Cluster Manager to configure the disk resource for **Mount1-1** so that it is dependent on the appropriate resource - for example, the resource for disk **F**. Similarly, you would configure the disk resources for all other disks mounted on disk F so that they depended on the disk resource for disk F.

After you run the wizard and fully configure the mounted disk, your last task is to configure the disk dependencies in Failover Cluster Manager. For each disk resource for a mounted hard disk drive, open the Properties sheet and, on the **Dependencies** tab, specify a dependency on the disk resource for the host drive (where the mounted drives reside). This ensures that the Cluster service brings the host drive online first, followed by the drives that are dependent on it.

After you configure the dependencies, you can view a dependency report. To view a dependency report, click the service or application in Failover Cluster Manager, and then, under **Actions**, click **Show Dependency Report**. The following illustration shows four mount points that are configured with the correct dependencies on the disk on which they are mounted:

Four mount points with dependencies configured



Additional references

Migrate Cluster Roles to Windows Server 2012 R2

Migrating Clustered Services and Applications to Windows Server 2012

Additional References

- Overview of failover clusters:
  - What's New in Failover Clustering in Windows Server 2012
  - Failover Clustering Overview
  - Failover Clustering Hardware Requirements and Storage Options
  - Validate Hardware for a Failover Cluster
- Community resources:

    - Windows Server Migration forum
    - Clustering Forum for Windows Server 2012
- Deploying failover clusters:
    - Create a Failover Cluster
    - Deploy a Hyper-V Cluster
- Cluster configuration:
    - Configure and Manage the Quorum in a Windows Server 2012 Failover Cluster
    - Use Cluster Shared Volumes in a Failover Cluster