THALES

Windows Integration Guide

FOR CTE V7.6.0

Using CTE with Microsoft SQL AlwaysOn and SQL File Tables

This section discusses using CTE with Microsoft SQL AlwaysOn and SQL File Tables. It contains the following topics:

- Using CTE (Standard Policy) with Microsoft SQL
- Using CTE (LDT Policy) with Microsoft SQL

Using CTE with Microsoft SQL

- Using CTE with SQL
- Using CTE with SQL FileTables
- Installing CTE on Microsoft SQL AlwaysOn
- Data Transformation (Encryption in place)
- Copy/Restore
- SQL Server Policy Tuning

Using CTE with SQL

You must stop the SQL service before guarding the SQL DB. When this occurs, the SQL Server replication may become unsynchronized. When restarting, it may take a brief period of time for the SQL Server replication to resynchronize with the other node. The SQL Server issues a warning against any attempted failovers during that brief period.

Note

Minimizing the duration for which the SQL Server service is stopped is beneficial for reducing the resynchronization period.

Using CTE with SQL FileTables

SQL FileTables allows you to store files and documents in special tables in the SQL Server called FileTables, but access them from Windows applications as if they were stored in the file system, without making any changes to your client applications. For some of the use cases, you can use FileTables with CTE.

Considerations

- The CTE Agent must be installed on the same server where the FileTables reside. If the FileTables reside on your SQL server, then you should install the CTE Agent on your SQL server.
- If multiple servers access the SQL FileTables:
  - Install CTE agent on all of the servers.
  - Protect all of the FileTable folders with the same CTE policy.

Caution

Accessing the FileTable without CTE may corrupt the data.

- When you create a new FileTable, alter, or drop FileTables, this may require applying a new GuardPoint.
- Every FileTable has a separate FileTable Folder so you must apply separate GuardPoints for each FileTable.
- You must apply a unique GuardPoint to each VNN path.

  For example, if you configure two FileTables on an SQL Server, then the remote SQL administrator system must apply one GuardPoint to each configured VNN name.
- Guarding on a VNN name is similar to guarding a network path with CTE.
- If you want to access the FileTables from multiple remote systems, you must install CTE agent on those systems and apply the GuardPoints.

Caution

LDT is not supported with SQL FileTables. Only use offline Data Transformation to transform the SQL data.

Advantages

- System administrator cannot see the data locally on the SQL server because no CTE Agent is installed on the SQL server.
- The data transferring between servers is also encrypted.

Supported FileTables Use Cases

CTE supports the following FileTables use cases:

CTE Data Transformation of existing files in FileTables

Configuration guidelines:

1. Install CTE agent on the remote server.
2. Create a new FileTable, or Identify the FileTable folder for the existing FileTable.
3. Create an offline Data Transformation policy and apply to the GuardPoint on the FileTable folder.
4. Run the Dataxform utility to transform the data.

Protect files in SQL FileTables with CTE

Configuration guidelines:

1. Install CTE agent on the remote server.
2. Create a new FileTable, or Identify the FileTable folder for the existing FileTable.
3. Create a production policy and apply the GuardPoint on the FileTable folder.
4. Once the GuardPoint is active, you can use the file table to load and access files.

Protect files with SQL AlwaysOn Availability Groups with CTE

When the database that contains the FILESTREAM, or FileTable data, belongs to an AlwaysOn availability group, the FILESTREAM and FileTable functions accept or return virtual network names (VNNs) instead of computer names.

Configuration guidelines:

1. Install CTE agent on the remote server.

2. Create a new FileTable, or Identify the virtual network names (VNNs) for the existing FileTable.

3. Create a production policy and apply the GuardPoint to the VNN name

4. Once the GuardPoint is active, you can use the FileTable to load and access files.

5. When you enable FILESTREAM on an instance of SQL Server, it creates an instance-level share to provide access to the FILESTREAM data. Access this share by using the computer name in the following format:

   ```
   \\<computer_name>\<filestream_share_name>
   ```

6. In an AlwaysOn availability group, the computer name is virtualized by using a Virtual Network Name, (VNN). When the computer is the primary replica in an availability group, and databases in the availability group contain FILESTREAM data, then SQL creates a VNN-scoped share to provide access to the FILESTREAM data. Applications that use the file system APIs have to use the VNN-scoped share, which has a path in the following format:

   ```
   \\<VNN>\<filestream_share_name>
   ```

Install CTE on remote systems and guard the SQL Server VNN names

In this use case, CTE is installed on the SQL administrator system (a separate system from where the SQL Server resides) and a GuardPoint is applied to the VNN name.

Unsupported FileTables Use Cases

CTE does not support the following use cases:

1. Install CTE agent on the SQL Server and locally apply the GuardPoint on the SQL Server storage.

2. Access FileTables with Transact-SQL.

3. Access FileTables with File I/O APIs on the SQL server. Perform all file I/O on the remote system running the CTE agent.

Installing CTE on Microsoft SQL AlwaysOn

This section describes how to implement CTE with Microsoft SQL AlwaysOn in a variety of configurations for primary and secondary replica servers, and assumes that you have a basic understanding of Microsoft SQL database.

You may want to keep the primary server decrypted to serve all users, and use the secondary database for running reports or backups.

- If the database is encrypted, then the Volume Shadow copy-related backups will snapshot and backup encrypted protected data.

- Administrators with the `apply_key` permission can run a query and pull down reports from the secondary database server without affecting the performance of the primary database server.

- The secondary server could be in a remote Data Recovery location. You may want to secure it with encryption.

- LDT is supported with SQL AlwaysOn. See Using LDT with SQL AlwaysOn for more information.

Methods for Initial Encryption

There are multiple methods for performing the initial encryption of the databases. Decide on which of the following methods best fits your environment. For more information on transforming data, see the *CTE-Live Data Transformation with Data Security Manager*.

- Data Transformation – Encrypt data in place
- Backup and Restore to a GuardPoint
- Copy and paste the data into a GuardPoint

Configuration 1

- Databases on primary server and secondary replica servers require encryption
- Database name and location of secondary replica server are the same as the primary server

To perform the procedure:

1. Perform a full backup of the primary database.
2. Change the primary database to offline mode.
3. Confirm the creation of a data transformation and/or standard policy.
4. Guard the folder containing the primary database files with that policy:
    a. If using 'Encrypt data in place' as the selected method of encryption, execute the data transformation and then apply the standard policy.
    b. If using the 'Copy/Restore ' method of encryption, apply the standard policy on an empty folder/device.
5. On the secondary server, create a new folder to store the replicated database.

    > **Note**
    >
    > The folder name and the path must be the same as the primary server.

6. Guard the folder with the standard policy.
7. Perform step 4 above for additional secondary server(s).
8. Put the primary database back into online mode.
9. Setup SQL AlwaysOn High Availability group to perform FULL Data Synchronization.
   This copies the primary database and replicates it to secondary replica servers.

10. Verify that the databases in the secondary server are in “Synchronized” mode.

Configuration 2

- Database on the primary server does not require encryption, but the secondary replica database requires it
- Database names and locations for the secondary replica servers are the same as the primary server

To perform the procedure:

1. Perform a full backup of the primary database.
2. Confirm the creation of a data transformation and/or standard policy.
3. On the secondary server, create a new folder to store the replicated database.

   > **Note**
   >
   > The folder name and the path must be the same as the primary server.

4. Guard the folder with the standard policy.
5. Perform step 3 & 4 above for additional secondary server(s).
6. Setup SQL AlwaysOn High Availability group to perform **FULL Data Synchronization**.
   This copies the primary database and replicates it to secondary replica servers.
7. Verify that the databases in the secondary server are in “Synchronized” mode.

Configuration 3

- Databases on the primary and secondary replica servers require encryption
- Database name is the same, but the location of the secondary replica server is in a different location from that of the primary server

To perform the procedure:

1. Perform a full backup of the primary database.
2. Change the primary database to offline mode.

3. Confirm the creation of a data transformation and/or standard policy.

4. Guard the folder containing the primary database files with that policy:

   a. If using 'Encrypt data in place' as the selected method of encryption, execute the data transformation and then apply the standard policy.

   b. If using the 'Copy/Restore ' method of encryption, apply the standard policy on an empty folder/device.

5. On the secondary server, create a new folder to store the replicated database.

> **Note**
>
> The folder name and the path must be the same as the primary server.

6. Guard the folder with the encryption policy.

7. From secondary server, perform the restore to the primary database.

   a. Select the options **Restore with norecovery** and **Relocate all files to folder**.

   b. Specify the path of the new folder from step 5.

8. Repeat steps 4 & 5 above for any additional secondary server(s).

9. Setup SQL AlwaysOn High Availability group to perform **JOIN ONLY Data Synchronization**.
   This joins the secondary database to the SQL Always High Availability Group. It also establishes replication of new data and logs from the primary to the secondary replicated server.

10. Verify that the databases in the secondary server are in **Synchronized** mode.

Configuration 4

- Database on the primary server does not require encryption, but the secondary replica database requires encryption

- Database name is the same, but the location on the secondary replica server is in a different location than that of the primary server

To perform the procedure:

1. Perform a full backup of the primary database.

2. Confirm the creation of a data transformation and/or standard policy.

3. On the secondary server, create new folder to store the replicated database.

4. Guard the folder with the standard policy.

5. From secondary server, perform restore the primary database:

   a. Select the options **Restore with norecovery** and **Relocate all files to folder**.

   b. Specify the path of the new folder from step.

6. Setup SQL AlwaysOn High Availability group to perform **JOIN ONLY Data Synchronization**.

   Joins the secondary database to the SQL Always HA Group. It also establishes replication of new data and logs from the primary to the secondary replicated server.

7. Verify that the databases in the secondary server are in **Synchronized** mode.

Configuration 5

Following is an alternative method for protecting data in a MS SQL Server AlwaysON environment.

To perform the procedure:

1. Shut down SQL services completely, on the secondary node.

   > **Note**
   >
   > It is important to shut down the secondary node first, in order to keep the assignments the same.

2. Shut down SQL services completely on the primary node.

3. Create GuardPoints, using Data Transformation policies, on the directories containing the databases to be encrypted in the primary node.

> **Note**
>
> Perform encrypt-in-place encryption on each directory.

4. Create GuardPoints, using Data Transformation policies, on the directories containing the databases to be encrypted in the secondary node.

   > **Note**
   >
   > Perform encrypt-in-place encryption on each directory.

5. Delete the GuardPoints, using Data Transformation policies, from the primary node.

6. Create GuardPoints, using operational policies, on the four directories in the primary node.

7. Delete GuardPoints, using Data Transformation policies, from the secondary node.

8. Create GuardPoints, using operational policies, on the four directories in the secondary node.

9. Activate SQL services on the primary node.

10. Activate SQL services on the secondary node.

Data Transformation (Encryption in place)

For more information on transforming and encrypting data-in-place, see the CTE Agent: Data Transformation Guide.

Copy/Restore

For more information on transforming data using the copy and replace method, see the CTE Agent: Data Transformation Guide.

SQL Server Policy Tuning

In this section, you created and defined a process set for SQL Server that grants certain executables –in this case `sqlservr.exe-` unrestricted access to the database files. The need may arise to allow other executables, and/or users, access to the files.

You can grant this access by:

- Adding to the existing process set
- Creating a new one

The best option depends on the access requirements. The key decision is whether or not to select the **Apply Key** effect along with **Permit** or not. Omitting **Apply Key** on a security rule that still contains **Permit** allows the specified user or process to access to the data, but does not apply the encryption key, so therefore only shows them the data in its encrypted, cypher-text format. This is useful for anti-virus or backup software that may need to scan or copy the file, but does not necessarily need to see the contents.

Using LDT with Microsoft SQL

- Using LDT with SQL AlwaysOn
- Using LDT with SQL FILESTREAM

Using LDT with SQL AlwaysOn

To guard a directory with an LDT (Live Data Transformation) policy, you must temporarily close all of the files in that directory. In an SQL Server AlwaysOn environment, this may entail temporarily stopping the SQL Server service on the node that is being guarded. Once the directory is guarded, then you can start the SQL Server service immediately.

It is important to remember that the SQL Server AlwaysOn replication standard operating procedures.

- If one SQL Server service is taken offline for any reason, then once it is brought back on line, it takes the SQL Server a moment to re-synchronize the database nodes.
- The longer that secondary service was down, and the more inserts/updates and deletes that occurred on the still active node during that downtime, then the longer the synchronization period takes.

- During that synchronization period, any attempted fail over results in the SQL Server warning that data loss may occur if the fail over continues. However, once the SQL Server has completed re-synchronizing that secondary node, then any fail over is safe and does not result in loss of data.

Using LDT with SQL FILESTREAM

When applying a Live Data Transformation (LDT) GuardPoint to SQL Server with FILESTREAM enabled, a rekey may be triggered which never finishes. This can occur if SQL Server is renaming files when the GuardPoint is applied, which causes the rekey to start the scan process again. If the rekey seems to be taking a long time, stop the SQL service until the rekey finishes and then restart the SQL service.

CTE with Microsoft DFSR

The Microsoft Distributed File System Replication (DFS(R)) service is a multi-master replication engine used to keep folders synchronized on multiple servers. Using CTE with DFS(R) requires a unique configuration to make sure that all folders within a GuardPoint are only encrypted once, and that all sources to which those folders are replicated can access the proper encryption key to read the encrypted data.

This section contains the following topics:

- Overview
- Terms and Topology
- Configure DFS(R)
- CTE Configuration Workflow
- Creating Required DFS(R) Policy Components
- Using the Standard/Offline Encryption Method
- Using the LDT Encryption Method
- Troubleshooting

For more information about DFS(R), see the Microsoft DFS(R) documentation at Distributed File System Replication.

Overview

This section details the CipherTrust software suite and will act as an operational (run book) guide for managing the CipherTrust security environment. It contains the steps to configure CTE agent on a Windows Server host running Microsoft Distributed File System (DFS) with Replication.

Microsoft introduced the Distributed File System in Server 2003 as a value-add for customers seeking low-cost data resiliency across high-latency, low-bandwidth WAN links. DFS is a free feature included with all subsequent versions of Windows Server software. DFS uses a namespace architecture where nested data folders use a replication service to synchronize similar data directories on 2 or more servers. Architects choose this model to ensure fault tolerance, preserve active uptime and to improve access performance by spreading the workload across many servers: client requests are routed to the nearest DFS folder for a given namespace. CTE began supporting DFS(R) starting with Windows 2012 R2 and continues to do so through Windows 2022 and subsequent versions.

Use this document for guidance on encrypting DFS data depending on how it is currently deployed. It contains a number of scenarios, often highlighting important configuration considerations applicable for the project. Before encrypting any DFS data, however, Thales Support assumes that the customer has made a backup of the latest production data, has a good understanding of their own DFS(R) deployment and has a fully-developed encryption plan.

The information listed here details best practices and tool sets when using encryption for DFS data. It starts by defining the terms and layouts for this technology and ends by reviewing common problems that may compound DFS(R) encryption and troubleshooting tips.

CTE Encryption Methods

CTE supports two encryption methods:

- **Standard offline data transformation**

  Where the data is **unavailable** while it is being encrypted or rekeyed.

- **Live Data Transformation**

  Where the data is encrypted and rekeyed in the background while it remains **accessible to users**. This method requires a separate license for the LDT feature.

While DFS(R) policies have some unique required components, the basic policy and GuardPoint creation process is identical to non-DFS(R) environments. For details about offline data transformation, see the CTE Data Transformation Guide. For details about LDT, see the CTE Live Data Transformation with CipherTrust Manager.

Considerations with DFS(R)

If you are using CTE in a DFS(R) environment, keep in mind the following:

- You should always back up your data prior to beginning the encryption process and you should have a full backup of the data in the hub server before you restore a spoke.
- You cannot place a GuardPoint *anywhere* on the boot drive, so if your DFS(R) replication point is currently `C:\`, or a directory under `C:\` such as `C:\data\`, you need to move that data and its replication point to a new volume on the server before you can encrypt it.
- If you are backing up your DFS data, make sure that your backup software is not backing up the archive bit. File replication is triggered by file version change or a modified time stamp. As such, there is a chance that updating the archive bit may cause issues that trigger a replication storm, which will then put a heavy encryption load on the servers.
- You must add the CTE GuardPoint at or above the level of the DFS(R) replication point. For example:
    - If the replication point is `D:\`, the CTE GuardPoint must also be at `D:\`. Adding a GuardPoint on a directory in `D:\`, such as `D:\data\`, will fail.
    - If the replication point is `D:\data\`, you can add a GuardPoint at `D:\data\` or `D:\`, but you *cannot* add a GuardPoint on a subdirectory of `D:\data\` such as `D:\data\HR-files\`.
- When you set a replication point, Microsoft automatically creates a private directory called `<dir name>\DfsrPrivate` that resides with that replication point. For example, if the replication point is set on `D:\`, the private directory would be `D:\DfsrPrivate`. If the replication point is set on `D:\data\`, the private directory would be `D:\data\DfsrPrivate`.

    How this private directory must be handled depends on the the encryption method that you are using.

    - For **Standard** encryption, you must guard the private directory with the same policy that you use for the main GuardPoint. If the GuardPoint is at the root of the volume (for example, `D:\`), this happens automatically. But if you are guarding a specific directory, such as `D:\data\`, you need to create a second GuardPoint using the same policy on `D:\data\DfsrPrivate`.

    - For **LDT**, you must guard the private directory with the same policy that you use for the main GuardPoint, even if the GuardPoint is at the root level. (For example, you must have a GuardPoint for both `D:\` and `D:\DfsrPrivate\`.) In addition, you must exclude this directory from LDT processing.

- The policy you specify for a DFS(R) GuardPoint **cannot** contain a resource set in any of the key rules included in the policy. All files in the guarded directory, and its subdirectories, must be encrypted with the same encryption key **without exception**. Additionally, if you rekey the GuardPoint, all files must be rekeyed with the same encryption key.

- If you want to change from one encryption key to an entirely different encryption key (as opposed to rekeying the data with a new version of the existing key), you must decrypt the data and remove all existing GuardPoints so that you have a clean environment. Then you can start the CTE encryption process over from the beginning.

> **Caution**
>
> **You cannot change from one encryption key to another if any of the existing data is still encrypted with the old key. If you attempt to do so, you may encounter data replication errors and you may need to delete the entire volume and recreate it.**

- When CTE encrypts data on a node, the encrypted data must be replicated to other nodes in the configuration. This may result in increased replication activity on the network.

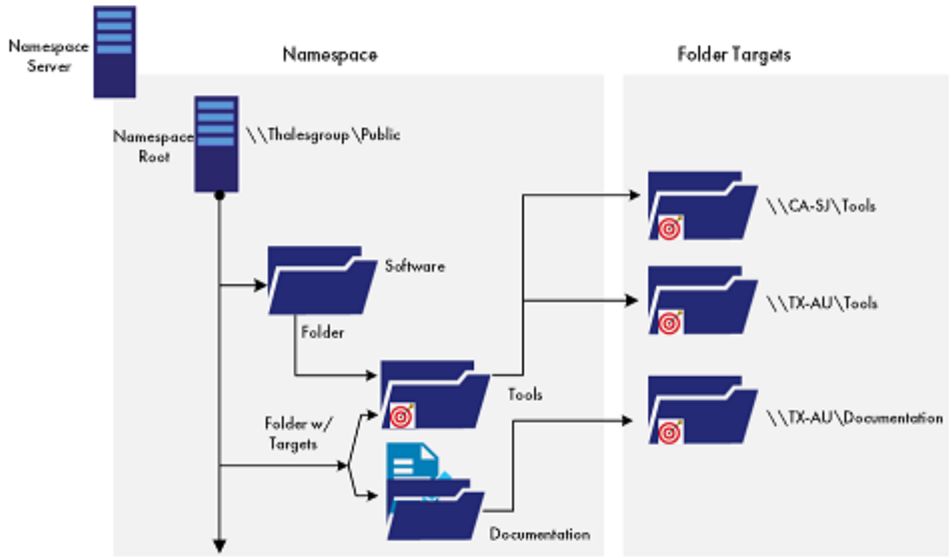
Terminology and Topology

DFS(R) uses a compression algorithm known as remote differential compression (RDC). RDC detects changes to the data in a file and enables DFS Replication to replicate only the changed file blocks instead of the entire file. DFS(R) uses a database engine that keeps track of all file deltas and is used when reconciling replicated data across servers.

Term	Definition
DFS Namespace	A central namespace through which you can see a unified view of the shared folders that are included in the DFS.
DFS Namespace Server	The server that hosts the DFS Namespace. DFS Namespace Root is the top level of the DFS namespace.

Term	Definition
	The namespace root and the DFS namespace use the same name.
DFS Folder	A folder presented to a client within the DFS namespace, but below the DFS root. A DFS folder can exist on the same server that is hosting the DFS root, but it is not required. DFS folders commonly represent file system resources located on other servers.
DFS Tree	A DFS tree is a reference to the DFS hierarchy. The tree starts with the DFS root, and contains all of the DFS folders defined within the root.
Replicated folder	A folder that stays synchronized on each member. As the data changes in each of the replicated folders, the changes replicate across connections between all members of the replication group.

In order to use DFS(R), administrators designate a namespace on a namespace server. Folders nested within the namespace contain all of the data that is distributed across many servers.



The **Tools** folder, in the preceding graphic, is part of a replication group composed of two servers, known as members, which participate in the replication of one or more replicated folders.

Creating multiple replicated folders in a single replication group simplifies the process of deploying replicated folders because the topology, schedule, and bandwidth throttling for the replication group are applied to each replicated folder. You can administer DFS(R) by using DFS Management, the DFS(R) Admin and DFS(R) diag commands, or scripts that call WMI. The connections between all members form the replication topology.

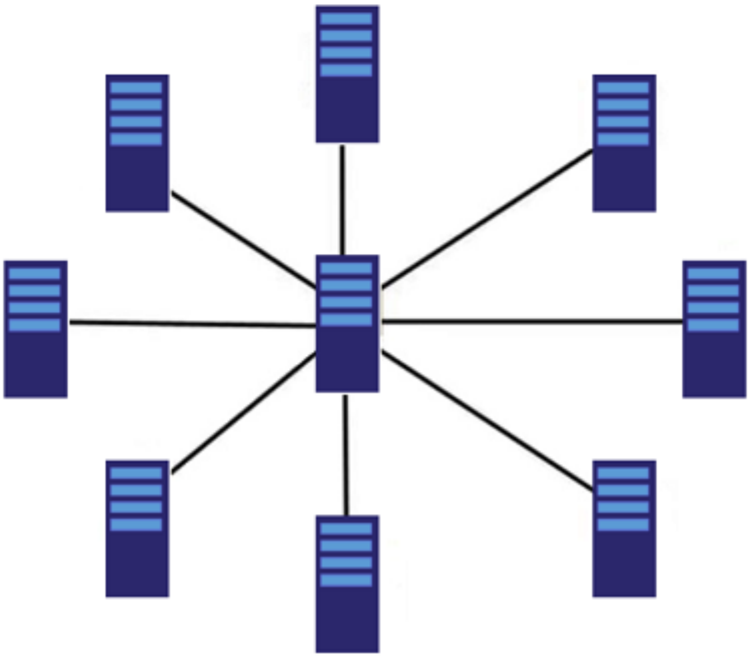
DFS(R) Topology

How you deploy CTE in a DFS(R) environment depends on the topology you have chosen for your DFS(R) configuration. Microsoft offers several topology options for DFS(R):

Hub and Spoke

In this configuration, there is a central server (an explicitly-designated hub) whose content is replicated on multiple satellite servers (the spokes). While each spoke server has a two-way communication channel with the hub server, none of the spoke servers can communicate with each other. If the data changes on one spoke server, that server communicates the changes back to the hub server and the hub server initiates the data replication on all of the other spoke servers.

This configuration allows you to encrypt servers one at a time, starting with the hub and then moving outwards to the spokes. This type of topology tends to be very efficient, but the problem with it is that if the initial master fails, then all replication ceases to function until it is online again. This topology is typically used for WANs that consist of faster network connections between major computing hubs and slower links connecting branch offices.

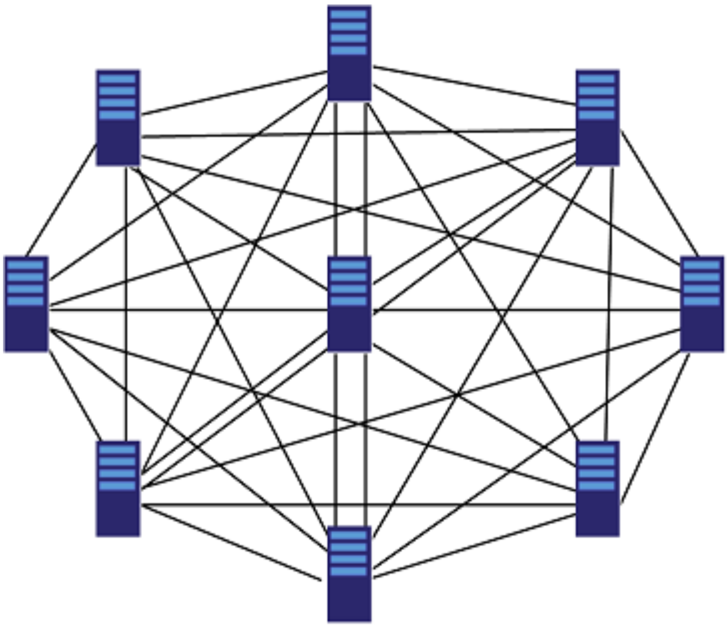


Full Mesh

A full mesh topology allows every replica to replicate with every other replica. In this configuration, data replication can be initiated by any server on all of the other servers. The advantage of using this topology is that replication continues functioning even if a server drops off-line. Users see no interruption in service. Deploying this model is useful for maintenance operations and satisfies data resiliency. The disadvantage to using this method is that it can result in an excessive amount of replication traffic.

As of Windows 2012, Microsoft added support for more than 5 replicas in a set. In addition, DFS(R) now supports multiple Namespaces: good for encrypting different namespaces with different AES keys (when required for “fencing” requirements).

In this configuration, you must stop the replication service while you encrypt the data on all servers in the mesh. You cannot restart the replication service until the initial encryption has completed on all servers.



No Topology

This option allows for creating a replication group without defining any replication topology. This means administrators may create their own custom replication topology later on.

Configure DFS(R)

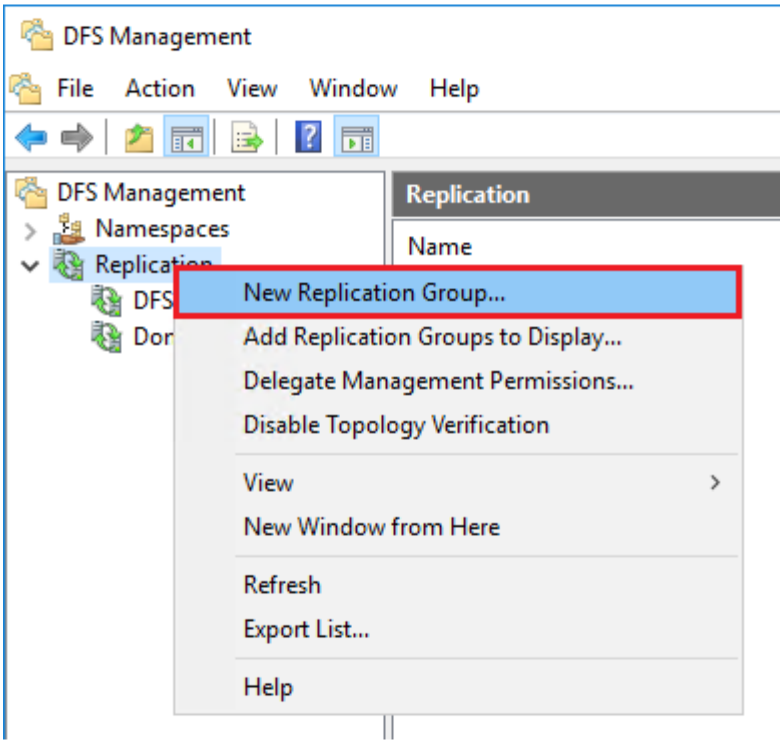
Note

The following instructions were written for Windows 2016. If you are using a subsequent version, steps may differ slightly. See Distributed File System Replication for more information.

1. Launch the Server Manager applet.

2. Select **Tools > DFS Management**.

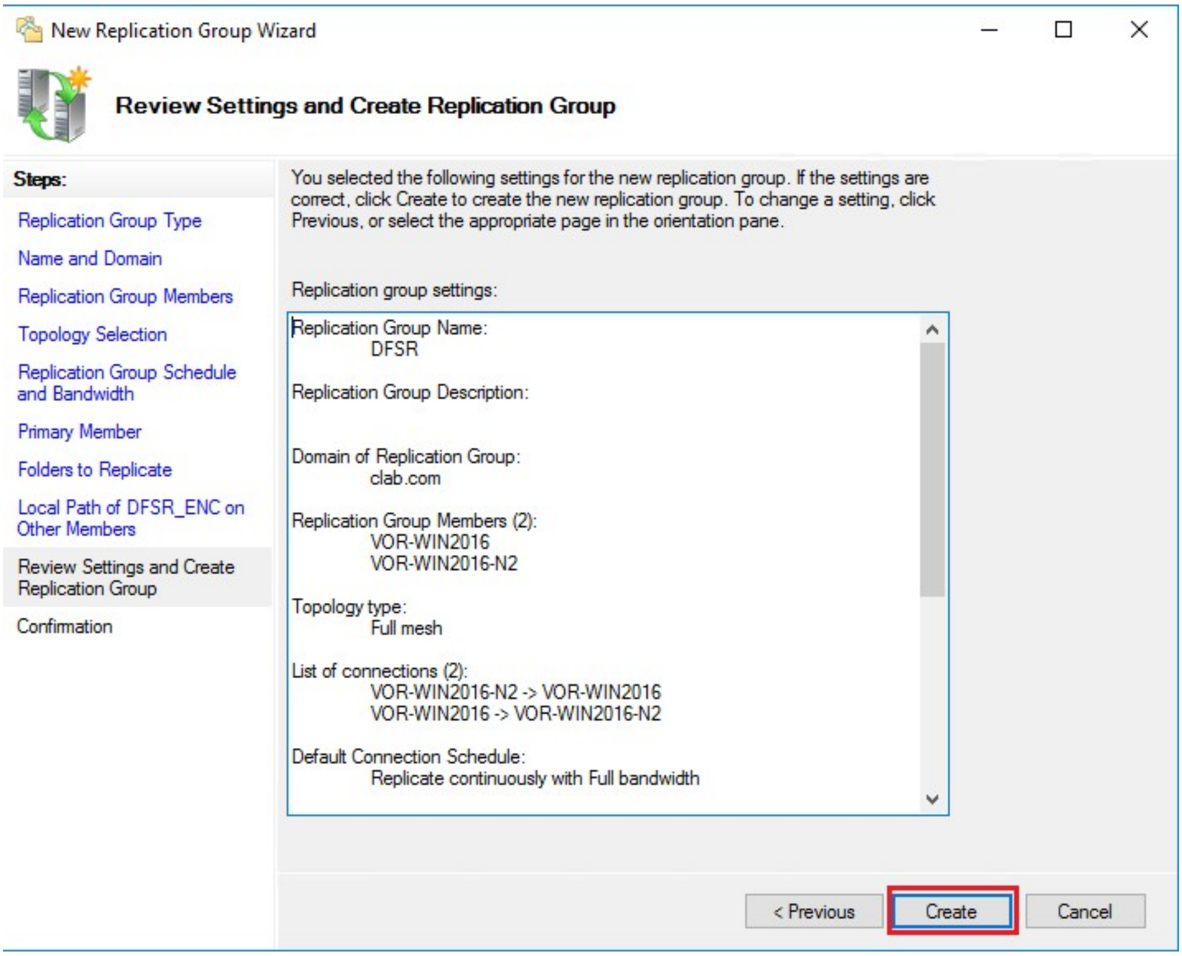
3. Right-click on Replication and select **New Replication Group …**



4. For replication group, select **Multipurpose Replication Group**. Then click **Next**.

5. Enter a Replication Group name and the Domain name. Click **Browse** to select the domain. Then click **Next**.

6. In the Replication Group Members window, click **Add** and enter all of the DFS(R) replication group member servers. Then click **Next**.


New Replication Group Wizard
Review Settings and Create Replication Group
Steps:
Replication Group Type
Name and Domain
Replication Group Members
Topology Selection
Replication Group Schedule and Bandwidth
Primary Member
Folders to Replicate
Local Path of DFSR_ENC on Other Members
Review Settings and Create Replication Group
Confirmation
< Previous
Create
Cancel

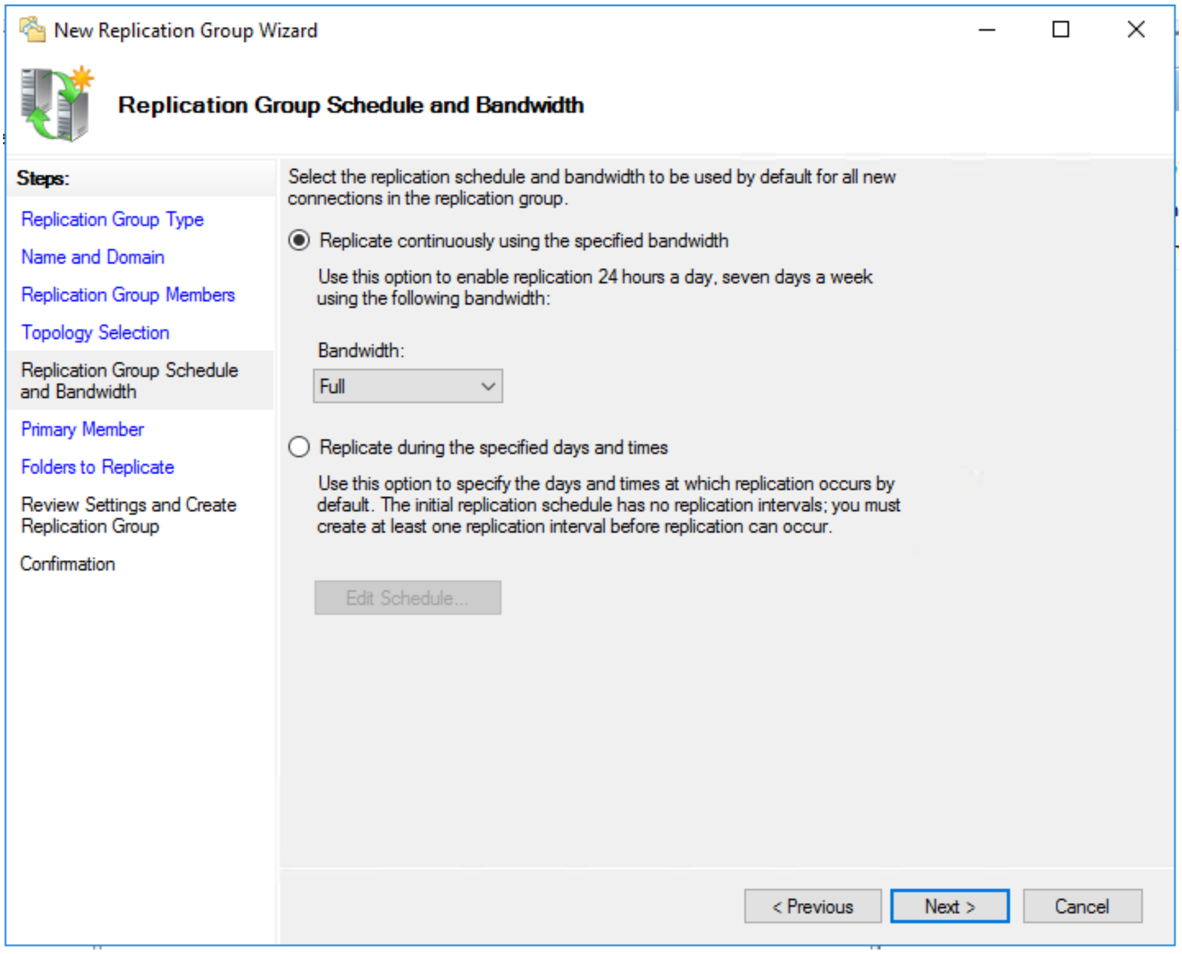
You selected the following settings for the new replication group. If the settings are correct, click Create to create the new replication group. To change a setting, click Previous, or select the appropriate page in the orientation pane.

Replication group settings:

```
Replication Group Name:
          DFSR

Replication Group Description:


Domain of Replication Group:
          clab.com

Replication Group Members (2):
          VOR-WIN2016
          VOR-WIN2016-N2

Topology type:
          Full mesh

List of connections (2):
          VOR-WIN2016-N2 -> VOR-WIN2016
          VOR-WIN2016 -> VOR-WIN2016-N2

Default Connection Schedule:
          Replicate continuously with Full bandwidth
```

7. Select the replication topology. Then click **Next**.

8. Set the replication schedule. Then click **Next**.



Replication Group Schedule and Bandwidth

Steps:

- Replication Group Type
- Name and Domain
- Replication Group Members
- Topology Selection
- Replication Group Schedule and Bandwidth
- Primary Member
- Folders to Replicate
- Review Settings and Create Replication Group
- Confirmation

Select the replication schedule and bandwidth to be used by default for all new connections in the replication group.

● Replicate continuously using the specified bandwidth

Use this option to enable replication 24 hours a day, seven days a week using the following bandwidth:

Bandwidth: Full

○ Replicate during the specified days and times

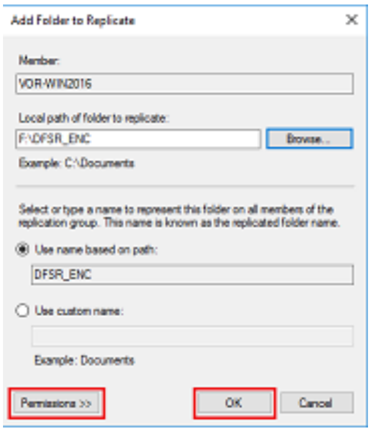
Use this option to specify the days and times at which replication occurs by default. The initial replication schedule has no replication intervals; you must create at least one replication interval before replication can occur.

9. In the Primary Member section, select the Primary DFS(R) node. Then click **Next**.

10. In Folders to Replicate, click **Add**.

11. Click **Browse** and navigate to the folder on the primary node to place under DFS(R) replication.

12. Once the path is set, click **Permissions**.



Member: VOR-WIN2016

Local path of folder to replicate: F:\DFSR_ENC

Example: C:\Documents

Select or type a name to represent this folder on all members of the replication group. This name is known as the replicated folder name.

● Use name based on path: DFSR_ENC

○ Use custom name:

Example: Documents

13. Change the permissions or keep the existing folder permissions. Click **OK**.

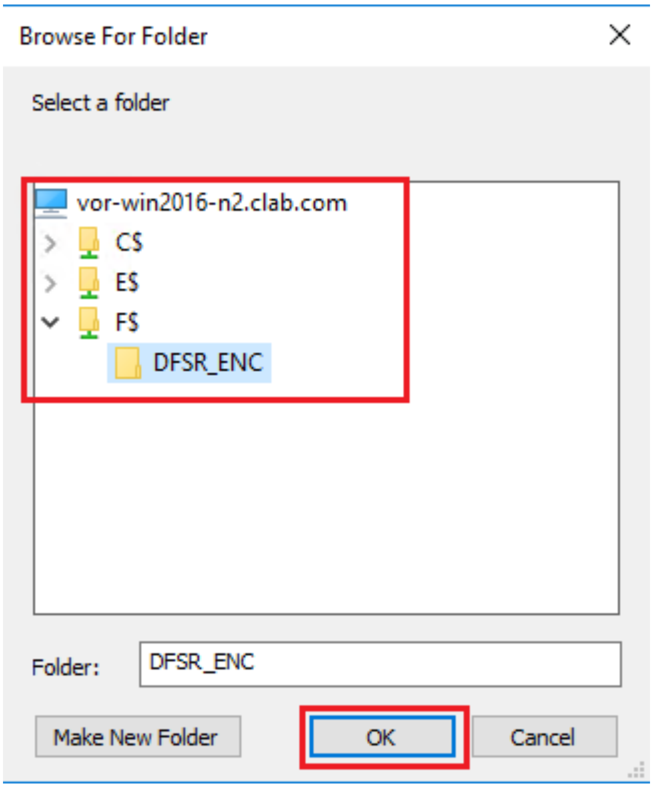
14. On the local path for other members, click **Edit**.

15. Click **Browse** and select the target replication folder on the replication group member.

Note

If the path is different from the Primary member, then you cannot use CipherTrust Client Groups. You must set GuardPoints separately on individual member nodes.

16. Browse the folder structure for the member node and select the target replication folder. Click **OK**.



Browse For Folder

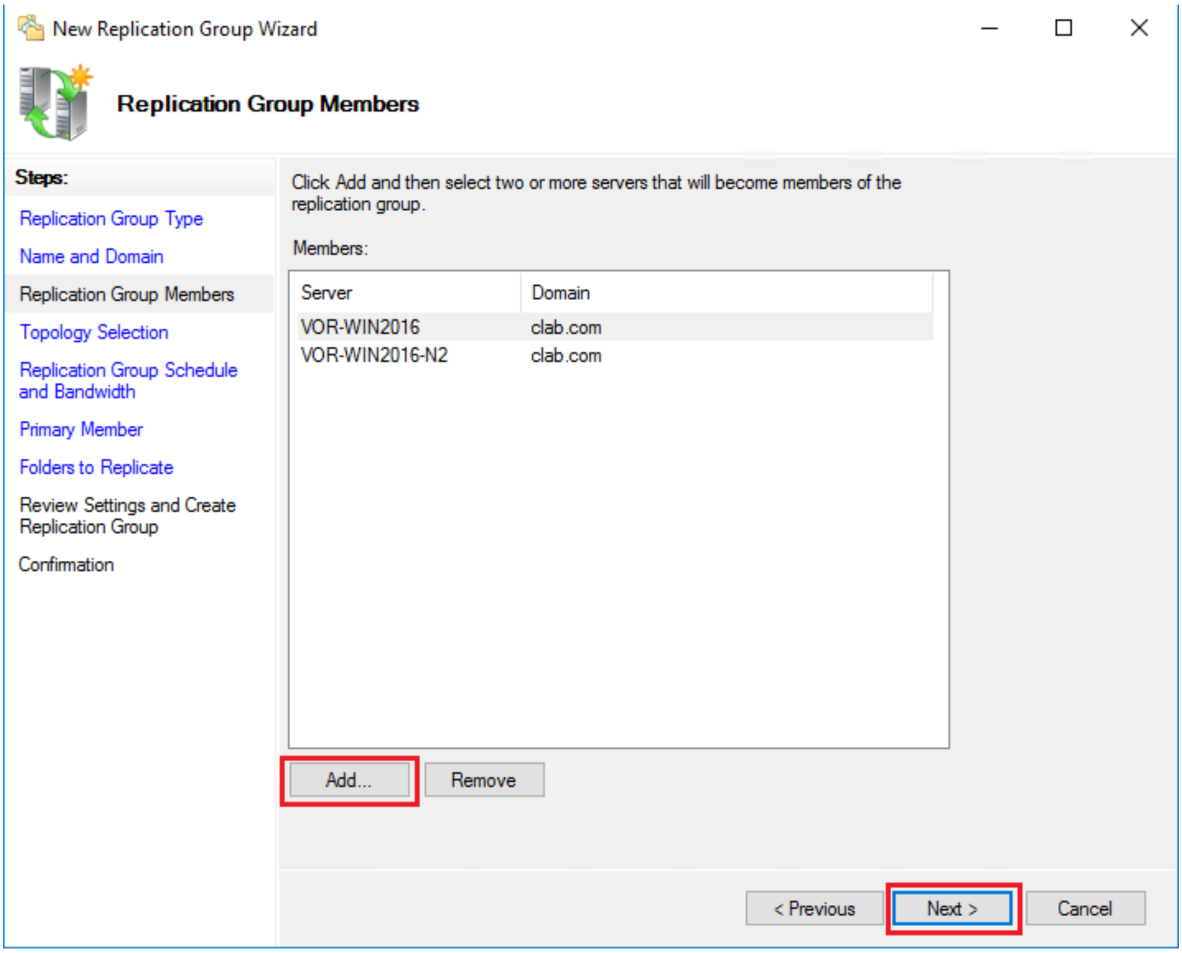
Select a folder

vor-win2016-n2.clab.com
- C$
- E$
- F$
  - DFSR_ENC

Folder: DFSR_ENC

17. If the target replication folder displays correctly, click **OK** to continue. If it does not, click **Browse** and reset the folder.

18. Review the configuration and click **Create** if it all looks correct.


New Replication Group Wizard
Replication Group Members
Steps:
Replication Group Type
Name and Domain
Replication Group Members
Topology Selection
Replication Group Schedule and Bandwidth
Primary Member
Folders to Replicate
Review Settings and Create Replication Group
Confirmation
Add...
Remove
< Previous
Next >
Cancel

Click Add and then select two or more servers that will become members of the replication group.

Members:

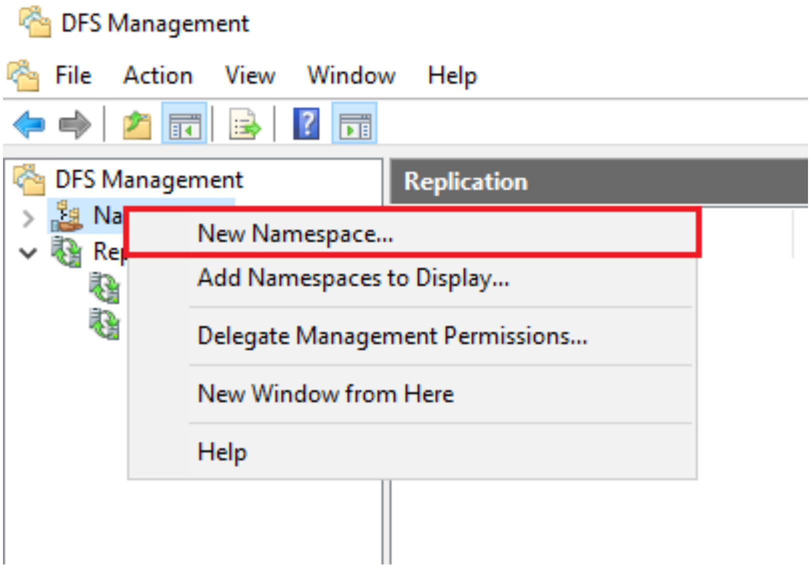
Server	Domain
VOR-WIN2016	clab.com
VOR-WIN2016-N2	clab.com

19. Click **Close** for a successful DFS(R) deployment.

Configure Namespace

DFS(R) namespace allows you to access the data remotely, through a network share path. For CipherTrust Transparent Encryption, it must have a policy that has a security rule that contains a user set. The user set must include privileged users linked to the process `ntoskrnl.exe`. This provides additional security for sensitive data protected on a DFS(R) folder.

1. In the DFS management tool, right-click on Namespace and select **New Namespace…**

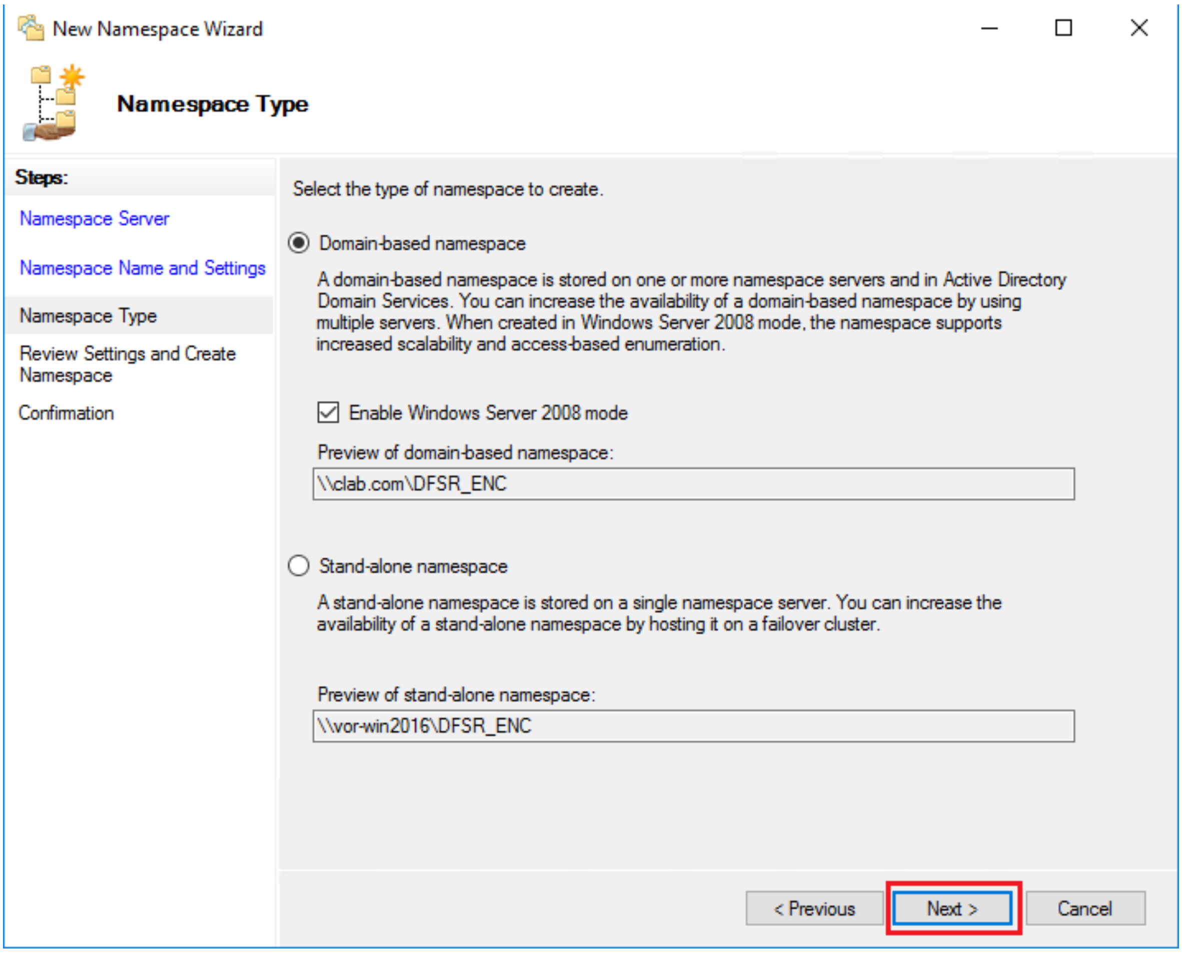


2. Click **Browse** and search for the replication group member to host the namespace.

3. Select the node that will host the DFS(R) Namespace. Click **OK** and **Next**.

4. Enter a name for the Namespace and click Next.

5. Accept the default for the Domain-based namespace and click **Next**.



Select the type of namespace to create.

● Domain-based namespace

A domain-based namespace is stored on one or more namespace servers and in Active Directory Domain Services. You can increase the availability of a domain-based namespace by using multiple servers. When created in Windows Server 2008 mode, the namespace supports increased scalability and access-based enumeration.

☑ Enable Windows Server 2008 mode

Preview of domain-based namespace:

\\clab.com\DFSR_ENC

○ Stand-alone namespace

A stand-alone namespace is stored on a single namespace server. You can increase the availability of a stand-alone namespace by hosting it on a failover cluster.

Preview of stand-alone namespace:

\\vor-win2016\DFSR_ENC

6. Review the settings and click **Create** to create the Namespace.

7. After a successful outcome, click **Close**.

Install the DFS(R) Role and Features

Consult the Microsoft documentation to install DFS by using Server Manager

Stopping and Starting the DFS Services

When setting up the topology, you will have to start and stop the DFS services. Following are the steps to stop and start the DFS services on a DFS management node. On all the nodes in a DFS(R) topology, there are two services running:

- DFS Namespace Service
- DFS Replication Service

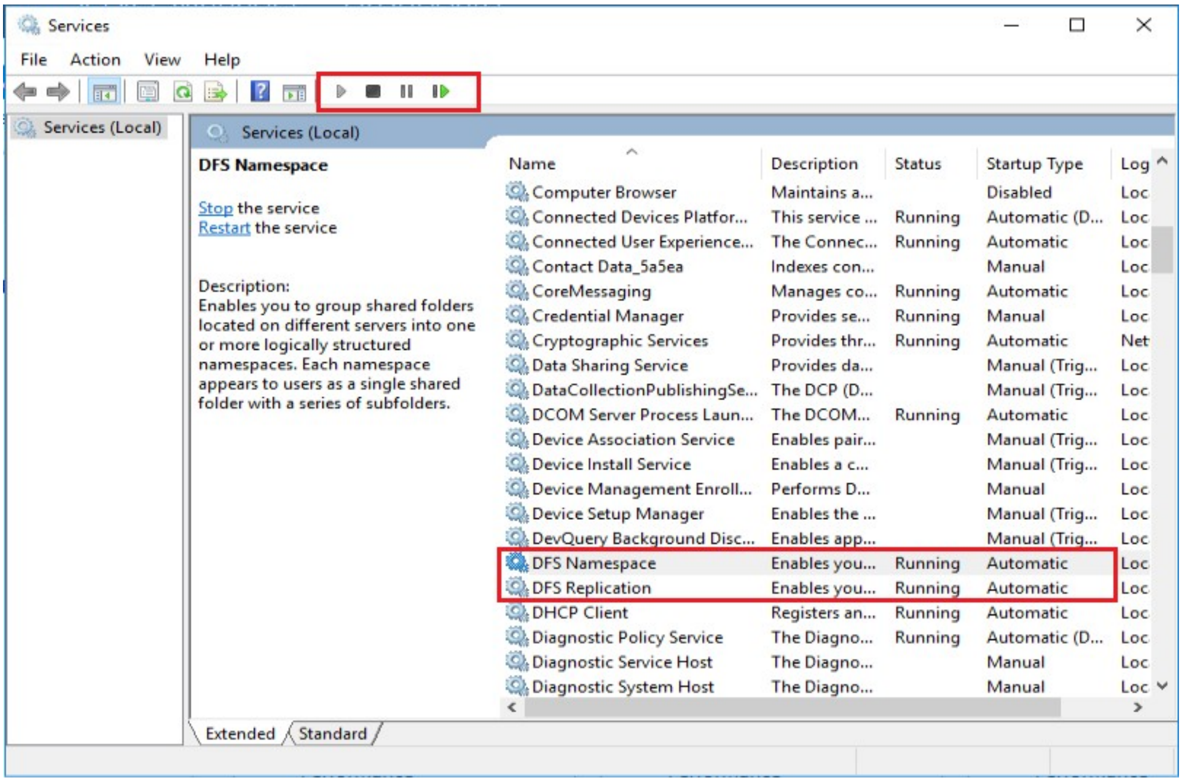
Both of these services must be stopped before you can apply a CTE GuardPoint successfully:

1. Right-click on the start menu and click **Run**.

2. Type `services.msc` and hit enter. This launches the Services Management Console.

3. Locate the DFS Namespace and DFS Replication Services.

4. Select each one individually and click Stop.



DFS Namespace

Stop the service
Restart the service

Description:
Enables you to group shared folders located on different servers into one or more logically structured namespaces. Each namespace appears to users as a single shared folder with a series of subfolders.

Name	Description	Status	Startup Type	Log
Computer Browser	Maintains a...		Disabled	Loc
Connected Devices Platfor...	This service ...	Running	Automatic (D...	Loc
Connected User Experience...	The Connec...	Running	Automatic	Loc
Contact Data_5a5ea	Indexes con...		Manual	Loc
CoreMessaging	Manages co...	Running	Automatic	Loc
Credential Manager	Provides se...	Running	Manual	Loc
Cryptographic Services	Provides thr...	Running	Automatic	Net
Data Sharing Service	Provides da...		Manual (Trig...	Loc
DataCollectionPublishingSe...	The DCP (D...		Manual (Trig...	Loc
DCOM Server Process Laun...	The DCOM...	Running	Automatic	Loc
Device Association Service	Enables pair...		Manual (Trig...	Loc
Device Install Service	Enables a c...		Manual (Trig...	Loc
Device Management Enroll...	Performs D...		Manual	Loc
Device Setup Manager	Enables the ...		Manual (Trig...	Loc
DevQuery Background Disc...	Enables app...		Manual (Trig...	Loc
DFS Namespace	Enables you...	Running	Automatic	Loc
DFS Replication	Enables you...	Running	Automatic	Loc
DHCP Client	Registers an...	Running	Automatic	Loc
Diagnostic Policy Service	The Diagno...	Running	Automatic (D...	Loc
Diagnostic Service Host	The Diagno...		Manual	Loc
Diagnostic System Host	The Diagno...		Manual	Loc

CTE Configuration Workflow

In order to configure CTE with DFS(R), you must complete the following tasks:

1. Make sure you have a valid backup of the data you intend to encrypt.

2. Identify the volumes or folders you intend to encrypt.

3. Select an encryption method and make sure you understand how to create and deploy CTE GuardPoints using that encryption method. For details, see one of the following documents:

    - CTE Data Transformation Guide

    - CTE-Live Data Transformation with CipherTrust Manager

4. Create a DFS(R) Process Set and User Set for the policy. For details, see Creating Required DFS(R) Policy Components.

5. Create the policies and GuardPoints you need to protect your data, using the process appropriate to the selected encryption method. For details, see one of the following:

    - Using the Standard Encryption Method

    - Using the CTE-LDT Encryption Method

6. Consider configuring all nodes within a DFS topology under a single CipherTrust Client Group

Creating Required DFS(R) Policy Components

DFS(R) uses two services to for the replication process, `dfsrs.exe` and `ntoskrnl.exe` that must be associated with the NT AUTHORITY user. In order to do this, you need to create a process set and a user set that must then be combined into a security rule in the policy.

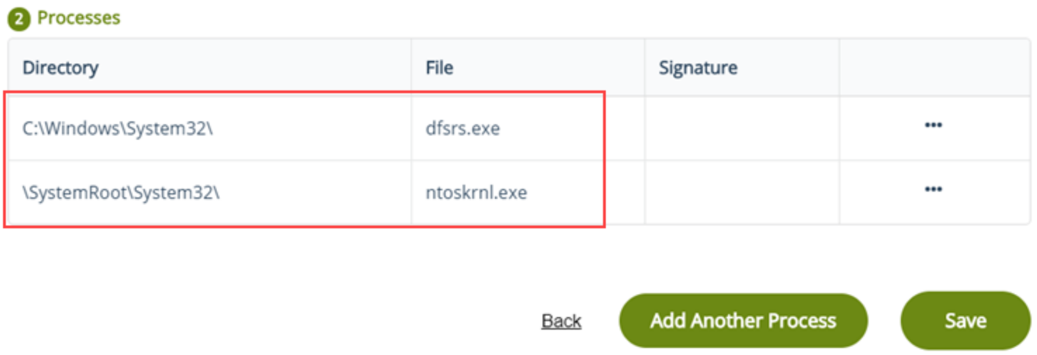
> **Note**
>
> Once you create these components, you can use them in any number of policies for both standard and LDT GuardPoints.

1. Log into the CipherTrust Manager Console and switch to the correct domain.
2. Launch the **Transparent Encryption** application.
3. In the left-hand menu bar, expand **Policies** and select **Policy Elements**.
4. Create a process set for the required DFS(R) processes:
   a. Click **Process Sets**.
   b. Click **Create Process Set**.
   c. In the **Name** field, enter a name for this process set. In this example, we will use DFS(R)-Processes.
   d. Click **Next**.
   e. Enter the first DFS(R) process:
      - In the **Directory** field, enter `C:\Windows\System32\`.
      - In the **File** field, enter `dfsrs.exe`.

   f. Click **Next**.
   g. Below the table, click **Add Another Process**:
      - In the **Directory** field, enter `\SystemRoot\System32\`.

- In the **File** field, enter `ntoskrnl.exe`.

h. Click **Next**. The process set should look like this:



2 Processes

Directory	File	Signature	
C:\Windows\System32\	dfsrs.exe		•••
\SystemRoot\System32\	ntoskrnl.exe		•••

i. Click **Save** to save the process set.

5. Create the user set for the required `NT AUTHORITY` user:

a. Click **User Sets** tab.

b. Click **Create User Set**.

c. In the **Name** field, enter a name for the user set. In this example we will use `Local_NT_AUTHORITY`.

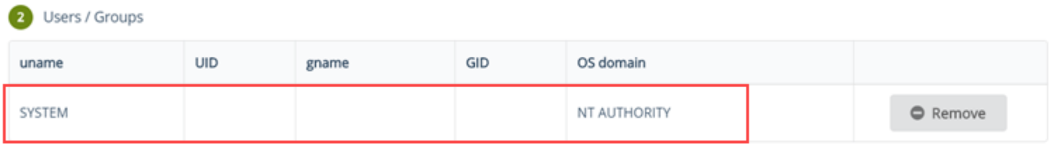
d. Click **Next**.

e. Click the **Manually Add Users** tab.

f. In the **uname** field, enter `SYSTEM`.

g. In the **OS domain** field, enter `NT AUTHORITY`.

h. Click **Next**. The user set should look like this:



2 Users / Groups

uname	UID	gname	GID	OS domain	
SYSTEM				NT AUTHORITY	⊖ Remove

i. Click **Save** to save the user set.

j. Optionally, create another user set for other authorized users in the namespace. For example, you may want to add the "Administrator" user in each of the domains that are part of the namespace. You can create as many separate user sets as required.

6. Create the mandatory security rule for DFS(R):

The first security rule in a standard policy is for the DFS(R) process set (created in the previous section). This rule allows the DFS(R) processes to access the data and manage the replication under `DfsrPrivate` folder. The minimum permissions must be:

Rule	Permissions	Description
Process Set	DFS(R)_service	Contains the process `dfsrs.exe` and `ntoskrnl.exe`
Action	all_ops	Allows all read/write operations
Effect	Permit	Allows data to be copied/moved/replaced as encrypted

7. When you have finished created the required components, you can use the components to create the appropriate policy for your chosen encryption method.

Using the Standard Encryption Method

If you want to encrypt your data using the standard (offline) encryption method, you need to create two different policies. The first policy is the initial encryption policy that specifies the symmetric key that you want to use to encrypt the data for the first time. The second is the production policy that you want to use for day-to-day operations on the encrypted data.

The initial encryption must be done while the volume or directory is offline, and users and applications must be prevented from accessing the data until the entire encryption process has finished. Once this initial encryption has been completed, any new or changed data in the GuardPoint will be automatically encrypted as it is added.

> **Note**
>
> If you want to encrypt the data without restricting access during the encryption process, you can use the LDT feature. For details, see Using the CTE-LDT Encryption Method.

To use the standard encryption method:

1. Make sure that you have created the required policy components for DFS(R) as described in Creating Required DFS(R) Policy Components.

2. Create the initial encryption and production policies as described in Creating Standard Policies for DFS(R).

3. Create the GuardPoints that you want to use. The GuardPoint creation method depends on your DFS(R) topology. For details, see one of the following:

   - Creating Standard GuardPoints with the DFS(R) Hub and Spoke Topology

   - Creating Standard GuardPoints with the DFS(R) Full Mesh Topology

Using the LDT Encryption Method

If you want to encrypt your data using LDT, you need to create a Live Data Transformation policy and use that to create your GuardPoints. All encryption occurs in the background while users continue to access the data.

With LDT, the data will be automatically rekeyed periodically, based on the expiration date and the life span of the versioned key used to encrypt the data.

To use the LDT:

1. Make sure you have created the required policy components for DFS(R) as described in Creating Required DFS(R) Policy Components.

2. Make sure that you have a versioned encryption key.

3. Create the LDT policy as described in Creating a LDT Policy for DFS(R).

4. Create the GuardPoints you want to use as described in Creating a LDT GuardPoint for DFS(R).

5. Consider using QoS for optimal DFS(R) performance.

Troubleshooting

Common DFS(R) Configuration Mistakes

DFS(R) uses a staging area quota when processing replication tasks. If the allocated space is too small, performance is negatively impacted. An improperly sized staging area may also cause a replication loop among downstream nodes. Due to the intensive nature of encryption processing DFS data, you should evaluate your environment readiness. For more information about proper sizing techniques, see How to Determine the Minimum Staging Area DFSR Needs for a Replicated Folder.

Review the following configuration mistakes as part of site readiness preparations:

Improper or Untested Seeding

To save downtime, administrators may choose to pre-seed new member replicas with DFS data before configuring all of the members in a replication group. In this manner, the initial synch consists of mostly delta changes rather than entire object transactions. See Replacing DFSR Member Hardware or OS (Part 2: Pre-seeding) for information on the advantages of pre-seeding.

Common issues include:

- ACL mismatch between source and target
- Changes were made to the files after they were copied to the new member
- No UAT testing was done to verify the pre-seeding process worked as expected

High DFS(R) Backlog

Customer DFS(R) deployments should be relatively up-to-date in replicating files across multiple nodes. High backlogs, especially over an extended period of time, mean that considerable amounts of data is out-of-sync. Unwanted conflict resolution may occur during these periods. Introducing encryption in this scenario would severely degrade performance and, likely require troubleshooting.

Hub Node – Single Point of Failure

The DFS primary active node is a single-point-of-failure in a Hub-and-Spoke topology. Fortunately, if the hub server goes down, all spoke servers retain the last-known-good data. All local changes are recorded locally and will not replicate until the primary node is back online. Spoke servers retain their own copies of their deltas and do not share them with others in the topology. You can confirm this risk, and confirm that offline backups are taken nightly to ensure optimal RTO (Recovery Time Objective).

Jet Database

DFS(R) maintains one Jet database per volume. As a result, placing all of your replicated folders on the same volume puts them all in the same Jet database. If that Jet database has a problem that requires repair or recovery of the database, all of the replicated folders on that drive are affected. It is better to spread replicated folders around, using as many drives as possible to provide maximum uptime for the data.

Windows Server patch-level

Prepare for the deployment by checking for the latest software update for their DFS(R) servers. For replication, always make sure DFS(R) and NTFS are at least at the latest version listed. Proactively patching the DFS(R) servers is advisable, even if everything is running normally, as it will prevent your servers from being affected by a known issue.

DFS(R) as Backup

DFS(R) is not a bona fide backup solution. To be fully protected, customers must backup their data offline. DFS(R) was not designed as a backup solution. One of DFS(R)'s design goals is to be part of an enterprise backup strategy in that it gets your geographically distributed data to a centralized site for backup, restoration and archiving. Multiple members do offer protection from server failure; however, this does not protect your data from accidental deletions. Encrypting customer data should never occur without a full backup.

Stopping DFS(R) Replication

Sometimes, you may need to temporarily stop replication. Changing the replication status has consequences. The proper method is to set the schedule to no replication for the Replication Group in question. The DFS(R) service must be running to be able to read updates in the journal. Additionally, do not stop the DFS(R) service for long periods of time (days, weeks). Doing so may cause a journal wrap to occur (if many files are modified, added, or deleted in the meantime). DFS(R) will recover from the journal wrap, but in large deployments, this takes a long time and replication does not occur, or happens very slowly, during the journal wrap recovery. Monitor and prepare the environment prior to encryption.

File System Policies

Do not configure file system policies on replicated folders. The file system policy reapplies NTFS permissions at every Group Policy refresh interval. This can result in sharing violations because an open file does not replicate until the file is closed.

Backup Software

Having DFS data on other servers helps protect the data against a catastrophic failure, but does nothing to protect against data corruption. If a file becomes corrupted, the corruption gets replicated to other targets. Because the data should be identical on each DFS replica, backing up only one of the replicas is usually sufficient. Thales recommends that you backup at least the primary active node or hub.

Another important consideration regarding the backup process is that it is very critical to configure the backup software to not update the archival bit. The reason for this is that file replication is triggered by file version change or a modified time stamp. Therefore, there is a chance that updating the archive bit may cause issues that trigger a replication storm.

Troubleshooting DFS(R)

Unexpected problems may arise while encrypting replicated DFS data. Following are some of the common tasks recommended when configuring CipherTrust Transparent Encryption with DFS(R).

Encrypted Files Under DFSRPrivate Folder

In situations where the LDT Exclusion Registry key was deleted, and the LDT rekeyed files are under the DFSRPrivate folder, following are the steps to reverse this.

On all nodes:

1. Stop DFS(R) service first on both nodes.

2. Add `LDTExclusionGPList` to `vmmgmt/Parameters` with the path to `DFSRPrivate` if not already there.

3. Disable the `DFSRPrivate` GuardPoint.

4. Open the command line as an administrator.

5. Copy the `DFSRPrivate` directory to a temporary backup, type:

   ```
   xcopy "E:\DFSRTest\DFSRPrivate" c:\backupOfDFSRPrivate /E /Y /H /Q /O /K
   ```

6. Delete all files in the `DFSRPrivate`, type:

```
del /S /Q E:\DFSRTest\DFSRPrivate
```

7. Guard `DFSRPrivate` again.

8. Restart the DFS(R) service

9. Test replication of new files.

Double Encryption

A common cause for data corruption is that the data may have become double-encrypted. This can happen if existing encrypted data is written into a GuardPoint because it gets encrypted a second time. You can check for this by copying the data out of the GuardPoint into a clear location with a user who has 'apply_key' rights. Next, mount a GuardPoint on top of the copied data in the clear location using the same policy as the original GuardPoint. If the data then becomes viewable inside that newly mounted GuardPoint, this means that the data was double-encrypted.

To recover the data:

1. Copy all of the double encrypted data into a clear location.

2. Disable the original GuardPoint.

3. Copy the data back into the original location that is now unguarded.

4. Re-guard the original GuardPoint. Data should now be viewable in the original GuardPoint.

Logs

Always download and parse the domain logs before doing anything else. Note the timestamp of certain DFS(R)-related error messages and compare them against similar timestamped log entries on the DFS servers. You may also run `agenthealth` on the DFS server to gather more extensive detail of the domain information.

The `agentinfo` support collection script resides in one of the following paths on systems where CTE agent is installed, depending on version:

```
 C:\program files\vormetric\DataSecurityExpert\agent\vmd\bin
C:\program files\vormetric\DataSecurityExpert\agent\shared\bin
```

The customers DFS environment may detail a lot of issues related to encrypting data, especially in relation to replicated, malformed or open handled data.

Different Keys for Different Folders

You may require that some folders be encrypted with a different key from others, perhaps due to a required SLA. Fencing the data is an excellent method for keeping different enterprise data separate. Normally, DFS nests target folders under one namespace. To encrypt the data, you would use a single key applied to the operational policy. However, if you want to use many keys, target folders must exist under a separate namespace. Windows 2012 and subsequent versions support multiple namespaces. Plan accordingly, allocating resources where necessary and staging target folders appropriately. This is also known as root scalability.

CTE Client Groups

Another consideration is the folders that are under DFS(R) control. If the folders and volumes are identical on all nodes of the DFS topology (i.e. F:\Data) then it is considered best practice to configure all nodes within a DFS topology under a single CipherTrust Client Group. This ensures consistency when apply Guard Paths across the DFS topology.

To create a client group:

1. Open the CTE application.

2. Click **Clients > Client Groups**.

3. Click **Create Client Group**. The Create Client Group dialog box displays.

4. Enter the following:

   - Client Group name: **DFSR**
   - Password creation method: **Generate**
   - Cluster Type: **Non-Cluster**
   - Client profile: The DFSR profile you created
   - Communication Enabled: Slide to activate

5. Click **Next**.

6. On the Add Clients window, select **Client group members** and click **Next**.

7. Select **Inherit Client Group Settings** and click OK.

8. On the Add GuardPoint page, click **Create GuardPoint**.

9. Select the DFSR Policy in the Policy section. Click **Browse** to browse to the host, select the DFSR mount point and DfsrPrivate folder, then click **Create**.

10. Confirm that the GuardPoints are correct and click **Next**.

11. On the confirmation page, verify all is correct and then click **Create**.

Mixed DFS(R) Topologies with Client Groups

In those environments where the DFS(R) folders are different on the individual nodes, then the GuardPoints must be configured for each DFS(R) node. Client groups cannot be deployed as the guard paths on each node will be different. In these cases, Thales recommends stopping the DFS services on this node before guarding the path. For a full Mesh topology, all of the nodes must be transformed and guarded at the same time. The last step after guarding the data is to Start the DFS(R) services.

Secure Start

This section describes encrypting an Microsoft Active Directory (AD) with the Secure Start feature. It contains the following topics:

- Secure Start Overview
- Prerequisites
- Encrypt by Moving the AD Service into a Guarded Directory
- Encrypt Data in Place with Offline Transformation
- Encrypt with an LDT Transformation Policy
- Configure the Time Out Failure
- Recover a Server After it Loses Connection to the Key Manager
- Other Use Cases
- Best Practices for Encrypting and Protecting the AD Service

Secure Start Overview

Secure Start offers data protection for applications which start earlier in the boot sequence than VMD (Vormetric Daemon). For example, the Microsoft Active Directory

(AD) system service starts very early in the boot sequence. To determine if another application qualifies, contact Thales technical support.

Note

- Secure Start is included with CTE. You do not have to purchase it separately.
- Secure Start is supported on Windows Server 2008 R2 and later versions.

There are three methods for encrypting the AD directory:

- Encrypt by Moving the AD Service into a Guarded Directory
- Encrypt Data in Place with Offline Transformation
- Encrypt with an LDT Transformation Policy

Prerequisites

Prior to using Secure Start to guard your AD database:

1. Backup your AD database:
   a. Navigate to **Administrative Tools**.
   b. Click **Windows Server Backup**.
   c. Click **Action > Backup Once**.
   d. Follow the instructions in the Backup Wizard to create a backup of the server in a local drive.

   Note

   When the backup operation completes, it saves the server backup in <*backup drive*>:\WindowsImageBackup\<BackupComputerName>.

2. Perform a system state backup.
3. Obtain the Microsoft DSRM (Data Services Restore Mode) password.
4. Ensure that your AD database is not in `c:\Windows\NTDS`.

Warning

Do not put your AD database in `c:\Windows` or `c:\Program files`. Secure Start cannot encrypt or decrypt any files in those folders.

Encrypt by Moving the AD Service into a Guarded Directory

You can move the AD service into a directory protected by a standard or LDT production policy. This method does not require the initial data transformation step. When you move the AD service into this directory, CTE immediately encrypts the data with either policy.

Note

This step occurs when the system is in DSRM mode, so users have no access to the AD service.

Create the AD GuardPath directory

Create the directory in which the AD service will reside.

1. Log in to the Active Directory Server in DSRM mode using the DSRM password. User ID is Administrator.

2. Create a folder to which you will move the AD database.

Apply Secure Start GuardPoints to a Directory with CipherTrust Manager

To apply Secure Start GuardPoints in **CM**:

1. In the CipherTrust Manager Applications Page, click **CTE > Clients > <clientName>**.

2. Click **Create GuardPoint**.

3. In the Policy field, select a policy.

4. Set Type to **Auto Directory**.

5. Click **Browse**, navigate to, and select, the folder that you just created for the AD database.

6. Select the option: **Secure Start**.

7. Click **Create**.

8. Click **No** to the question, "Would you like to use these GuardPoint settings on another GuardPoint with a different path?" because you are only guarding the AD database.

Verify the Secure Start GuardPoint with CLI

After the policy is pushed to the Active Directory Server, verify the GuardPoints.

To verify the GuardPoints, type:

```
 voradmin ss verify <GuardPoint_path>
Successfully completed the command verify
Success from kernel -Successfully verified the secure start GP
```

Move the AD Database into the Secure Start GuardPoint

Move your AD database from the default location (`c:\windows\NTDS`) to this newly created protected folder. To move the AD database:

1. In DSRM mode, login using the DSRM password. User ID is Administrator.

2. Start NTDSUTIL utility, type:

   ```
   activate instance ntds
   ```

   a. Type:

      files

3. Type:

   ```
   move db to \<GuardPoint>
   ```

4. Type:

   ```
   move logs to \<GuardPoint>
   ```

5. Exit NTDSUTIL utility.

6. Reboot the system into normal mode. The Active Directory Services automatically starts after rebooting.

> **Note**
>
> This step occurs when the system is in DSRM mode, so users have no access to the AD service.

Encrypt Data in Place with Offline Transformation

Encrypting the AD database with a standard (production), or offline policy is very similar to encrypting other data with a standard (production), or offline policy.

The advantage to encrypting data in place is that it saves space. When you copy/move a directory into a guarded directory, you will need twice as much space to store the data because you leave a copy of the data in the original folder, as a precaution, until the original directory has been successfully moved and encrypted. Once the data is transformed, then you can delete the directory that contains the decrypted/clear data.

Using this method, you perform an Initial Data Transformation using the `dataxform` command line utility. During this transformation, access to the GuardPoint data is blocked. After initial transformation, you remove the initial policy, and then apply a production policy, so users can access the data.

Note

This step occurs when the system is in DSRM mode, so users have no access to the AD service.

If your AD service is installed in the default directory, `C:\Windows\NTDS`, you must move it to another directory before you can encrypt it. See Encrypt by Moving the AD Service into a Guarded Directory for more information.

To encrypt the data:

1. In DSRM mode, login using the DSRM password. User ID is Administrator.
2. Create and apply a `dataxform` policy to the GuardPoint directory.
3. Run the `dataxform` command.
4. Remove the `dataxform` policy on the GuardPoint and replace it with a production policy.
5. Reboot out of DSRM mode.

Encrypt with an LDT Transformation Policy

Encrypting the AD database with an LDT policy uses the same steps as encrypting with a standard production policy. The only difference is that you select an LDT policy instead of a standard one. See Encrypt by Moving the AD Service into a Guarded Directory for more information for more information.

Note

If your AD service is installed in the default directory, `C:\Windows\NTDS`, you must move it to another directory before you can encrypt it.

Configure the Time Out Failure

During the initial access to a Secure Start GuardPoint, the CTE agent sets a timer. The default duration is 30 seconds, but you can configure the duration. Minimum duration is one second, maximum duration is 300 seconds.

Data inside the GuardPoint is accessible without CipherTrust Manager connectivity until the timeout is reached. VMD service activates and makes a secure connection to the CipherTrust Manager. After the VMD makes a secure connection, the agent verifies that it is connected to correct CipherTrust Manager. If the VMD fails to connect to the CipherTrust Manager, the timeout is reached, and if AD is installed, the agent shuts down the system for data security purposes.

Note

In DSRM mode, when the timeout occurs, CTE removes the keys from memory. However, CTE does not shut down the system.

In normal mode, CTE shuts down the AD server. For any other application, or if AD is not installed, Secure Start does not shut down the server. However, the data inside the GuardPoint becomes inaccessible until CipherTrust Manager connectivity is restored, or you issue a challenge/response, or password. After the timer has expired, CTE denies any further access to the Secure Start GuardPoint.

1. To configure the timeout duration in seconds, use the `voradmin ss settimeout <timeout>` command. For example:

   ```
    voradmin ss settimeout 220
   Successfully completed the command settimeout
   Successfully set the Secure Start timeout value to 220 seconds
   ```

2. To verify the timeout duration, type:

   ```
    voradmin ss gettimeout
   Successfully completed the command gettimeout
   Secure Start timeout value is set to 220 Seconds
   ```

Recover a Server After it Loses Connection to the Key Manager

Prerequisites

Before rebooting your active directory servers, ensure that CipherTrust Manager connectivity is strong. If it is not strong, restore the CipherTrust Manager connectivity.

> **Note**
>
> When trying to fix a CipherTrust Manager connectivity issue, you can log in to DSRM mode. In DSRM mode, there is no requirement to increase the timeout, because in DSRM mode, the AD system does not shut down after timeout expires.

DSRM Mode

The first method for recovering a server relies on manual CipherTrust Manager connection troubleshooting:

1. Boot into DSRM mode.

2. Attempt to resolve why the server is not connecting to the CipherTrust Manager.

3. Fix that CipherTrust Manager connectivity issue.

4. Reboot into normal mode.

Other Use Cases

Using Secure Start GuardPoints, you can also secure an SQL Server on Microsoft Azure in certain scenarios. SQL system services in Azure also boot earlier in the boot sequence than the VMD (Vormetric Daemon) agent service.

> **Note**
>
> To determine if another application qualifies, contact Thales technical support.

Boot a Windows Server in Azure

To move and guard the AD database, you must boot the AD server into DSRM mode.

To boot a Windows Server 2012/2016 Domain Controller into DSRM remotely in Azure:

> **Note**
>
> The Windows Server 2012/2016 domain controller must be running and accessible through Windows Remote Desktop.

1. Establish a Remote Desktop session on the domain controller.
2. Open an command prompt as Administrator and type:

   ```
   > bcdedit /set safeboot dsrepair
   ```

3. Reboot the domain controller. The Remote Desktop session disconnects.
4. Wait a few minutes, then establish a new Remote Desktop session. The domain controller will be running in DSRM.
5. To reboot into normal mode, open an command prompt as Administrator and type:

   ```
   > bcdedit /deletevalue safeboot
   ```

6. Reboot the domain controller.

Best Practices for Encrypting and Protecting the AD Service

Thales recommends the following best practices when using Secure Start with an AD service.

Access Control with Secure Start

User can setup a restricted access control policy with encryption to prevent the unauthorized access of AD database files. The restricted policy with Secure Start:

- Prevents a rogue user from logging into the system, and moving or copying the AD database files to another directory and tampering with it.
- Denies permissions, after you setup and guard files, so that no one can move a file from the guarded directory. Plus it restricts any other unwanted/unnecessary process or users from tampering with AD files.
- Provides permission for an authorized user who needs access to AD services and files.

Creating a Minimal Policy Required for AD with Access Control

When creating a normal, strict policy for access control, you must allow access to the following processes and directories for Active Directory.

Processes

```
secfsd.exe (C:\Program
Files\Vormetric\DataSecurityExpert\agent\secfs\ sec\bin\)
lsass.exe (C:\Windows\System32\)
vds.exe (C:\Windows\System32\)
vssvc.exe (C:\Windows\System32\)
wbengine.exe (C:\Windows\System32\)
ntoskrnl.exe (C:\Windows\System32\)
```

Users

`NT AUTHORITY\SYSTEM`

To create a minimal policy:

1. Create a User Set named **AD_Minimum_User_Set** with the following parameters:

ID	Uname	osDomains
1	SYSTEM	NT AUTHORITY

2. Create a Process Set named: **AD_Process_Set** with the following parameters:

ID	Directory	Base Name
1	C:\Program Files\Vormetric\DataSecurityExpert\agent\secfs\sec\bin	secfsd.exe
3	c:\Windows\System32|ntoskrnl.exe	
4	c:\Windows\System32|vds.exe	
5	c:\Windows\System32|vssvc.exe	
6	c:\Windows\System32|wbengine.exe	
7	c:\Windows\System32|lsass.exe	

3. Create a Security rule set with the following parameters:

Order	User	Process	Action	Effect	Browsing
1	AD_Minimum_User_Set	AD_Process_Set	all_ops	Audit, Permit, Apply	key
2				Audit, Deny	Yes

Creating a Restricted Policy in DSRM Mode

Create the following policy for the initial transformation of an AD database in DSRM mode. The policy allows access to the local administrator.

In DSRM mode, you use the `NTDSUTIL` utility to perform maintenance for an Active Directory.

To create a restricted policy:

1. Create a User Set named **AD_Minimum_User_Set** with the following parameters:

ID	uname	osDomains
1	SYSTEM	NT AUTHORITY
2	Administrator	localhost

2. Create a Process Set named: **AD_Process_Set** with the following parameters:

ID	Directory	Base Name
1		secfsd.exe

ID	Directory	Base Name
	C:\Program Files\Vormetric\DataSecurityExpert\agent\secfs\sec\bin	
2	c:\Windows\System32\	ntdsutil.exe
3	c:\Windows\System32\	ntoskrnl.exe
4	c:\Windows\System32\	vds.exe
5	c:\Windows\System32\	vssvc.exe
6	c:\Windows\System32\	wbengine.exe
7	c:\Windows\System32\	lsass.exe

3. Create a Security rule with the following parameters:

Order	User	Process	Action	Effect	Browsing
1	AD_Minimum_User_Set	AD_Process_Set	all_ops	Audit, Permit, Apply key	Yes
2				Audit, Deny	Yes

Guard Directories

The best practice for guarding a directory with a Secure Start GuardPoint is to:

1. Create a directory.

2. Guard that directory with a standard production or LDT policy. Follow the steps in Apply Secure Start GuardPoints to a Directory with CipherTrust Manager.

3. Move the AD service into that directory.

Perform Subsequent System State Backups

After you move an AD service into a guarded directory, or out of a guarded directory:

1. Perform another system state backup.

2. Save this subsequent backup to a different location.

Exchange DAG

This section describes encrypting email databases using Microsoft Exchange database availability group (DAG). It contains the following topics:

- Exchange DAG Overview
- CTE Policies for Exchange DAG
- Encrypting with CTE-LDT in an Exchange DAG Environment
- Encrypting with a Standard CTE Policy in the Exchange DAG Environment
- Decrypting with CTE-LDT in an Exchange DAG Environment

Exchange DAG Overview

A DAG is a high-availability (HA) and data-recovery feature of the Microsoft Exchange Server. A DAG, which can consist of up to 16 Exchange mailbox servers, automates recovery at the database level after a database, server or network failure. You can now use CTE for Windows to encrypt Exchange DAG mailboxes.

You can encrypt the Exchange databases with a standard (offline) policy or an CTE-Live Data Transformation (CTE-LDT) policy. In an offline policy, users cannot access the database during initial data encryption. With a CTE-LDT policy, CTE encrypts the data while users and applications are accessing the files. CTE-LDT is used for Initial data transformation as well as transparent encryption and decryption.

Note

For more information about CTE-LDT and standard data transformation, see *CTE-Live Data Transformation with CipherTrust Manager, CTE-Live Data Transformation with Data Security Manager*, or the *CTE Data Transformation Guide*.

Supported Use Cases for CTE in an Exchange DAG Environment

CTE has been tested by Thales in the following scenarios:

- Initial data transformation of Exchange databases using either CTE-Live Data Transformation or standard data transformation.

- Transparent encryption or decryption of the Exchange database on DAG nodes.
- Key rotation using a CTE-LDT policy.
- Adding a new node to the Exchange DAG Environment.

  Thales has only tested an Exchange DAG environment with two nodes, however, Thales does **not** anticipate any issues with using more than two nodes.

Thales also tested the following Exchange DAG operations during the above scenarios:

- Failover/Failback of databases from one node to another node and making both databases active on each node.
- Adding new Databases to the existing nodes.

Unsupported Use Cases

The following scenarios are not supported:

- Using different encryption keys on Exchange DAG nodes; both nodes must use the same encryption key
- The encryption of Exchange Binaries.
- Using nodes in a different subnet, data center, or site. (Thales is not testing this scenario, but we do not believe it will cause any issues.)

CTE Policies for Exchange DAG

The CTE policies you need depend on the type of encryption you will be using.

- When you use CTE-LDT encryption, you only need to create one Live Data Transformation policy. This policy will be used for both the initial data encryption and guarding the data in production. CTE-LDT requires a versioned CBC or CBC_CS1 key in order to perform automatic key rotation.
- When you use standard encryption, you need to create two policies:
  - The **initial encryption** policy specifies the current encryption key (if any) and the encryption key you want CTE to use when it encrypts the data. This policy also denies access to any other process trying to access the GuardPoint.

    You apply the initial encryption policy when you first create the GuardPoint, and you leave it in place until all of the data has been encrypted. After that, you remove this policy from the GuardPoint.

- ◦ The **production policy** specifies the same encryption key as the initial encryption policy along with any security rules you want to use to protect your data in production. After the initial encryption has completed, you apply the production policy to the GuardPoint and allow users and applications to access the now-protected data.

> **Note**
>
> There are no special CTE policy requirements for Exchange DAG with either CTE-LDT or Standard encryption. Therefore, you can use the same policies in an Exchange DAG environment that you use for any other CTE-protected directory.
>
> The only special requirement for Exchange DAG is the guard path you specify when you create the GuardPoint. You must guard the Mailbox directory only. Do not guard above or below the Mailbox directory. For details, see Encrypting with CTE-LDT in an Exchange DAG Environment or Encrypting with a Standard CTE Policy in the Exchange DAG Environment.

How you create these policies depends on the key manager that you are using. For details, see one of the following:

- Creating a Policy for CTE-LDT Encryption with CipherTrust Manager
- Creating Policies for Standard Encryption with CipherTrust Manager

Creating a Policy for CTE-LDT Encryption with CipherTrust Manager

When you use CTE-LDT encryption, you only need to create one policy. This policy will be used for both the initial data encryption and guarding the data in production. CTE-LDT requires a versioned CBC or CBC_CS1 key in order to perform automatic key rotation. For details, see the *CTE-Live Data Transformation with CipherTrust Manager* guide for the version of CTE that you are using.

1. Log into CipherTrust Manager and launch the **CTE** application.
2. In the left-hand menu bar, click **Policies**.
3. Click **Create Policy**.

4. For **Name**, make sure you use a name that clearly designates this as a CTE-LDT policy. You will need to be able to find this policy name from the list of all available policies when you create the GuardPoint.

5. For **Policy Type**, select **Live Data Transformation**.

6. Click **Next** to go to the Security Rules page. CipherTrust Manager should have automatically added a security rule for **Action**: `key_op`, **Effect**: `permit,applykey`. If this security rule is not there, click **Back** and make sure you have selected **Live Data Transformation** in the **Policy Type** field.

7. Enter any other security rules you want to use based on your production environment requirements. You can add as many security rules as you need to define who should have access to the protected data.

   For more information about the type of rules you may want to use, or ways to exclude some data from encryption, see the *CTE-Live Data Transformation with CipherTrust Manager* guide for the version of CTE that you are using.

8. When you are done specifying your security rules, click **Next** to go to the Key Rules page.

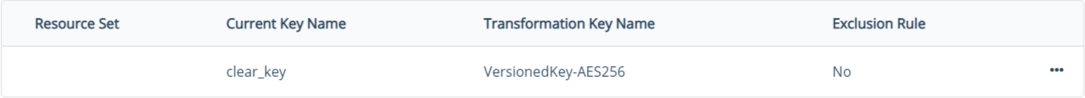
9. Click **Create Key Rule** and enter the following information:

   - In the **Current Key Name** field, click **Select** to specify the current encryption key used for the data. If the data is unencrypted, specify `clear_key` as the encryption key.

   - In the **Transformation Key Name** field, click **Select** to specify the versioned encryption key you want to use to encrypt the data. When you are done, click **Add**.

   > **Tip**
   >
   > You can also create a new key at this point if desired. For details on creating an encryption key, see your CipherTrust Manager documentation.

   For example:



Resource Set	Current Key Name	Transformation Key Name	Exclusion Rule	
	clear_key	VersionedKey-AES256	No	•••

10. Click **Next** to go to the Confirmation page.

11. Verify your selections and click **Save** to save the policy.

Creating Policies for Standard Encryption with CipherTrust Manager

When you use standard encryption, you need to create two policies:

- The *initial encryption* policy specifies the current encryption key (if any) and the encryption key you want CTE to use when it encrypts the data. This policy also denies access to any other process trying to access the GuardPoint.

  You apply the initial encryption policy when you first create the GuardPoint, and you leave it in place until all of the data has been encrypted. After that, you remove this policy from the GuardPoint.

- The *production* policy specifies the same encryption key as the initial encryption policy along with any security rules you want to use to protect your data. After the initial encryption has completed, you apply the production policy to the GuardPoint and allow users and applications to access the now-protected data.

Creating the Initial Encryption Policy

1. Log into CipherTrust Manager and launch the **CTE** application.

2. In the left-hand menu bar, click **Policies**.

3. Click **Create Policy**.

4. For **Name**, make sure you use a name that clearly designates this as an initial-encryption policy and not a production policy. You will need to be able to find this policy name from the list of all available policies when you create the GuardPoint.

5. For **Policy Type**, select **Standard**.

6. Enable the **Data Transformation** check box.

7. Click **Next** to go to the Security Rules page. CipherTrust Manager should have automatically added a security rule for **Action**: `key_op`, **Effect**: `permit,applykey`. If this security rule is not there, click **Back** and make sure you have enabled the **Data Transformation** check box.

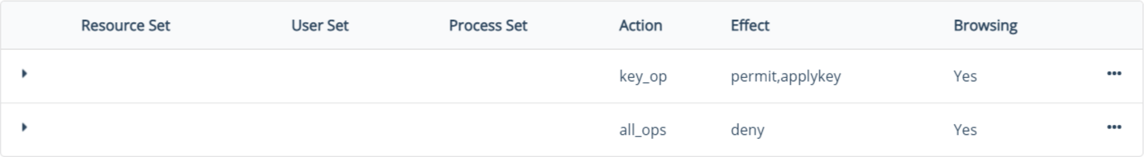
8. Click **Create Security Rule** and do the following:

   a. In the **Action** field, select `all_ops`.

   b. In the **Effect** field, select `deny`.

   c. Click **Add** to return to the Security Rules page.

   You should now have two security rules, as shown:

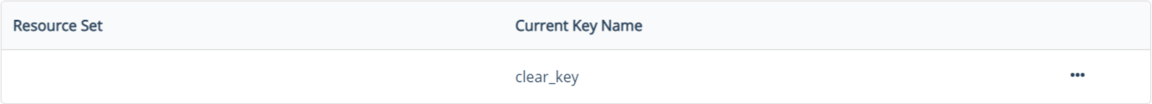


	Resource Set	User Set	Process Set	Action	Effect	Browsing	
▸				key_op	permit,applykey	Yes	•••
▸				all_ops	deny	Yes	•••

9. Click **Next** to go to the Key Rules page.

10. Click **Create Key Rule**.

11. In **Current Key Name**, click **Select** to specify the current encryption key used for the data. If the data is unencrypted, specify `clear_key` as the encryption key. When you are done, click **Add**. For example:



Resource Set	Current Key Name	
	clear_key	•••

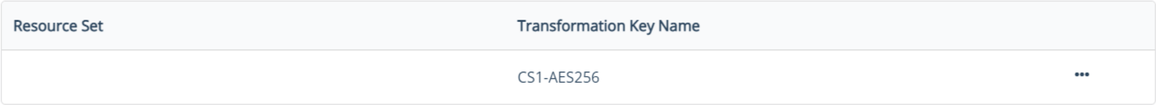
Tip

You can also create a new key at this point if desired. For details on creating an encryption key, see your CipherTrust Manager documentation.

12. Click **Next** to go to the Data Transformation page.

13. Click **Create Data Transformation Rule**.

14. In the **Transformation Key Name** field select the encryption key you want to use to encrypt the data. This key must match the one specified in the production policy you intend to apply to the GuardPoint after the data has been encrypted. For example, if you want to encrypt the data with the key CS1_AES256, you would specify the following transformation rule:



Resource Set	Transformation Key Name	
	CS1-AES256	•••

15. Click **Next** to go to the Confirmation page.

16. Verify your selections and click **Save** to save the policy.

Creating the Production Policy

1. Launch the **CTE** application.

2. In the left-hand menu bar, click **Policies**.

3. Click **Create Policy**.

4. For **Name**, make sure you use a name that clearly designates this as a production policy and not an initial encryption policy. You will need to be able to find this policy name from the list of all available policies when you create the GuardPoint.

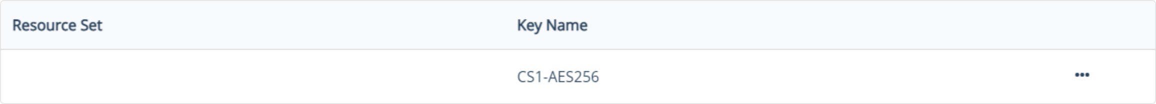
5. For **Policy Type**, select **Standard**.

6. Click **Next** to go to the Security Rules page. Enter the security rules you want to use based on your production environent requirements. You can add as many security rules as you need to define who should have access to the protected data.

7. When you are done, click **Next** to go to the Key Rules page.

8. Click **Create Key Rule**.

9. In **Key Name** field, click Select to specify the encryption key used to transform the data in the initial encryption policy. When you are done, click **Add**. For example:

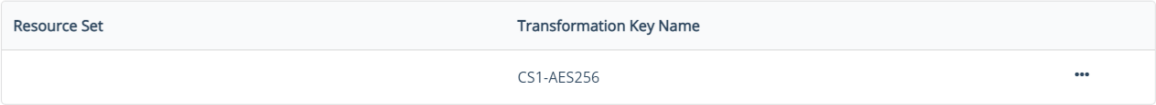


Resource Set	Key Name	
	CS1-AES256	•••

10. Click **Next** to go to the Data Transformation page.

11. Click **Create Data Transformation Rule**.

12. In the **Transformation Key Name** field select the encryption key you want to use to encrypt the data. This key must match the one specified in the production policy you intend to apply to the GuardPoint after the data has been encrypted. For example, if you want to encrypt the data with the key CS1_AES256, you would specify the following transformation rule:



Resource Set	Transformation Key Name	
	CS1-AES256	•••

13. Click **Next** to go to the Confirmation page.

14. Verify your selections and click **Save** to save the policy.

Encrypting with CTE-LDT in an Exchange DAG Environment

Prerequisites

Before you can start the CTE-LDT data encryption process, you need to:

- Create or identify the CTE policy you want to use for data encryption. CTE-LDT uses a single Live Data Transformation policy for both initial encryption and subsequent rekeys, so the policy you use should have all the access control rules you want to use for your data when it is in production. For details, see CTE Policies for Exchange DAG.
- Set your Quality of Service (QoS) settings. QoS enables administrators to manage and control CTE-LDT impact to application workloads by monitoring and controlling the use of host system resources, such as memory or I/O utilization, during data transformation.

For details about using CTE-LDT, see *CTE-Live Data Transformation with Data Security Manager* or *CTE-Live Data Transformation with CipherTrust Manager* for the version of CTE that you are using.

Procedure

1. In the **Exchange Admin Center**, make Exchange node 1 the primary node. Make node 1 the active node and move all of the databases to that node.

2. Make all of the databases active on node 1.

3. Suspend all databases on node 2. Wait for 2-3 minutes for the database to finish with replication so the database will be suspended.

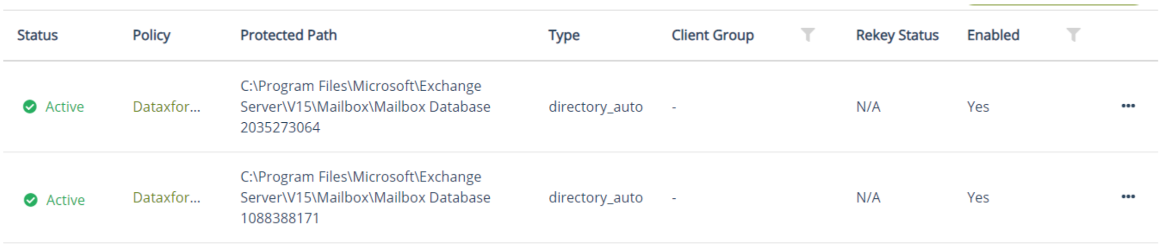
Warning

Make sure that all of the Exchange services in node 2 are down and not accessing the Exchange databases. All Exchange Services must be stopped, all databases must be suspended, and all data replication between the nodes must be stopped. Any file access on the node during the encryption process could cause data corruption.

4. When you are certain that all Exchange DAG services have been suspended on node 2, create the GuardPoints you want to use on node 2 with the appropriate Live Data Transformation policy. When you create the GuardPoints:

   - Make sure you are guarding each host individually. Do not assign the GuardPoints using a Host or Client Group because you only want these GuardPoints to exist on node 2 at this point.
   - **Imporant**: When you specify the guard path, only guard the Mailbox Database. Do *not* guard at a higher or lower directory. For example:
     - **Correct**: `C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 1088388171\`
     - **Incorrect**: `C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 1088388171\Inbox\` — This is *not* correct because it's below the mailbox database directory.
   - Make sure that **Secure Start** is on for the GuardPoints.

   The following example shows two correctly-specified GuardPoints in CipherTrust Manager:



Status	Policy	Protected Path	Type	Client Group	Rekey Status	Enabled	
✓ Active	Datafor...	C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 2035273064	directory_auto	-	N/A	Yes	•••
✓ Active	Datafor...	C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 1088388171	directory_auto	-	N/A	Yes	•••

   Live data transformation on node 2 begins as soon as the GuardPoints become active on node 2.

5. Wait until CTE-LDT has finished transforming the data in all GuardPoints on node 2.

6. In the **Exchange Admin Center**, go to the Exchange Database tab and **Resume** all Passive database copy on node 2.

7. Wait for the server to move to the healthy state. If it does not, wait for some more time for the Content Index state to change to Healthy.

   Warning

   It may take a few minutes for the Exchange Service to resync. Monitor the Exchange logs on the system and make sure that replication is working. Make sure that database replication finishes and databases are in a healthy state before proceeding.

8. In the **Exchange Admin Center**, move all of the databases from node 1 to node 2.

   Now the databases on node 1 are mounted as passive. All databases on node 2 are mounted as active.

9. Create the same GuardPoints on node 1 that you created on node 2.

Warning

- Make sure that all GuardPoints on node 1 are identical to those on node 2.
- You must guard the same databases with the same Live Data Transformation Policy and the same encryption key on both nodes.

Encrypting with a Standard CTE Policy in the Exchange DAG Environment

Prerequisites

Before you can start the standard (offline) data encryption process, you need to:

- Decide if you will be using the copy/restore method or the *CTEdataxform* utility in order to perform the initial encryption. For details about these methods and their specific benefits and limitations, see the *CTE Data Transformation Guide* for the version of CTE that you are using.
- Create or identify the encryption key that you want to use for the initial data encryption.
- Create or identify the Standard policies that you want to use for the initial data encryption and for protecting the data in production after it has been initially encrypted. For details, see Creating Policies for Standard Encryption with CipherTrust Manager.

Procedure

1. In the **Exchange Admin Center**, make Exchange node 1 the primary node.
   This means that node 1 is mounted as the active node and node 2 is mounted as the passive node.

2. Make all of the databases active on Exchange node 1.

3. Go to the Exchange Database tab and suspend all database on node 2.
   Make sure that all of the exchange database services in node 2 are down and not accessing the Exchange databases. This process can take several minutes.

   Warning

   Make sure that all of the Exchange services in node 2 are down and not accessing the Exchange databases. All Exchange Services must be stopped, all databases must be suspended, and all data replication between the nodes must be stopped. Any file access on the node during the encryption process could cause data corruption.

4. When you are certain that all Exchange DAG services have been suspended on node 2, create the GuardPoints you want to use on node 2 with the appropriate Standard data transformation policy that you want to use for the initial data encryption. When you create the GuardPoints:

   - Make sure you are guarding each host individually. Do not assign the GuardPoints using a Host or Client Group because you only want these GuardPoints to exist on node 2 at this point.

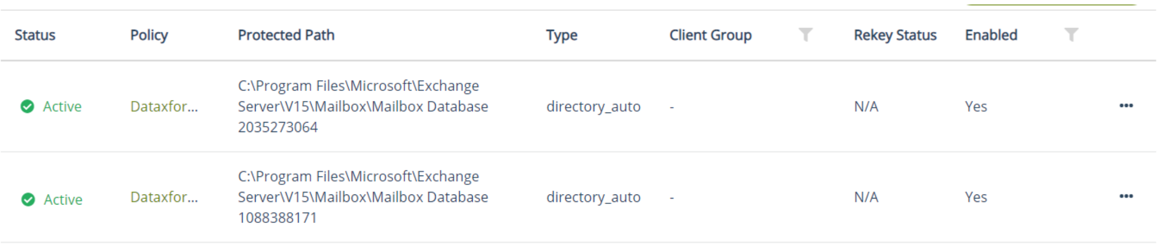
   - **Important**: When you specify the guard path, only guard the Mailbox Database. Do not guard at a higher or lower directory. For example:

     - **Correct**: `C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 1088388171\`

     - **Incorrect**: `C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 1088388171\Inbox\` — This is *not* correct because it's below the mailbox database directory.

   - Make sure that **Secure Start** is on for the GuardPoints.

   The following example shows two correctly-specified GuardPoints in CipherTrust Manager:



Status	Policy	Protected Path	Type	Client Group	Rekey Status	Enabled	
✓ Active	Dataxfor...	C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 2035273064	directory_auto	-	N/A	Yes	•••
✓ Active	Dataxfor...	C:\Program Files\Microsoft\Exchange Server\V15\Mailbox\Mailbox Database 1088388171	directory_auto	-	N/A	Yes	•••

5. After all GuardPoints on node 2 have been enabled, run the `dataxform` utility for each GuardPoint:

```
dataxform --rekey --print_stat --gp <directory>
```

6. After the data transformation is finished, unguard each mailbox on node 2, then re-guard each mailbox on node 2 with the appropriate Production policy.

   Note

   Use the same Key/Policy on both nodes.

7. In the **Exchange Admin Center**, go to the Exchange Database tab and resume all databases on node 2.

After a few minutes, all nodes should become Healthy.

> **Warning**
>
> **It may take a few minutes for the Exchange Service to resync. Monitor the Exchange logs on the system and make sure that replication is working. Make sure that database replication finishes and databases are in a healthy state before proceeding.**

8. In the **Exchange Admin Center**, try to move a database from node 1 to node 2. If the data move is successful this means that node 2 is mounted as the active node and node 1 is mounted as the passive node.

9. Create the same GuardPoints on node 1 that you created on node 2. Make sure that all GuardPoints on node 1 are identical to those on node 2.

> **Warning**
>
> **You must guard the same databases with the same Standard Policy and the same encryption key on both nodes.**

Decrypting with CTE-LDT in an Exchange DAG Environment

Prerequisites

- Make sure that the LDT state is set to REKEYED before unguarding.
- Make sure that all of the files inside the GuardPoint are at the same version of the key.
    - Run the LDT report to find the version:

      ```
      voradmin ldt report <GuardPoint path> [<logfile>]
      ```

    - Run the Key map report to find the version:

      ```
      voradmin ldt key [report|map] <key_name, version> <GuardPoint path>
      ```

Procedure

1. Make sure that all of the Exchange services in node 2 are down and not accessing the Exchange databases.

   > **Note**
   >
   > Suspension can take 2-3 Minutes.

2. In the **Exchange Admin Center**, make Exchange node 1 the primary node.

   This means that node 1 is mounted as the active node and node 2 is mounted as the passive node.

3. Make all of the databases active on Exchange node 1.

4. Go to the Exchange Database tab and suspend all databases on node 2.

5. Unguard the database folders that you previously guarded on node 2.

6. Delete all of the metadata on all of the database folders on node 2, type:

   ```
   voradmin ldt attr delete [<file name path> | <guard path>]
   ```

7. Guard with an LDT policy set for Encryption to Clear on node 2.

   > **Note**
   >
   > You must clone the current version of the encryption key to use as the current key in the new LDT policy and `clear_key` as the transformation key.

8. Go to the Exchange Database tab and resume all databases on node 2.

   > **Note**
   >
   > After a few minutes, the databases should become healthy automatically. If not, wait for the LDT process to decrypt the data. Make sure that all of the data is transformed back to clear and that the LDT state is set to **REKEYED**.

9. Move the database from node 1 to node 2.

10. Repeat this procedure for node 1.

11. After both nodes are rekeyed and transformed from encryption to clear, unguard them:

    a. In the **Exchange Admin Center**, make Exchange node 1 the primary node. This means that node 1 is mounted as the active node and node 2 is mounted as the passive node. 2. Make all of the databases active on Exchange node 1.

    b. Go to the Exchange Database tab and suspend all databases on node 2.

    c. Unguard the database folders that you previously guarded on node 2.

    > **Warning**
    >
    > **Always ensure that you are unguarding a passive node.**

    d. Repeat this procedure for Node 1.

Storage Spaces Direct

This chapter describes how CTE integrates with Windows Storage Spaces Direct (S2D) hyper-converged clusters. It contains the following sections:

- S2D Overview
- Deployment Options
- Supported Use Cases

S2D Overview

S2D uses industry-standard servers with local-attached drives to create high-availability (HA) software-defined storage. SD2 is included in Windows Server 2019 Data center and Windows Server 2016 Data center, both of which are supported by CTE.

S2D extends the stack of usable storage devices to storage devices such as SATA and SAS HDD's, SSD's and NVMe (Non-Volatile Memory Express) disks to create shared disk volumes. S2D supports clusters of a minimum of two nodes, and a maximum of 16 nodes and 400 drives. S2D aggregates the available storage into a Storage Pool.

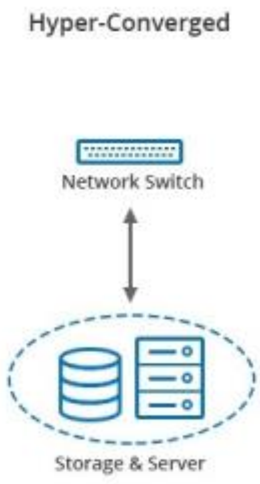
The hyper-converged deployment option runs virtual machines on the servers providing the storage.

A complete description of the S2D product, and instructions on how to set up the S2D environment is available on the Microsoft website at https://docs.microsoft.com/en-us/windows-server/storage/storage-spaces/storage-spaces-direct-overview

Deployment Options

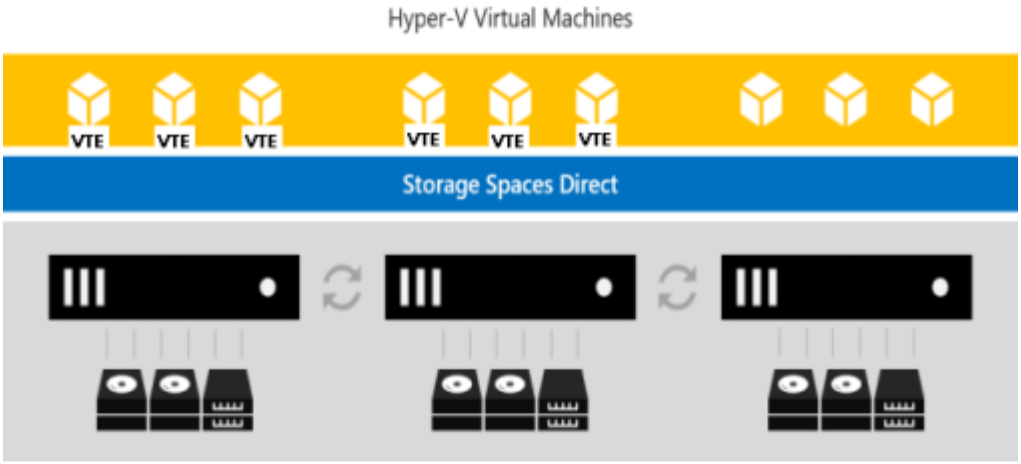
CTE supports S2D in a hyper-converged infrastructure where computing and storage components are in a single cluster as shown in the following figure.

Hyper-converged infrastructure



Hyper-converged with S2D and CTE virtual machines run on the servers providing the storage. In the following figure, CTE is installed inside 6 of the VMs to protect the data.

High-Level view of S2D



You can use all the capabilities of CTE to protect the data in the VMs in a S2D hyper-converged deployment. These capabilities are described in Supported Use Cases.

Supported Use Cases

Thales tested only Hyper-converged deployments in the following scenarios.

- Initial Data Transformation of data using:
  - Live Data Transformation
  - Offline Data Transformation
- Transparent Encryption/Decryption of structure and unstructured data
- Key rotation using a Live Data Transformation policy

Using CTE with Quantum StorNext

This chapter describes how to configure CTE and Quantum StorNext devices to interoperate to allow CTE policies to apply to storage managed by Quantum StorNext.

This section contains the following topics:

Overview of using CTE with Quantum StorNext

Quantum StorNext Fibre Channel-connected devices provide shared file access to third party storage for workstation clients and are optimized for simultaneous access to very large files such as video files. The Quantum StorNext file system is known as SNFS or by its older name, CVFS.

You can encrypt and control access to SNFS files with policies by installing CTE Agents on Windows clients that are configured for access to the SNFS file system. Some limitations apply to this integration, such as supported operating systems, supported SNFS features, concurrent read/write access by multiple clients, and GuardPoint settings (see the next section for more information about these limitations).

CTE and Quantum StorNext Compatibility

The following sections list the supported operating systems and CTE settings supported for use with Quantum StorNext file systems. Important unsupported configuration parameters are also listed.

Supported StorNext Server and Client Configurations

The CTE integration with SNFS file systems works only with certain SNFS versions, SNFS storage policies, and client operating systems.

Configuration parameter	Windows
StorNext (SNFS) operating system version	6.x
StorNext metadata controller (MDC) server OS type	Windows MDC supported
StorNext replication policy	Not supported
StorNext deduplication policy	Not supported
StorNext truncation	Supported
StorNext full and partial backup	Supported
StorNext expand file system	Supported
StorNext data migration	Supported
StorNext read-ahead cache	Disable for use with CTE
Client operating systems	Windows Server 2012 R2, Windows Server 2016
StorNext LAN client	DLC
StorNext mount method: locally mounted directory	Supported
StorNext mount method: CIFS	Not supported
StorNext mount method: NFS	Not supported

Supported GuardPoint and Key Settings for SNFS File Systems

When configuring CTE GuardPoints and keys for SNFS, keep in the mind the compatibility limitations listed in the following table.

Configuration element	Windows
Offline data transformation	Supported
Live Data Transformation (LDT)	Not supported
Key manager compatibility	See the *Compatibility Matrix for CTE Agent with Data Security Manager or the Compatibility Matrix for CTE Agent with Data Security Manager* for your CTE version
Guard unstructured data	Supported
Guard structured data	Not supported
GuardPoint type: Directory (including entire SNFS volume)	Supported
GuardPoint type: Raw device	Not supported
GuardPoint type: Block device	Not supported
GuardPoint mount option: manual guard	Not applicable
GuardPoint mount option: auto guard	Supported
GuardPoint mount option: automount	Not supported
AES-CBC key type	Supported
AES-CBC-CS1 key type	Not supported

Supported Concurrent Access Read/ Write Scenarios

If you want to allow access by multiple clients (users) to CTE-protected SNFS files under the same GuardPoint, just read-only access is supported. StorNext file locking is not implemented in CTE, so there is currently no way to prevent concurrent conflicting writes to the same file. As a result, Thales does not support write access to the same GuardPoint from multiple clients.

To enable read access to the same GuardPoint from multiple clients, ensure that all clients are configured to use the same policy and key.

Configuration parameter	Windows
Read/write access from a single LAN client to a GuardPoint	Supported
Read/write access from two or more LAN clients to the same GuardPoint	Not supported
Read-only access from one, two, or more LAN clients to the same GuardPoint	Supported

Setting up CTE and Quantum StorNext Integration

For the most part, CTE integration with Quantum StorNext is the same as for any standard file system. The next section provides an overview of the steps involved in making CTE work with SNFS. Later sections provide more information about the steps that are new or differ significantly from a typical CTE setup.

Integration Task Overview

The table below provides an overview of the steps involved in setting up SNFS and CTE to work together. As noted in the table, some of these tasks are described in the documentation for your selected key manager. Some of these steps may need to be performed by other staff members at your organization if you have divided the security administration duties as recommended by Thales and you don’t have access to the key manager.

Task	Key configuration notes	For more information
Install and configure a Quantum StorNext MDC server for use with CTE	Disable the StorNext read-ahead cache. Only certain StorNext policies, features, and mount types are supported. See Supported StorNext Server and Client Configurations.	See Installing and Configuring a Quantum StorNext MDC Server for Use with CTE.
Install and configure Quantum StorNext clients for use with CTE	Only certain operating systems are supported. See Supported	See Installing and configuring Quantum StorNext DLC Clients for Use with CTE.

Task	Key configuration notes	For more information
	StorNext Server and Client Configurations.	
Create a domain for one or more SNFS hosts, or add them to an existing domain	No difference from standard CTE agent configuration.	See “Domain Management” in your key manager documentation.
Add the host to the key manager	No difference from standard CTE agent configuration.	See “Configuring Hosts and Host Groups” in your key manager documentation.
Install and register the CTE Agent on the host system	No difference in installation.	See Getting Started with CTE for Windows
Create encryption keys (optional)	AES-CBC-CS1 keys are not supported on Windows. See the note in Supported GuardPoint and Key Settings for SNFS File Systems.	See “Managing Keys” in your key manager documentation. For information about AES-CBC-CS1 keys, see Enhanced Encryption Mode.
Configure host groups containing one or more StorNext LAN clients (optional)	No difference from standard CTE agent configuration.	See “Configuring Hosts and Host Groups” in your key manager documentation.
Configure policies (including user, process, and resource sets) to control access or enable encryption	No difference from standard CTE agent configuration.	See “Configuring Policies” in your key manager documentation.
Configure one or more GuardPoints	Some GuardPoint settings are not supported. See Supported GuardPoint and Key Settings for SNFS File Systems.	See “Managing GuardPoints” in your key manager documentation

Installing and Configuring a Quantum StorNext MDC Server for Use with CTE

Install and configure a Quantum StorNext metadata controller (MDC) server using the Quantum StorNext documentation as a guide. The CTE integration works with

Windows StorNext MDCs. Ensure that you configure the StorNext server to work with the settings supported by CTE as listed in Supported StorNext Server and Client Configurations. For example, you must disable the StorNext read-ahead cache and only certain StorNext policies, features, and mount types are supported.

Installing and Configuring Quantum StorNext DLC Clients for Use with CTE

Install and configure Quantum StorNext DLC clients using the Quantum StorNext documentation as a guide. The CTE integration works with Windows StorNext DLCs.

Ensure that you configure DLC clients to work with the settings supported by CTE as listed in Supported StorNext Server and Client Configurations. For example, only certain operating systems are supported.

Note

Just read-only access is supported if multiple StorNext LAN clients will access files in the same GuardPoint. For more information, see Supported Concurrent Access Read/Write Scenarios.

Choosing a Mounting Method

There are two methods for mounting a StorNext file system on Windows in the StorNext Client Configuration application:

- Map to Drive Letter
- Map to Directory

Both methods are supported in CTE. If you mount the StorNext file system using the Map to Directory method, you must create the directory on the Windows computer before assigning that directory in the Client Configuration application. For example, the default Map to Directory folder is `C:\Mount\snfs1`. If you use that default, you must create `C:\Mount\snfs1` before mounting the StorNext file system in the Client Configuration application.

If you change mounting methods (drive letter to directory or vice versa), you may need to close and reopen Windows Explorer or reboot the computer for the change to take effect.

Installing the CTE Agent on Each StorNext LAN client

Install a CTE Agent on each computer that is set up as a StorNext LAN client and for which you want to set policies. For supported operating systems, see the table in Supported StorNext Server and Client Configurations.

Use any installation method supported for your operating system. For details, see Getting Started with CTE for Windows.

Encrypt Microsoft OneDrive files with CTE

This document describes how to encrypt files in Microsoft OneDrive with CTE.

Use Cases

- Encrypt Microsoft OneDrive files with CTE for **standard** policies on a **Windows server**

Test Environment

- CTE Agent: 7.4.0

- CipherTrust Manager: 2.10.0

- OS: Windows 10 Azure VM (OneDrive is built into Windows 10)

- OneDrive setup (Host VM/OS) for other Windows platforms:

  1. Download the OneDrive Application from Microsoft and install it.

  2. Follow the steps in Sync files with OneDrive in Windows to sync the files using OneDrive.

Steps

To integrate CTE with Microsoft OneDrive:

1. Deploy a CipherTrust Manager in Microsoft Azure
2. Install and Register the CTE Agent
3. Encrypt data on a OneDrive folder with CTE

> **Note**
>
> Microsoft OneDrive is already configured in the Azure VM environment and can be seen in File Explorer. View files in your OneDrive cloud from OneDrive.live account.

Deploy a CipherTrust Manager in Microsoft Azure

- Deploy a CipherTrust Manager in Microsoft Azure

Install and Register the CTE Agent

1. Install CTE Agent on a client system where OneDrive is mounted. Refer to CTE Agent Quick Start Guide for details.
2. Register the CTE Agent with the CipherTrust Manager deployed in Azure.

Encrypt data on a OneDrive folder with CTE

OneDrive Files On-Demand allows you to access all of your files in your OneDrive cloud storage without having to download them and use local storage space. If this is enabled, the files and folders are shown as Reparse Points locally. You can still see all of your files as online-only files in File Explorer, but they do not use local space. When you are connected to the Internet, you can use the files like every other file on your device.

Note

- Before guarding, enabling, disabling, or unguarding, pause syncing on OneDrive.
- For the `vmlfs driver`, the Files-On-Demand feature of OneDrive **must be disabled**. For the `vmfiltr driver`, the Files-On-Demand feature of OneDrive does not need to be disabled.

Use Case 1: Encrypting new data in OneDrive

When creating a GuardPoint for new data, use a Standard Encryption policy to encrypt this data.

Policy Type		Standard
Security Rules		
Line 1	Effect	Audit, Permit
Line 1	Action	all_ops
Process Set	OneDrive	C: \Users\winAdministrator\AppData\Local\Microsoft\OneDrive\OneDrive.exe, C: \Windows\sysWOW64\OneDrive.exe
Security Rules		
Line 2	Effect	Audit, Permit, ApplyKey
Line 2	Action	all_ops

After you create and guard a folder in OneDrive:

1. Create/copy files that should be encrypted to a backup file. The files, when viewed online are in encrypted form. Same files in VM (locally) – in clear.
2. After disabling the GuardPoint:
   - Files in OneDrive cloud – encrypted
   - Same files in VM (locally) – encrypted
3. After enabling the GuardPoint:
   - Files in OneDrive cloud – encrypted
   - Same files in VM (locally) – in clear

Use Case 2: Encrypting already existing data in OneDrive

For data that already exists in a OneDrive, you have to:

1. Create the GuardPoint and apply an initial transformation policy.

2. When initial transformation is finished, apply the Production policy.

3. Resume the OneDrive/CTE synchronization.

Policy Type	**Data Transformation: Initial Transformation**	
Security Rules:		
1	Effect	ApplyKey,Audit,Permit
1	Action	Key_op
2	Effect	Deny, Audit
2	Action	all_ops
Key Selection Rules	Clear_key	
Data Transformation rules	key1	

Policy Type		**Data Transformation: Production**
Security Rules		
Line 1	Effect	Audit, Permit
Line 1	Action	all_ops
Process Set	OneDrive	C:\Users\winAdministrator\AppData\Local\Microsoft\OneDrive\OneDrive.exe, C:\Windows\sysWOW64\OneDrive.exe
Security Rules		
Line 2	Effect	Audit, Permit, ApplyKey
Line 2	Action	all_ops
Key Selection Rules		key1

See the CTE Data Transformation Guide for more information.

Setting up Microsoft DPM with CTE

Microsoft System Center Data Protection Manager (DPM) is a robust enterprise backup and recovery system that contributes to your BCDR (Business Continuity and Disaster Recovery) strategy by facilitating the backup and recovery of enterprise data. The DPM is a server-agent configuration setup. DPM agent is usually pushed over to the managed host by the DPM server. See Data Protection Manager for more information.

Prerequisite

If you are using LDT, create two new LDT policies. If you are not using LDT, create two new standard policies.

LDT No-View policy

1. In CipherTrust Manager, create a new LDT policy.
2. In that policy, create a process set called **DPM process** which contains the following directory and file:

Directory	File
C:\Program Files\Microsoft Data Protection Manager\DPM\bin	DPMRA.exe

3. Create a security rule for a noview key that contains the following criteria:

Order	Field	Value
1	Action	key_op
	Effect	permit, applykey
2	Process Set	DPM process
	Action	all_ops
	Effect	permit, audit
3	Action	all_ops
	Effect	permit, applykey, audit
4	Action	all_ops
	Effect	deny, audit

Note

Once you have the policy with the Process Set (No-View) rule applied, you can guard any directory, disk, or even a bare metal system backup and recovery, and the encrypted data is backed up.

LDT Open Policy

1. In CipherTrust Manager, create a new LDT policy.

2. Create a security rule for an open key, (where the backup job backs up decrypted data to clear format), that contains the following criteria:

Order	Field	Value
1	Action	key_op
	Effect	permit, applykey
2	Action	all_ops
	Effect	permit, applykey, audit
3	Action	all_ops
	Effect	deny, audit

Note

If you use an open security rule, the backed up data is not encrypted. You can restore the backup job to a baseline directory. It is clear and readable.

Standard No-View Policy

1. In CipherTrust Manager, create a new standard policy.

2. In that policy, create a process set called **DPM process** which contains the following directory and file:

Directory	File
C:\Program Files\Microsoft Data Protection Manager\DPM\bin	DPMRA.exe

3. Create a security rule for a noview key, (where the backup job backs up the encrypted data), that contains the following criteria:

Order	Field	Value
1	Process Set	DPM process
	Action	all_ops
	Effect	permit, audit
2	Action	all_ops
	Effect	permit, applykey, audit
3	Action	all_ops
	Effect	deny, audit

Note

Once you have the policy with the Process Set (No-View) rule applied, you can guard any directory, disk, or even a bare metal system backup and recovery, and the encrypted data is backed up.

Standard Open Policy

If you are not using LDT, create a standard policy:

1. In CipherTrust Manager, create a new standard policy.

2. Create a security rule for an open key, (where the backup job backs up decrypted data to clear format), that contains the following criteria:

Order	Field	Value
1	Action	all_ops
	Effect	permit, applykey, audit
1	Action	all_ops
	Effect	deny, audit

Note

If you use an open security rule, the backed up data is not encrypted. You can restore the backup job to a baseline directory. It is clear and readable.

Installing and Setting up CipherTrust Transparent Encryption

1. Copy dataset `E:\office2007_bk` folder to `E:\data folder`.

2. Install Windows CTE agent and register the host to your key manager.

3. Using the policy that you created in the Prerequisites section, guard the `to E:\data` folder

Installing and Configuring DPM

- Setup and configure DPM v2022 for your Windows 2019 Server.

Setting up DPM

1. Start **System Center 2022 DPM Administrator Console**.

2. Add a DPM Storage pool volume:

   a. Click **Management > Disk Storage > Add**.

   b. Select an available volume (ex. F:) and click **OK**.

   c. Click **Yes** to allow DPM to format the volume before adding it to storage pool.

3. Create a **Protection Group**.

   a. Click **Protection > New** and click **Next**.

   b. Select **Server** and click **Next**.

   c. Select **All volumes >** `E:\data` and click **Next**.

   d. Enter the **Protection group name** (ex. Protection Group 1) and click **Next**.

   e. Click **Specify Short-term Goals** and click **Next**.

f. Choose **Replica Creation Method (Automatically or manual)** and click **Next**.

g. Select **Run a consistency check if a replica becomes inconsistent** or **Run a daily consistency check according to the following schedule** and click **Next**.

4. Click **Create Group**.

5. Set up **Recovery**:

   a. Click **Recovery > Select Recover** and click **Next**.

   b. In Review Recovery Selection, select **Recover to the original location** or **Recover to an alternate location** and then enter the alternate location in the field and click **Next**.

   c. Select recovery type:

      - Create Copy
      - Skip
      - Overwrite

   d. Click **Recover** to start.

Validating Setup

1. Run **Windiff** to compare the original dataset `E:\office2007_bk` folder and recovery GuardPoint `E:\data` folder. Make sure that there is no data corruption.

2. View a text file from `E:\restore`. Make sure that you see cipher text.

3. Using the policy that you created in the Prerequisites section, guard the `E:\restore` folder.

4. View a text file from GuardPoint `E:\restore` folder and make sure that you see clear text.

5. Run **Windiff** again to compare GuardPoint `E:\data` folder and the recovery GuardPoint `E:\restore` folder. Make sure that there is no data corruption.

6. Update GuardPoint `E:\data` folder by adding, deleting or modifying a text file.

7. In **DPM**, click **Protection**, then right-click on `E:\data` and select **Create recovery point**.

8. Choose **Create a recovery point after synchronizing** and click **OK**.

9. Click **Recovery**, then right-click on `E:\data` and select **Show All Recovery Points**.

10. Select latest protection and click **Recovery > Select Recover** and click **Next**.

11. In Review Recovery Selection, select **Recover to the original location** or **Recover to an alternate location** and then enter the alternate location in the field. Click **Next**.

12. Select recovery type:

    - Create Copy
    - Skip
    - Overwrite

13. Click **Recover** to start.

14. Using the policy that you created in the Prerequisites section, guard the `E:\restore1` folder.

15. Run **Windiff** to compare the GuardPoint `E:\data` folder and recovery GuardPoint `E:\restore1\data` folder. Make sure that there is no data corruption.

Support Contacts

If you encounter a problem while installing, registering, or operating the product, please refer to the documentation before contacting support. If you cannot resolve the issue, contact your supplier or Thales Customer Support.

Thales Customer Support operates 24 hours a day, 7 days a week. Your level of access to this service is governed by the support plan arrangements made between Thales and your organization. Please consult this support plan for further information about your entitlements, including the hours when telephone support is available to you.

Customer Support Portal

The Customer Support Portal, at Thales Customer Support, is where you can find solutions for most common problems. The Customer Support Portal is a comprehensive, fully searchable database of support resources, including software and firmware downloads, release notes listing known problems and workarounds, a knowledge base, FAQs, product documentation, technical notes, and more. You can also use the portal to create and manage support cases.

Tip

You require an account to access the Customer Support Portal. To create a new account, go to the portal and click on the REGISTER link.

Telephone Support

If you have an urgent problem, or cannot access the Customer Support Portal, you can contact Thales Customer Support by telephone at +1 410-931-7520. Additional local telephone support numbers are listed on the support portal.

Email Support

You can also contact technical support by email at technical.support@Thales.com.