THALES

CipherTrust Transparent Encryption

CTE-Live Data Transformation with Data Security Manager

Release 7.1.1

Document Version 3

October 10, 2022

CTE-Live Data Transformation with Data Security Manager
October 10, 2022

Contents

Preface

CTE-Live Data Transformation with Data Security Manager describes how to configure and use CipherTrust Transparent Encryption - Live Data Transformation (CTE-LDT) when the host is registered to a Vormetric Data Security Manager (DSM). If you want to use CipherTrust Manager as your key manager, see the *CTE-Live Data Transformation with CipherTrust Manager* guide.

CTE-LDT is an optional, separately licensed feature of CipherTrust Transparent Encryption (CTE). With CTE-LDT, a Administrator can change the encryption key and re-encrypt GuardPoint data without suspending user or application access to the data.

Intended Audience

CTE-Live Data Transformation with Data Security Manager is for security teams who want to rekey the existing GuardPoint data, or who need to perform an initial encryption of their GuardPoint data.

Assumptions

CipherTrust Transparent Encryption - Live Data Transformation is an enhancement to existing functionality. This documentation assumes the reader is familiar with the following CipherTrust products and processes:

- Vormetric Data Security Manager (DSM)
- CipherTrust Transparent Encryption (CTE)
- Key management
- Data encryption

The CTE Agent Documentation Set

The following guides are available for CTE Agent:

- *CTE Agent for Linux Quick Start Guide*
- *CTE Agent for Linux Advanced Configuration and Integration Guide*
- *CTE Agent for Windows Quick Start Guide*
- *CTE Agent for Windows Advanced Configuration and Integration Guide*
- *CTE Agent for AIX Installation and Configuration Guide*
- *CTE Data Transformation Guide*
- *CTE-Live Data Transformation with Data Security Manager*
- *CTE-Live Data Transformation with CipherTrust Manager*
- *Compatibility Matrix for CTE Agent with CipherTrust Manager*
- *Compatibility Matrix for CTE Agent with Data Security Manager*
- *Compatibility Matrix for CTE Agent for AIX with CipherTrust Manager*
- *Compatibility Matrix for CTE Agent for AIX with Data Security Manager*

- *Release Notes for CTE for Linux Version 7.1.1.66*
- *Release Notes for CTE for Windows Version 7.1.1.66*
- *Release Notes for CTE for AIX Version 7.1.1.26*

To access any of these guides for the latest releases of CTE Agent, go to https://thalesdocs.com/ctp/cte/index.html.

Document Conventions

The document conventions describe common typographical conventions and important notice and warning formats used in Thales technical publications.

Typographical Conventions

This section lists the common typographical conventions for Thales technical publications.

Table 3-1: Typographical Conventions

Convention	Usage	Example
bold regular font	GUI labels and options	Click the **System** tab and select **General Preferences**.
bold italic monospaced font	Variables or text to be replaced	https://<***Token Server name***>/admin/ Enter password: <***Password***>
`regular monospacedfont`	• Commands and code examples • XML examples	`session start iptarget=192.168.253.102`
italic regular font	GUI dialog box titles	The *General Preferences* window opens.
	File names, paths, and directories	*/usr/bin/*
	Emphasis	*Do not* resize the page.
	New terminology	*Key Management Interoperability Protocol (KMIP)*
	Document titles	See *CTE-Live Data Transformation with Data Security Manager* for information about CipherTrust Transparent Encryption.
quotes	• File extensions • Attribute values • Terms used in special senses	".js", ".ext" "true" "false", "0" "1+1" hot standby failover

Notes, Tips, Cautions, and Warnings

Notes, tips, cautions, and warning statements may be used in this document.

A Note provides guidance or a recommendation, emphasizes important information, or provides a reference to related information. For example:

> **Note**
> It is recommended to keep tokenization keys separate from the other encryption/decryption keys.

A tip is used to highlight information that helps you complete a task more efficiently, such as a best practice or an alternate method of performing the task.

> **Tip**
> You can also use Ctrl+C to copy and Ctrl+P to paste.

Caution statements are used to alert you to important information that may help prevent unexpected results or data loss. For example:

> **CAUTION**
> **Make a note of this passphrase. If you lose it, the card will be unusable.**

A warning statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data. For example:

> **WARNING**
> **Do not delete keys without first backing them up. All data that has been encrypted with deleted keys cannot be restored or accessed once the keys are gone.**

Sales and Support

If you encounter a problem while installing, registering, or operating this 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.

For support and troubleshooting issues:

- https://supportportal.thalesgroup.com
- (800) 545-6608

For Thales Sales:

- https://cpl.thalesgroup.com/encryption/contact-us
- CPL_Sales_AMS_TG@thalesgroup.com
- (888) 267-3732

Chapter 1: Introduction to CTE-LDT

This chapter contains the following topics:

Overview of CTE-LDT

CipherTrust Transparent Encryption - Live Data Transformation (CTE-LDT) is an optional, separately licensed feature of CipherTrust Transparent Encryption (CTE). With CTE-LDT, after enabling a GuardPoint, a Administrator can encrypt, or rekey, GuardPoint data without blocking user or application access to the data. In CTE-LDT, *rekey* means decrypting data with the current cryptographic key and then encrypting it with a new cryptographic key. The concept of rekey, and how CTE-LDT rekeys data, is described in this document.

After enabling GuardPoints, CTE-LDT performs *initial* encryption or rekeying in the background, unnoticed by users. The data stays live and available. This accelerates CTE deployments and eliminates the need to block application and user access to data during encryption or rekey operations, which can seriously inconvenience users and affect operational efficiency.

With CTE-LDT, the Administrator can create a single CTE-LDT policy for both initial encryption and subsequent rekeying. The same policy applies to production access and security rules without restricting user or application access to data. Applications have continuity of access to GuardPoint data during CTE-LDT.

WARNING

To prevent data loss or corruption, you must stop all applications and users that are accessing files inside a GuardPoint before enabling a Live Data Transformation encryption policy for that GuardPoint. Terminating the applications closes all files that are currently being accessed inside the GuardPoint.

Unlike non-Live Data Transformation policies, however, you do *not* need to keep the GuardPoint offline while data transformation takes place. Instead, you can restart all applications as soon as the GuardPoint has been applied to the host, and CTE will perform the data encryption in the background. This is the only application service downtime required when using CTE-LDT.

Use Cases

This section provides a summary of typical uses for CTE-LDT. The concepts mentioned in this section are described in more detail throughout the rest of this guide.

1. Encrypt unprotected data.

   When protecting files in a directory, you must encrypt them. This process is called *initial data encryption*.

2. Convert non-CTE-LDT GuardPoints to CTE-LDT GuardPoints

   Use when you have existing GuardPoints that are protected with policies created before you started using CTE-LDT.

3. Rekey process.

   Changing the key from one version to another version of the same key provides more security. Using CTE-LDT, you can change the encryption keys to more secure keys.

4. Transform the encrypted data to clear data.

Keys in CTE-LDT (Versioned Keys)

CTE-LDT uses *versioned* keys. Each version of a particular versioned key has the same key name and encryption algorithm, but its own unique cryptographic material. That means that data encrypted with version 3 of a key named `LDT-Key` *cannot* be decrypted by *any* other version of `LDT-Key`, even though the key name remains the same.

Versioning allows you to add one key to your CTE-LDT encryption policy and then use different versions of that same key to periodically re-encrypt your data over time. CTE-LDT uses the new key material to transform data to the new key version, as part of the same Live Data Transformation policy that also protects the data. The process of re-encrypting data with a new version of the existing key is called *Key Rotation*.

When you create a versioned key in the DSM, you set its *life span*. The life span is the duration (in days) of the versioned key. You also define an *initial expiration date*. When the key reaches its expiration date, it automatically rotates to a new version. After that it rotates every time its life span ends. For example, if you set an initial expiration date of October 10, 2021 and a life span of 10 days, the data will be autoamatically rekeyed first on October 10th and then on October 20th, October 30th, and so on.

Versioned keys all have the same properties as non-versioned keys, plus three additional required properties:

- **Automatic Key Rotation**: Always select this option to make the key automatically rotate. Selecting this option tells the DSM that the key is a versioned key.
- **Key Version Life Span (days)**: Frequency of key rotation in days. Applies only if Automatic Key Rotation is selected.
- **Expiration Date**: CTE-LDT uses the initial version of the key until this date occurs. On the expiration date, CTE-LDT creates a new key version. Subsequently, it creates new versions based on the Key Version Life Span.

> **WARNING**
> **You *must* specify the Expiration Date. Without an Expiration Date, a versioned key never rotates from the initial key version, and consequently, never triggers CTE-LDT.**

Rekey | Key Rotation

In CTE-LDT, *rekeying* or *key rotation* means decrypting the data with a previous version of the key and re-encrypting it with a new version of the key. CTE-LDT allows users and applications to access data while CTE-LDT is rekeying the data. Rotating the key and re-encrypting the GuardPoint data with the new version of the key helps to maintain a high level of data security.

Most often, the rekey happens automatically, because each versioned key has a Key Version Life Span that specifies the lifespan of the key. In addition to this automatic key version rotation, you can manually generate a new version of the current key if a new version is required.

For more information, see:

- "Creating and Viewing Versioned Keys" on page 33
- "Rotating Encryption Keys (Rekey)" on page 51

CTE-LDT Policies

In CTE-LDT, you define a single policy for initial data encryption and subsequent rekeying. The policy specifies:

- **Current key**

  Associated with data that you want to protect using CTE-LDT. This is either a non-versioned key from an earlier policy, or `clear_key`, which means that the data is not currently encrypted.

- **Transformation key**

  The versioned key that CTE-LDT applies to transform the data from the key used for initial data transformation. When the transformation key rotates, it transforms the data from a previous version of the transformation key to a new version.

  > **Note**
  > Transformation key and versioned key are used interchangeably throughout this document.

As soon as CTE-LDT applies the policy to a GuardPoint and enables protection for it, CTE-LDT triggers an initial transformation from the current key to the transformation key.

> **Note**
> With DSM 6.1 or later, you may select current or transformation keys for CBC or CBC-CS1 encryption modes.

When the transformation key expires, it generates the next version of the versioned key with new cryptographic material. The DSM or CipherTrust Manager then pushes the policy to the hosts. The policy now contains the new version of the key. This initiates a rekey process on the GuardPoint to transform data to the new version of the transformation key specified in the policy.

Users and applications can continue accessing data without any interruption during initial encryption and subsequent key transformations.

> **Note**
> During CTE-LDT policy creation, you must use the Apply Key effect in your policy. If you do not, then end users can see the clear text data until the file is transformed.

CTE-LDT Runtime Flow

This section presents an overview of how CTE-LDT works and what to expect when CTE-LDT is enabled and running in your environment. All of the tasks mentioned here are described in more detail later in this chapter.

First, the administrator completes CTE-LDT setup:

1. Upload the CTE-LDT license on the DSM or CipherTrust Manager.
2. Register CTE-LDT hosts with the DSM or CipherTrust Manager and be sure that the hosts are licensed for CTE-LDT.
3. Create one or more versioned keys.

   For each key, specify **Automatic Key Rotation**, the **Expiration Date**, and the **Key Version Life Span**.
4. Define Live Data Transformation policies which use the versioned key(s) and contain rules governing CTE-LDT operations.

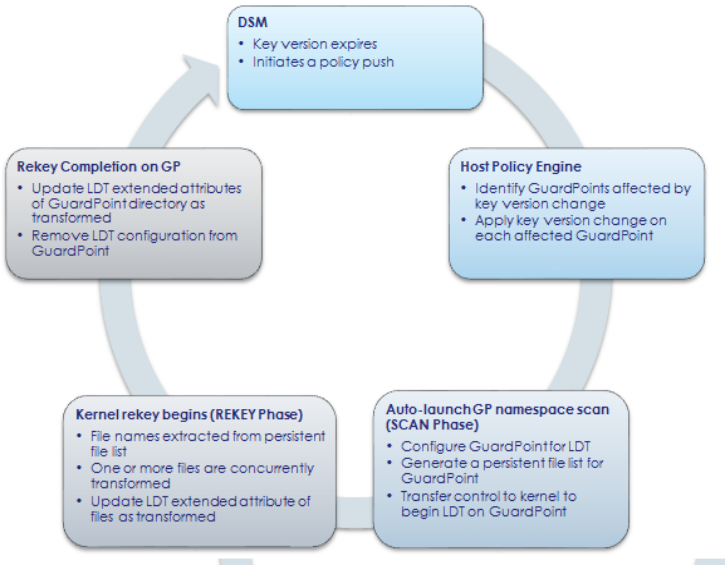
5. Optionally provide Quality of Service (QoS) settings for the CTE-LDT hosts. The QoS settings control the:
   - Window of time in which CTE-LDT operations are allowed to run.
   - Percentage of CPU resources that CTE-LDT can use, or the amount of data to transform according to the QoS setting per the Administrator .

   Note: Configuring the QoS settings is highly recommended as a best practice.

When these items are set up, CTE-LDT is ready to transform and encrypt data by applying policies to GuardPoints for live initial transformation and subsequent rekeys, as well as enforcement of security rules.

The following figure shows the sequence of events during ongoing usage of CTE-LDT.

Figure 1-1: Phases of CTE-LDT runtime operation after keys and policies are defined



1. **Initial data transformation starts or key expires**

   CTE-LDT begins when an Live Data Transformation policy is first applied to a GuardPoint or when a current key version expires. The DSM pushes the new policy, or the notification of a key version change, to the hosts that are protected by the policy. (If the same versioned key is used in multiple policies, *all* of the hosts associated with the policies that contain the key are notified when the key changes.)

2. **New key version triggers a rekey on the affected GuardPoints**

   On each host/client , CTE determines which GuardPoints are using the key that has just rotated to a new version. CTE starts an CTE-LDT rekey on each of those GuardPoints.

   On Windows, you must wait for the current key rotation process to finish before you can launch another rekey request. On Linux, if another rekey is already underway on that GuardPoint, the new rekey is queued for later execution. For details, see "Rotating Encryption Keys While a Rekey is in Progress (Relaunch)" on page 55.

3. **Scan for files**

   On each GuardPoint where CTE has started a rekey, CTE-LDT determines which files to transform. CTE-LDT takes inventory of files encrypted with earlier versions of the rotated key and makes a persistent list of the files for transformation. During this phase, the Rekey Status in the GuardPoint Status window of the DSM Management Console displays Starting, then Scanning.

   The scan phase might be interrupted, such as by a host reboot. In this case, when the host reboots and the GuardPoint is enabled again, the scan operation starts over from the beginning.

4. **Rekey/Key Rotation**
   a. Each file, from the persistent list of files, is decrypted using the old version of the key. The old key is applied to each file and then re-encrypted using the new version of the key. Note that new files created during the CTE-LDT process do not need to be rekeyed, as they inherit the new version of the key. Multiple files and multiple regions of files are rekeyed simultaneously.
   b. The CTE-LDT extended attribute of each file is updated. (For more about extended attributes, see "CTE-LDT Metadata in Extended Attributes" on page 72.)
   c. You can suspend and resume the CTE-LDT rekey operation manually, or through the QoS schedule. This manages the impact CTE-LDT has on other applications and processes.

   During this phase, the Rekey Status in the GuardPoint Status window of the DSM Management Console shows Rekeying or Suspended.

   If system errors occur during rekeying, such as IO errors or crashes, CTE-LDT can manage and recover from them after the system error is fixed.

5. **Finish**

   When all of the required files in the GuardPoint have been rekeyed, the system and storage resources used by CTE-LDT are released, except for the storage required for the extended attributes.

   CTE-LDT creates a rekey report, listing all of the files that were rekeyed. For more information, see "Obtaining a Rekey Report" on page 104.

   Upon completion of rekey, the Rekey Status in the GuardPoint Status window of the DSM Management Console shows Rekeyed.

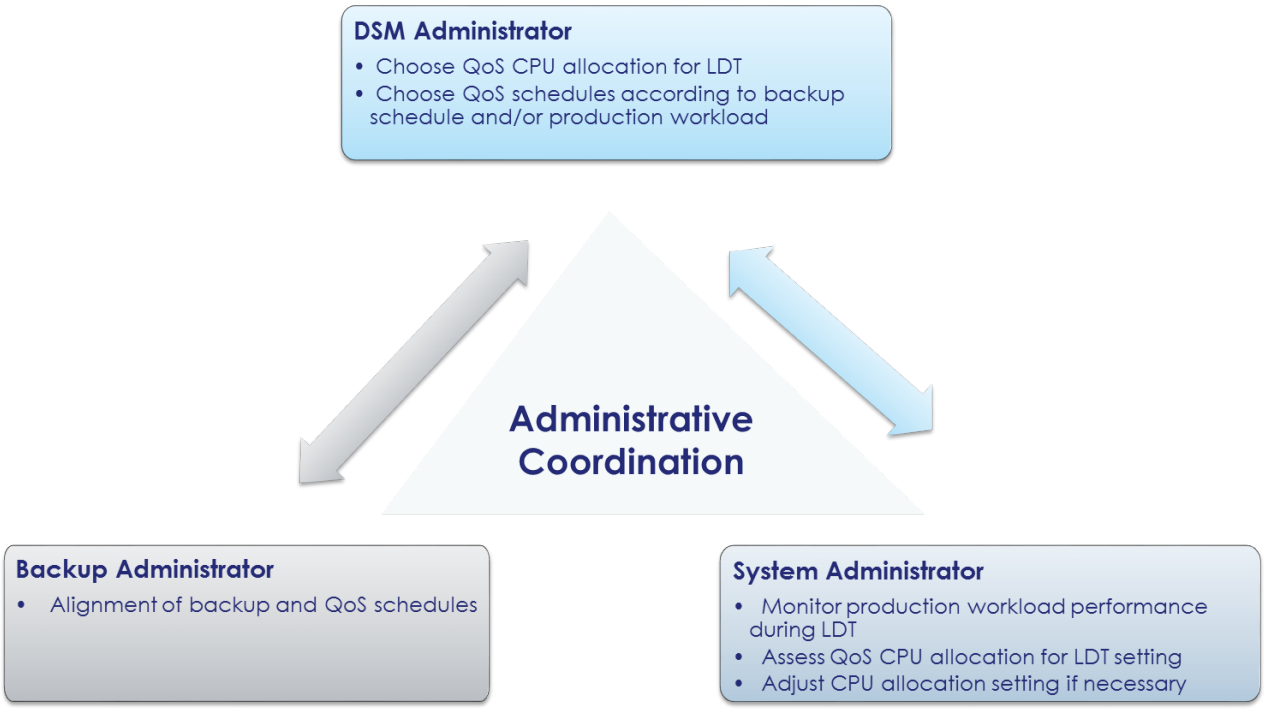
CTE-LDT Administrator Roles

Using CTE-LDT requires coordination and collaboration between several different administrators. Security, system, application performance and backup schedules are critical factors that affect planning. It is very important for the **DSM Security Administrator** to coordinate with the **system administrator** and the **backup administrators**. The following table describes the roles and responsibilities for these administrators.

CTE-LDT Administrators

Role	Responsibility	Actions
DSM Security Administrator	Administers Vormetric Data Security Manager (DSM)	Coordinates with system and backup administrators. Creates security policies and rules for CTE-LDT. Defines Quality of Service settings for CTE-LDT, applications, and backups.
System administrator	Administers servers on which CipherTrust Transparent Encryption (CTE) is deployed	Coordinates with the DSM Security Administrator to create Quality of Service schedule, taking into consideration the backup schedules.
Backup administrator	Manages data backups for data encrypted by CTE	Coordinates with the DSM Security Administrator to create Quality of Service schedule, taking into consideration the backup schedules.

The following process is recommended for coordination between the various administrators in order to achieve optimal QoS schedule settings and allocation of CPU resources when using CTE-LDT.



Resiliency

CipherTrust Transparent Encryption - Live Data Transformation is resilient to many user actions and system occurrences. Since it is designed to run periodically without intervention, it includes various features to provide this resilience.

Before CTE-LDT enables a GuardPoint, it checks for any inconsistencies in files that were undergoing rekey at the time when system operations were interrupted. If it finds any inconsistencies, CTE-LDT corrects them before it enables the GuardPoint. It should only take a few seconds to identify and correct any inconsistency. On Windows, this process is delayed until applications access the affected files.

During rekey, if an issue such as an I/O problem or system crash occurs, the resiliency features of CTE-LDT ensures the consistency of user data. Such issues can cause an interruption in the middle of a rekey operation. When system operations resume, CTE-LDT corrects the problems and then resumes rekeying.

Chapter 2: Getting Started

This chapter contains the following topics:

Using CTE-LDT

> **Notes**
> - If you are new to CTE and CTE-LDT, read the *DSM Administration Guide* first to familiarize yourself with the concepts of GuardPoints and Policies.
> - Before installing CTE-LDT, see the **Linux Kernel Support** to verify that your version of Linux is supported.

The following list contains the steps for successfully setting up and using CTE-LDT.

1. Install the CTE-LDT License on the DSM or CipherTrust Manager.

   CipherTrust Transparent Encryption - Live Data Transformation is a separately-licensed feature of CTE. Before you can use it, you have to install the license to activate it. CTE-LDT is licensed for a specific number of hosts.

   For details about licensing, see the *DSM Administration Guide*.

2. Install the CTE Agent and select the CTE-LDT feature during the install. For more information, see "Installing and Registering the CTE Agent Software on Linux" on page 24 or "Installing and Registering the CTE Agent Software on Windows" on page 27.

   > **Notes**
   > - If the host is already registered you can enable CTE-LDT through the DSM Management Console. For details, see "Enabling CTE-LDT on a Protected Host" on page 31.
   > - If you want to upgrade your CTE Agent to protect NFS/CIFS shares, see "Upgrading an Existing CTE Agent to Support NFS/CIFS Shares" on page 30.

3. Create Versioned Keys.

   CTE-LDT uses *versioned keys*. A versioned key rotates to the next version of the key generating new key material automatically without policy change. CTE-LDT encrypts data with keys that use encryption standards like AES-128 and AES-256. This allows data to be re-encrypted without users having to edit the policy.

   For more information, see "Keys in CTE-LDT (Versioned Keys)" on page 13 and "Creating and Viewing Versioned Keys" on page 33.

4. Create CTE-LDT Policies.

   CTE-LDT uses a single Live Data Transformation policy to address both initial encryption and subsequent rekeying. The same policy applies to production access and security rules without restricting user or application access to data. Applications have continuity of access to GuardPoint data during CTE-LDT.

   For more information, see "Creating CTE-LDT Policies" on page 35.

5. Set 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 CipherTrust Transparent Encryption - Live Data Transformation. Administrators can also choose schedules for data transformation, or manually pause or resume transformation operations.

   For more information, see "Quality of Service" on page 37.

6. Create GuardPoints and apply CTE-LDT policies to the GuardPoints.

   A GuardPoint is a directory in the file system hierarchy, where its contents have a CipherTrust data protection policy applied to it. The CTE Agent intercepts any attempt to access anything in the GuardPoint and uses the policies obtained from the DSM or CipherTrust Manager to grant or deny the access attempt. Typically data copied into a GuardPoint is encrypted, and only authorized users can decrypt and use that GuardPoint data.

   For more information, see "Creating a CTE-LDT GuardPoint" on page 48.

Backup/Restore

Before you enable a GuardPoint with an CTE-LDT Policy, make sure that you back up your data.

For more information, see "Backup Requirements" on page 22 and "Backing Up and Restoring CTE-LDT GuardPoints" on page 82.

Restrictions

Remember the following restrictions when using CTE-LDT:

- CTE-LDT does not support nested GuardPoints, where a GuardPoint is contained inside another GuardPoint.
- For HA clusters, CTE-LDT only supports the Asymmetric (active/passive) configuration. CTE-LDT does not support Symmetric (active/active) configuration.
- If you want to create CTE-LDT GuardPoints on Linux NFS shares or Windows CIFS shares, you must register the host with a Vormetric Data Security Manager (DSM) key manager. CipherTrust Manager does not currently support CTE-LDT GuardPoints on NFS/CIFS shares. This functionality will be added in a future CipherTrust Manager release.

  > **Note:** The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

Windows Only Limitations

CTE-LDT supports GuardPoints on CIFS network shared directories with the following restrictions:

- Only single host CIFS shares are currently supported. You cannot use CTE-LDT in a multi-host CIFS environment. Support for multiple CIFS hosts will be added in a future release.
- You cannot guard both CIFS shares with CTE-LDT and local directories on the same host (even if the local directories use a Standard CTE policy). Instead, you must choose one or the other when you install the CTE Agent. In addition, if you want to upgrade an existing CTE Agent to support CTE-LDT CIFS share GuardPoints, you must first remove any existing GuardPoints, uninstall the CTE Agent, and then install the latest CTE Agent in its place.

  For details, see "Installing and Registering the CTE Agent Software on Windows" on page 27 and "Upgrading an Existing CTE Agent to Support NFS/CIFS Shares" on page 30.

- Only unstructured data can be encrypted.
- If any files are opened exclusively by another application, CTE-LDT cannot rekey those files until the other applications have released the lock.
- CTE-LDT on a ReFS file system runs slowly because of limited support from the Extended Attributes on the ReFS file system.

> **Note:** Customers running older versions of ReFS.sys on Windows Server 2012 R2 should be aware of the memory growth issue encountered by the Thales engineering team. This issue seems to occur only when CTE-LDT is running on a large number of files. As the system memory consumption by REFS file system increases, it can eventually make the system unresponsive. This issue does not occur with the recent versions of ReFS file system drivers available on Windows Server 2016. After consulting with Microsoft, they suggest that all customers migrate to Windows Server 2016 if they are using ReFS file.

Linux Only Limitations

- CTE-LDT does not support Linux automounted file systems.
- CTE-LDT support is limited to `ext3`, `ext4`, and `XFS` file systems when `user_xattr` mount option is enabled.
- CTE-LDT does not support system hibernation (`pm-hibernate`) on Linux hosts where CTE-LDT is in use.
- CTE-LDT *does not* support GuardPoints on raw devices, but it *does* support GuardPoints on NFS shares starting with CTE release 7.1.0.
- For CTE-LDT on NFS shares, one and only one host can access the GuardPoint during data transformation operations. For details, see "GuardPoint Management Over an NFS Share" on page 80.
- If you are protecting the same GuardPoint directory in the same NFS share on mulitple hosts, the directory path for that GuardPoint can be different on each CTE host. However, you *must* use the same Live Data Transformation policy to guard the directory on all hosts.
- If any files in an NFS share GuardPoint are opened exclusively by another application, CTE-LDT cannot rekey those files until the other applications have released the lock.
- You cannot use CTE-LDT and Docker container on the same host.
- You cannot use CTE-LDT and OpenShift container on the same host.

Chapter 3: Setting Up CTE-LDT

This chapter contains the following topics:

System Requirements

CipherTrust Transparent Encryption - Live Data Transformation (CTE-LDT) requires the following environment.

> **Note**
> See the latest *Compatibility Matrix for CTE Agent with Data Security Manager*, in the online support portal, for a list of CTE versions and supported operating systems.

CipherTrust Software Requirements

Platform	CTE Agent Version	DSM Version
Linux (local FS)	To guard data on a local Linux file system, you must have a VTE Agent version 6.0.0 or higher *or* a CTE Agent version 7.0 or higher installed on each protected host.	DSM version 6.0 or higher.
Linux (NFS)	To guard data on a Linux NFS share, you must have a CTE Agent version 7.1.0 or higher installed on each protected host. For more information, see "Installing and Registering the CTE Agent Software on Linux" on page 24 and "Upgrading an Existing CTE Agent to Support NFS/CIFS Shares" on page 30.	DSM version 6.4.4 or higher.
Windows (local FS)	To guard data on a local Windows file system, you must have a VTE Agent version 6.0.0 or higher *or* a CTE Agent version 7.0 or higher installed on each protected host.	DSM version 6.1.x or higher.
Windows (CIFS)	To guard data on a Windows CIFS share, you must have a CTE Agent version 7.1.0 or higher installed on each protected host and you must have enabled guarding on network shares. For more information, see "Installing and Registering the CTE Agent Software on Windows" on page 27 and "Upgrading an Existing CTE Agent to Support NFS/CIFS Shares" on page 30.	DSM version 6.4.4 or higher.

CTE-LDT License Requirements

One CTE-LDT license is required for *each* protected host.

Host System Requirements

Memory requirements: Both Linux and Windows require a minimum of 8 GB memory on each protected host.

Disk space requirements: CTE-LDT requires a specific amount of disk space in the file system or NFS/CIFS share for each GuardPoint, over and above the space required for the guarded files themselves. CTE-LDT uses the additional space to store CTE-LDT metadata.

- For Linux, to estimate the amount of free file system disk space required by CTE-LDT on a Linux host, use the `voradmin ldt space` command.
- For Windows, the typical minimum space requirement for a GuardPoint is the number of files in the GuardPoint multiplied by 4K, plus 256MB. To estimate the amount of free space required for a GuardPoint, use the `voradmin ldt space <GuardPoint-name>` command.

Backup Requirements

For Windows GuardPoints in local file systems or CIFS shares, the application can use Volume Shadow Copy service (VSS) or roboCopy for backup and restore.

For Linux GuardPoints in local file systems, the backup application must have the capability to back up user-extended attributes. For example, Thales tested CTE-LDT with NetBackup. You can also use other applications that can back up user-extended attributes.

For Linux GuardPoints on NFS shares, the normal procedure is to back up the NAS server that serves the NFS clients rather than backing up the individual NFS clients. Because backups on NAS servers are remote from a CTE perspective, it is extremely important that CTE is *not* performing any data transformation in any GuardPoint while the NAS server is being backed up. This includes both initial data transformation and automatic or manual data rekey.

WARNING

Make sure that all live data transformation has completed on all NFS GuardPoints before you back up the NAS servers associated with the NFS shares on which those GuardPoints reside. You must also make sure that no automatic rekey tasks will start while the NAS backup is in progress.

Linux-Specific Requirements

To use CTE-LDT on an ext3 or ext4 file system, the block size **must** be 4K. Run `dumpe2fs` to determine the block size of the ext3 or ext4 file systems before using CTE-LDT. This limitation does not apply to XFS file systems or NFS shares. You can use CTE-LDT on XFS with a block size of 1K or 2K.

Local File System Requirements

For CTE-LDT to work properly on local file systems, the underlying file system must support and enable user-extended attributes. All of the file systems supported by CTE-LDT support these attributes. If you are using CTE-LDT with ext3 or ext4 mount points, you must explicitly enable the extended attribute mount option by editing `/etc/fstab` and adding the `user_xattr` mount option. In the other file systems supported by CTE-LDT, user extended attributes are enabled by default, so you do not have to explicitly enable them.

Example `/etc/fstab` entry for ext3 on Red Hat 6 or SLES:

```
/dev/sdb1 /disk2 ext3 defaults,user_xattr 0 0
```

Example `/etc/fstab` entry for ext4 on Red Hat 6 or SLES 12 (ext4 is not supported on SLES 11):

```
/dev/sdc1 /disk3 ext4 auto,users,user_xattr,exec 0 0
```

For more information about extended attributes, see "CTE-LDT Metadata in Extended Attributes" on page 72.

NFS Share Requirements

For GuardPoints on NFS shares, CTE-LDT embeds file specific LDT metadata in the beginning of each file during the initial data transformation phase. The size of each file becomes larger by 4096 bytes to accommodate this required metadata. Both the metadata and the file size increase are hidden from users and applications as long as the GuardPoint remains enabled.

If a GuardPoint is disabled, the metadata remains in each file and both the metadata and the file size increase become visible to users and applications.

> **Note**
> Thales highly recommends that NFS shared directory be mounted with `sync` option.

Supported Applications in Linux

For all of the supported operating systems for database applications, see the *Compatibility Matrix for CTE Agent with Data Security Manager*.

Replication

- `rsync`.
- Hardware/software based replication system.

SAP HANA Fibre Channel Systems

SAP HANA is compatible with CTE-LDT. See the *Compatibility Matrix for CTE Agent with Data Security Manager* and the *CTE Agent for Linux Advanced Configuration and Integration Guide*.

Windows-Specific Requirements

CIFS Share Requirements

For GuardPoints on CIFS shares, CTE-LDT embeds file specific LDT metadata in beginning of each file during the initial data transformation phase. This metadata remains hidden to users and other applications as long as the GuardPoint is enabled.

If a GuardPoint is disabled, the metadata remains in each file and becomes visible to users and other applications.

The size of each file becomes larger by 4096 bytes to accomodate the required CTE-LDT metadata.

Installing the CTE-LDT License

CTE-LDT installs with the CTE Agent software, but it is a separately-licensed CTE feature. Before you can use it, you have to install the license on the DSM to activate CTE-LDT on the CTE host. CTE-LDT is licensed for a specific number of hosts.

1. Obtain an CTE-LDT license from Thales. You can purchase CTE-LDT along with the DSM software or you can add CTE-LDT later.
2. Install the license on the DSM.

   If you purchase CTE-LDT with the DSM, it includes the CTE-LDT license in the same license file with the DSM license, so there is only a single file to install.

   If you add the CTE-LDT license later, you need to install an additional license file. For details, see the *DSM Installation and Configuration Guide*

To confirm that the CTE-LDT license is in effect:

1. Ask the Administrator to log on to the DSM Management Console and click **System > License**.
2. In the **Agent Type** column, find the row labeled **FS**. The number of host licenses displays in the **LDT License** column of this row.

Installing and Registering the CTE Agent Software on Linux

Install the CTE Agent software with the CTE-LDT feature on each host you want to protect. The following procedure describes how to use the interactive installer to install CTE and register CTE-LDT on a Linux host using the shared secret registration method. For additional registration options, see the *CTE Agent for Linux Advanced Configuration and Integration Guide*.

Prerequisites

Make sure you know the following information from the Administrator:

- The server name of the primary DSM as shown on the DSM Dashboard.
- The shared secret for the domain on the primary DSM with which you want to register the host.
- The name of the domain in the DSM with which you want to register the host.
- Optionally, the name of the host group in which this host should be included.

All of this information is case-sensitive and must exactly match the corresponding information in the DSM.

> **Note**
> If registration appears to freeze, verify that the DSM and CTE can communicate with each other over the network.

Procedure

1. Log on to the host where you will install the CTE Agent as `root`. You cannot install the CTE Agent without `root` access.
2. Copy or mount the installation file to the host system. If necessary, make the file executable with the `chmod` command.

3. Install the CTE Agent. A typical installation uses the following syntax:

   `# ./vee-fs-<release>-<build>-<system>.bin`

   For example:

   `# ./vee-fs-7.1.0-66-rh8-x86_64.bin`

   To install the CTE Agent in a custom directory, use the `-d <custom-dir>` option. For example:

   `# ./vee-fs-7.1.0-66-rh8-x86_64.bin -d /home/my-cte-dir/`

   > **Note:** If possible, Thales recommends that you use the default directory `/opt/vormetric`.

   To view all installer options, use the `-h` parameter. For example:

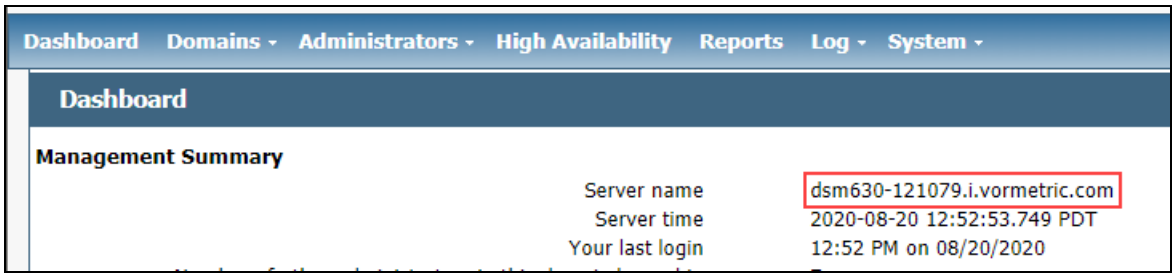
   `# ./vee-fs-7.1.0-66-rh8-x86_64.bin -h`

4. The Thales License Agreement displays. When prompted, type **`Y`** and press Enter to accept.

   The install script installs the CTE Agent software in either `/opt/vormetric` or your custom installation directory and then prompts you about registering the CTE Agent with a key manager.

```
Welcome to the CipherTrust Transparent Encryption File System Agent
Registration Program.

Agent Type: CipherTrust Transparent Encryption File System Agent
Agent Version: 7.1.1.30

In order to register the CipherTrust Transparent Encryption File System Agent
with a Vormetric Data Security Manager
    1) you must know the host name of the machine running the
       DSM (the host name is displayed on the
       Dashboard window of the Management Console), and
    2) unless you intend to use the 'shared secret' registration method,
       the agent's host machine must be pre-configured on the DSM as a
       host with the 'Reg. Allowed' checkbox enabled for this agent type
       on the Hosts window of the Management Console.

In order to register with a CipherTrust Manager you need a valid registration
token from the CM.

Do you want to continue with agent registration? (Y/N) [Y]:
```

5. Enter **Y** to continue with the registration process. The install script prompts you to enter the name of the DSM with which you want to register the host. This name must match the name shown in the **Server name** field in the **Management Summary** section on the DSM **Dashboard** page.



For example:

```
Do you want to continue with agent registration? (Y/N) [Y]: Y

Please enter the primary key manager host name: dsm630-121079.i.vormetric.com

You entered the host name dsm630-121079.i.vormetric.com
Is this host name correct? (Y/N) [Y]: Y
```

6. Enter the host name when prompted. If the Shared Secret registration in your DSM is configured to require an existing host entry, his name must match the name used on the **Add Host** page of the DSM Management Console.

```
Please enter the host name of this machine, or select from the following list. If
using the "fingerprint" registration method, the name you provide must precisely
match the name used on the "Add Host" page of the Management Console.

[1] host14.i.example.com
[2] Host-RHEL-14.i.example.com
[3] 10.3.14.90

Enter a number, or type a different host name or IP address in manually:
What is the name of this machine? [1]: 3
You selected "10.3.14.90".
```

7. When prompted for the registration method, enter **s** for shared secret registration and then enter the required information about the domain, optional host group, and optional host description. For example:

```
Would you like to register to the DSM using a
registration shared secret (S) or using fingerprints (F)? (S/F) [S]: S

What is the registration shared secret?
Please enter the domain name for this host: west-coast-domain
Please enter the host group name for this host, if any:
Please enter a description for this host: West Coast Data Center server 5

Shared secret    : ************
Domain name      : west-coast-domain
Host Group       : (none)
Host description : West Coast Data Center server 5
Are the above values correct?  (Y/N) [Y]: Y
```

8. At the hardware association prompt, select whether you want to enable the hardware association feature that prevents a clone of this machine from accessing keys in the DSM. The default is **Y** (enabled):

```
It is possible to associate this installation with the hardware of this
machine.  If selected, the agent will not contact the key manager or use any
cryptographic keys if any of this machine's hardware is changed.  This
can be rectified by running this registration program again.
Do you want to enable this functionality?  (Y/N) [Y]: Y
```

9. At this point, the install script asks you about some of the optional CTE features you may want to enable. Make sure you select Y for LDT.

   For example:

```
Do you want this host to have docker support enabled on the server?  (Y/N) [N]:

Do you want this host to have Efficient Storage support enabled on the server?
(Y/N) [N]:

Do you want this host to have LDT support enabled on the server?  (Y/N) [N]: Y

Do you want to configure this host for Cloud Object Storage?  (Y/N) [N]:
```

   Make sure you enter **Y** for LDT support.

10. At this point the installation script completes the installation and indicates that it successfully registered the host with the DSM.

```
Generating certificate signing request for the kernel component...done.
Signing certificate...done.
Generating EC certificate signing request for the vmd...done.
Signing certificate...done.
Generating EC certificate signing request for the vmd...done.
Signing certificate...done.
Successfully registered the CipherTrust Transparent Encryption File System Agent
with the Vormetric Data Security Manager on dsm630-121079.i.vormetric.com.
```

11. Verify the installation by checking the CTE processes on the host:
    - Run `vmd -v` to check the version of CTE.
    - Run `vmsec status` to display the CTE processes.
    - Look at the log files in `/var/log/vormetric/install.fs.log.<date>`, especially `vorvmd_root.log`.

Installing and Registering the CTE Agent Software on Windows

Install the CTE Agent software with the CTE-LDT feature on each host you want to protect. The following procedure describes how to use the interactive installer to install CTE and register CTE-LDT on a Windows host. For additional registration options, see the *CTE Agent for Windows Advanced Configuration and Integration Guide*.

Prerequisites

Make sure you know the following information from the Administrator:

- The server name of the primary DSM as shown on the DSM Dashboard.
- The shared secret for the domain on the primary DSM with which you want to register the host.
- The name of the domain in the DSM with which you want to register the host.
- Optionally, the name of the host group in which this host should be included.

All of this information is case-sensitive and must exactly match the corresponding information in the DSM.

> **Note**
> If registration appears to freeze, verify that the DSM and CTE can communicate with each other over the network.

Procedure

1. Log on to the host as a Windows user with System Administrator privileges.
2. Copy the CTE installation file onto the Windows system.
3. Double-click the installation file. The InstallShield Wizard for CipherTrust Transparent Encryption opens.
4. Verify the version of CTE you are installing and click **Next**.
5. On the *License Agreement* page, accept the License Agreement and click **Next**.
6. On the *Live Data Transformation for network shares* page:
   - Select **Yes** if you plan to use CTE-LDT with CIFS share GuardPoints on this host. If you select **Yes**, you will *not* be able to guard any local directories on this host, even if those directories use a Standard CTE policy. Only CTE-LDT GuardPoints on CIFS shares will be supported for this host.

     > **Note:** The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

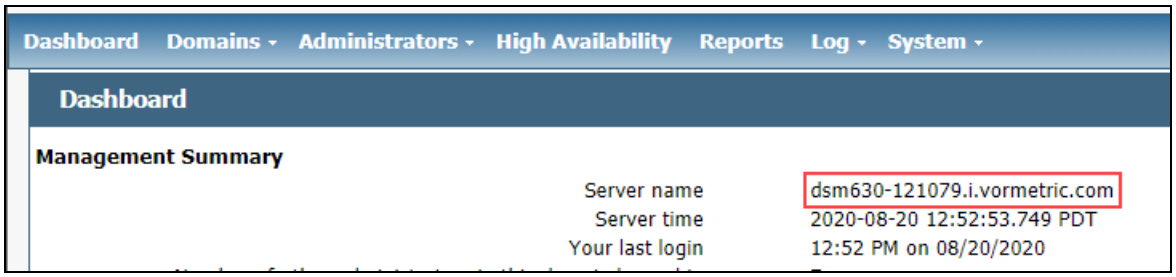
   - Select **No** if you only plan to create local file system GuardPoints on this host. If you select **No**, you will *not* be able to use CTE-LDT with any CIFS share GuardPoints on this host.

   When you are done, click **Next**.
7. On the *Destination Folder* page, click **Next** to accept the default folder or click **Change** to select a different folder. When you are done, click **Next**.

   > **Notes**
   > - Thales recommends that you install CTE in the default installation directory, `C:\Program Files\Vormetric\DataSecurityExpert\agent\`
   > - You must install the CTE Agent on the same drive as Windows. For example, if Windows is installed on the `C:` drive, you must install the CTE Agent on the `C:` drive.

8. On the *Ready to Install* page, click **Install**. When the installation is finished, the **Install Shield Wizard Completed** window opens.
9. On the *InstallShield Wizard Completed* page, make sure the **Register CipherTrust Transparent Encryption now** check box is selected and click **Finish**. The installer opens the Register Host wizard.
10. In the Register Host dialog box, verify the host's machine name and click **Next**.
11. On the *Gathering agent information* page, select the **File System** check box and click **Next**.

12. On the *Gathering Key Manager information* page, enter the FQDN of the Primary DSM. This name must match the name shown in the **Server name** field in the **Management Summary** section on the DSM **Dashboard** page.



When you are done, click **Next**. CTE communicates with the selected DSM to validate what features have been licensed and are available to the CTE Agent.

13. On the *Gathering host name information* page:
    - Specify the host name or IP address of the host. You can select the host name from the drop-down list or type it in the field. If you specify a host name, it must be resolvable by the DNS server.
    - To enable cloning prevention, select the **Enable Hardware Association** check box.
    - If you want to have CipherTrust Transparent Encryption - Live Data Transformation available on the host, select the **Enable LDT Feature** check box. For details on CTE-LDT, see the *CTE-Live Data Transformation with Data Security Manager*.
    - If you want to have the CTE-Efficient Storage feature available on the host, select the **Enable ES Feature** check box. For details about CTE-Efficient Storage, see the *CTE Agent for Windows Advanced Configuration and Integration Guide*.
    - Make sure the **Use Shared Secret Registration** check box is enabled.

    When you are done, click **Next**.

> **Note:** If you get the message “Only CTE-initiated communication is possible”, make sure that the DSM and CTE can communicate over the network.

14. On the *Gathering shared secret registration information* page, enter the following:
    - **Shared secret**:The shared secret established for the domain in the DSM to which you intend to add this host. Contact the Administrator for this value.
    - **Domain name**: The name of the DSM domain to which the host will be added. Contact the Administrator for this information.
    - **Host group** (optional): The name of the host group to which the host will be added. Contact the Administrator for this value.
    - **Host description** (optional): A user-defined description of the host to be registered.

> **WARNING**
> **The shared secret, domain name, and host group are case-sensitive. If any of these are entered incorrectly, an error message displays. If you exceed the number of allowable login attempts to the DSM, you will be locked out of the DSM. For more information, talk to your Administrator.**

    When you are done, click **Register**. CTE contacts the DSM and attempts to register the host with the specified options. The Register Host dialog box displays a message with the results of the registration request.

    If the registration completed successfully, click **Finish**.

15. Restart the host to complete the installation process.
16. After the host has rebooted, you can verify the installation by checking CTE processes:
    a. In the system tray of the protected host, right-click the CipherTrust Lock icon.
    b. Select **Status**. Review the information in the **Status** window to confirm that the correct CTE version is installed and registered.

Upgrading an Existing CTE Agent to Support NFS/CIFS Shares

> **Note**
> The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

Upgrading on Linux

On Linux, you must upgrade your CTE Agents to CTE version 7.1.0 or later and your DSM to version 6.4.4 or later in order to support GuardPoints on NFS shares. You can use the standard upgrade procedure for both the CTE Agents and the DSM.

Make sure that CTE-LDT is enabled through the DSM Management Console for the hosts you want to protect. For details, see "Enabling CTE-LDT on a Protected Host" on the facing page.

> **WARNING**
> **Do *not* upgrade the CTE Agent if any GuardPoints on the host are currently undergoing data transformation, either for initial data transformation or rekey. *All* data transformation in *all* GuardPoints must be completed before you upgrade. To verify this, use the `voradmin ldt list all` command to make sure none of the GuardPoints are in a partially-rekeyed state.**

Upgrading on Windows

On Windows, the requirements for GuardPoints created on CIFS shares using CTE-LDT are different from those created on local directories or for those created on CIFS shares using standard CTE policies. You cannot mix the two on a single host. Instead, each host must contain either CTE-LDT CIFS share GuardPoints or local directory/Standard policy GuardPoints.

If you want to use CTE-LDT on CIFS share GuardPoints on a host on which the CTE Agent is already installed, you must:

1. Remove any existing local directory GuardPoints.
2. Uninstall the current version of CTE Agent from the host.
3. Install CTE Agent version 7.1.0 or higher and respond **Yes** when prompted about guarding network shares. For details, see "Installing and Registering the CTE Agent Software on Windows" on page 27.

   If you do not see this prompt during installation, make sure that you have fully uninstalled the previous version of the CTE Agent.

In addition, you must upgrade your DSM to version 6.4.4 or later. You can use the standard upgrade procedure for the DSM.

Make sure that CTE-LDT is enabled through the DSM Management Console for the hosts you want to protect. For details, see "Enabling CTE-LDT on a Protected Host" below.

Setting the Linux Kernel Time Zone

The Linux kernel contains an internal time structure that may or may not contain time zone information. On system configurations that do not contain time zone information, CTE-LDT stores and displays timestamps for rekey beginning and ending in UTC (Coordinated Universal Time) rather than the system's local time zone. If this occurs, the administrator can set the kernel's internal time zone to the local time zone if they desire timestamps in their local time zone.

To set the Linux Kernel time zone information, at boot time type:

```
# hwclock --systz
```

The command sets the kernel's time zone to the local time zone and resets the System Time based on the current time zone.

> **Note**
> On systems that do not set the time zone by default, existing timestamps for completed rekeys remain in UTC, even if you run `hwclock --systz`. Only timestamps for new rekeys display the local time zone.

Enabling CTE-LDT on a Protected Host

When you install CTE on a host, you can enable CTE-LDT during the registration process as described in "Installing and Registering the CTE Agent Software on Linux" on page 24. If you have already registered the host with the DSM, you can enable CTE-LDT in the host entry in the DSM.

> **Note**
> The CTE-LDT license is valid for a certain number of hosts. Once you reach this limit, you can either purchase additional CTE-LDT licenses, or reclaim a license by removing a CTE-LDT host/client . For details, see "Removing CTE-LDT and Security Encryption" on page 96.
>
> You cannot disable CTE-LDT on a host once it has been enabled.

To enable CTE-LDT on an existing host:

1. In the DSM Management Console, click **Hosts > Hosts**.
2. In the Hosts table, click on the host name of the protected host.
3. On the Edit Host page, select the **Live Data Transformation** check box, then click **OK** or **Apply** to enable CTE-LDT on the protected host.

   > **Note:** After CTE-LDT has been enabled for the host, the check box is greyed out because you cannot change this option once it has been set.

Chapter 4: Using CTE-LDT

Setting up keys, policies, and GuardPoints with CTE-LDT is very similar to performing the same tasks in CTE without CTE-LDT. However, there are some differences.

This chapter contains the steps, in order, for successfully setting up and using CTE-LDT.

Adding Windows CIFS Credentials to the DSM

> **Note**
> The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

If you are using CTE-LDT on a Windows CIFS share, you need to specify the credentials that CTE should use when it accesses data on the share. You can define any number of credentials in the DSM, but you can only assign one credential per GuardPoint. That means the credential you specify must work for all nodes on which the GuardPoint is applied.

> **Tip**
> Thales recommends that you create a new system account for CTE-LDT that can access all data on the CIFS shares.

The DSM lets you select the credentials you want to use for a GuardPoint at GuardPoint creation time. For more information, see "Creating a CTE-LDT GuardPoint" on page 48.

To add credentials to the DSM, do the following:

1. Log into the DSM Management Console and switch to the domain you want to use.
2. In the top menu bar, select **Hosts > CIFS Share Credentials**.
3. Above the credentials table click **Add**.

4. In the CIFS Share Credential dialog, enter the following information:

Field	Description
Username	The username to use when logging in to the CIFS share. This can be a local Windows account name or a UPN account name. For example, `joe`. If the account is a UPN account that requires a domain, you can either specify the domain in this field using the "@domain" notation or you can enter the domain in the **User Domain** field. For example, `joe@mydomain.com`. If you enter a domain in this field, leave the **User Domain** field blank. **Note:** You cannot enter a hostname in this field. If you want to specify user@hostname, you must put the username in this field and the hostname in the **User Domain** field.
Password	The password for that username.
Confirm Password	Confirmation of the password for that username.
User Domain	An optional domain or hostname for the username, if required. For example: `myhostname` You should leave this field blank if: • The account is a local Windows account • You entered the domain information in the **Username** field.
Description	An optional description for the credential. This description is only displayed in the DSM.

For details about managing CIFS Share credentials, see the *DSM Administration Guide*.

Creating and Viewing Versioned Keys

In CTE-LDT, you create a *versioned* key for a Live Data Transformation policy and define the *life span* of the key. The life span is the duration (in days) of the versioned key. You also define an *initial expiration date*. When the key reaches its expiration date, it automatically rotates to a new version.

Although an CTE-LDT policy specifies the same key name, when the key rotates, the rotation of the key automatically starts CTE-LDT on the data in the GuardPoints protected with the CTE-LDT policy. There is no need to change the policy. However, you can manually rotate the key if circumstances require it.

Use the **Add Agent Key** page in the DSM Management Console when creating versioned keys.

CAUTION
Make sure that you are working in the Symmetric tab. CTE-LDT does not support asymmetric keys.

Creating a New Versioned Key for CTE-LDT

1. Log into the DSM Management Console and switch to the domain you want to use.
2. In the top menu bar, click **Keys**.
3. In the Key table, click **Add** and specify the key options you want to use. The following options are required for CTE-LDT keys. Any other key options can be set as desired.

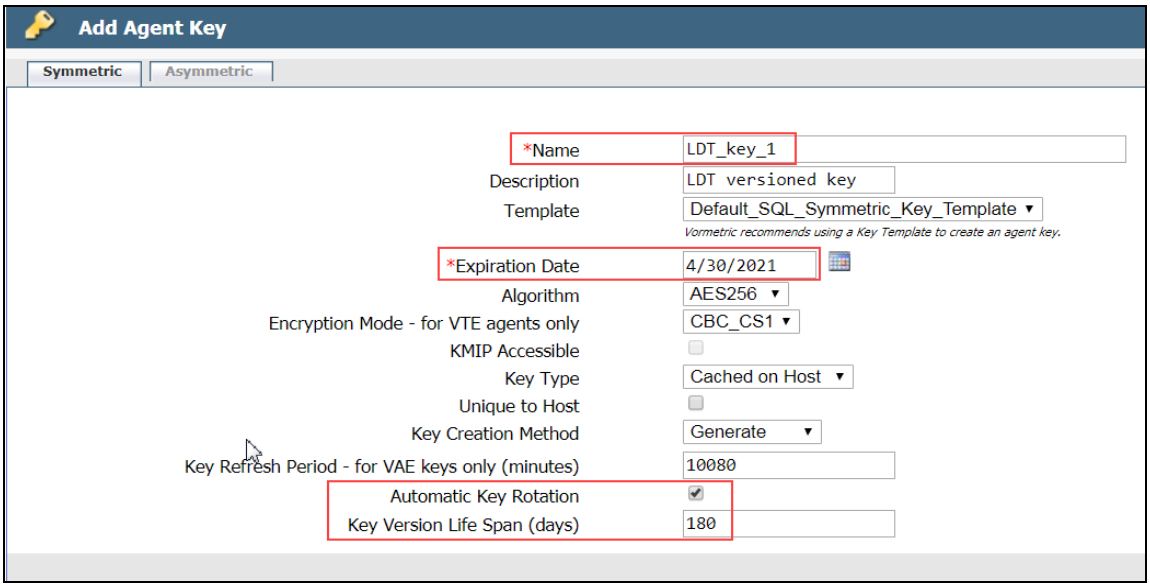
Field	Description
Name	The user-defined name for the key.
Expiration Date	CTE-LDT uses the initial transformation key specified in the CTE-LDT policy until this date occurs. On the expiration date, CTE-LDT creates a new version of the transformation key. Thereafter, it creates new versions based on the **Key Version Life Span**. You *must* specify an expiration date. Without an Expiration Date, a versioned key is always viewed as the initial key version. This means that it does not trigger CipherTrust Transparent Encryption - Live Data Transformation during initial transformation or subsequent key rotations.
Algorithm	For CTE-LDT, this must be set to AES256.
Encryption Mode	For CTE-LDT, this can be CBC or CBC CS1.
Automatic Key Rotation	For CTE-LDT, this must be enabled. Automatic key rotation is what makes this key a versioned key. After you enable this option, the DSM displays the **Key Version Life Span (Days)** field.
Key Version Life Span (Days)	The frequency of key rotation, in days. This value determines how often the DSM automatically generates a new version of the key after the initial expiration date has been reached. Creating a new version of the key triggers CipherTrust Transparent Encryption - Live Data Transformation on all GuardPoints using the key.

> **Note:** It is important to understand the interaction between the **Expiration Date** and the **Key Version Life Span**. CTE-LDT uses the initial version of the key until the Expiration Date. The Life Span has no effect until after the Expiration Date passes. Unless an administrator forces a rekey manually, there is no key version rotation and no live data transformation until the Expiration Date arrives.

When creating or modifying a versioned key, set the key version life span with care. The rekey process uses system resources, and it may contend with host applications that also require these resources. The more often you rekey, the more often the CTE-LDT process requires system resources. Determining the life span is a balance between security, compliance, and convenience. If you have no contention problems between the CTE-LDT process and your host applications, you can rekey frequently. If you do encounter contention problems, then you may want to choose a longer interval between key versions.

4. When you are done, click **Ok** to save the key.

The following figure shows an example of adding a versioned key. The key name is LDT_key_1, it expires on April 30, 2021, and it has a key version life span of 180 days. That means the DSM generates a new version of this key on April 30, 2021, which kicks off the CTE-LDT process for the first time. The DSM then generates a new key version every 180 days and automatically starts the CTE-LDT process over again.



For information about how versioned keys are rotated throughout the lifetime of the CTE-LDT policies that use them, see "CTE-LDT Runtime Flow" on page 14. For information about how to manually rotate a versioned key at any time, see "Rotating Encryption Keys (Rekey)" on page 51.

Viewing Versioned Key Information

To obtain information about keys, use the DSM Management Console. For example, you might use this procedure to:

- Find out which keys are versioned keys, and therefore available for use in CTE-LDT policies.
- Locate a past key version so you can use it to decrypt old data.

To view keys in the DSM Management Console:

1. In the DSM Management Console, click **Keys > Agent Keys > Keys**.

   The currently defined keys display. You can filter the list by key name using the **Search** box.

2. To find your versioned keys, look for a checked box in the **Versioned** column.

   If the **Current Version** for a key is 0, then this is the first version of the key and no key rotation has taken place. If the **Current Version** is greater than 0, the key has been rotated. Either the key version expired, which triggered an automatic rotation, or a Administrator manually initiated a rekey as described in "Manual Key Rotation" on page 52.

3. To find all versions of a key, click the name of the key in the **Name** column.
   - The **General** tab shows the most recent version of the key.
   - The **Versions** tab shows the key versions in reverse chronological order (most recent first). If the key has been rotated many times, the list takes up multiple pages. Click the arrow buttons to move from page to page.

Creating CTE-LDT Policies

CTE-LDT uses a single Live Data Transformation encryption policy to address both data transformation and ongoing protection. In contrast, if you do not use CTE-LDT, you need a separate policy for initial data transformation, or rekey, and another policy to protect the data while it is in production use. For more information about CTE-LDT policies, see "CTE-LDT Policies" on page 14.

To create a Live Data Transformation encryption policy:

1. Log into the DSM Management Console and switch to the domain you want to use.
2. Select **Policies > Manage Policies > Manage Policies**.
3. Click **Add**.
4. In **Policy Type**, choose **Live Data Transformation**.

   A default security access rule is automatically added in Security Rules. The action is `key_op`, and the effects are Permit and Apply Key. This rule permits key operations, for the policy, on all resources, without denying user or application access to resources. This allows it to perform a rekey operation whenever the encryption key rotates to a new version. This rule is required by CTE-LDT, so you cannot edit it, move it, or delete it.
5. Fill in the other fields as appropriate for an CTE-LDT policy. Some of the fields have special considerations when you use CTE-LDT:
   - **Policy Type**: Must be Live Data Transformation.
   - **Learn Mode**: Learn Mode provides a temporary method for disabling the blocking behavior of CTE/CTE-LDT policies. While useful for quality assurance, troubleshooting, and mitigating deployment risk, Learn Mode is not intended to be enabled permanently for a policy in production. This prevents the policy Deny rules from functioning as designed in the policy rule set.

     Ensure that the policy is properly configured for use in Learn Mode. Any Security Rule that contains a Deny effect must have Apply Key applied as well. This is to prevent data from being written in mixed states, resulting in the loss of access or data corruption.

     Apply Key will have no effect when combined with a Deny rule unless the policy is in Learn Mode.
   - **Security Rules**: The Security Rules panel contains the policy rules for the production workload. One default rule is added automatically to every CTE-LDT policy, as described above. After this default rule, the other security rules are specified in the same way, whether or not CTE-LDT is in use.

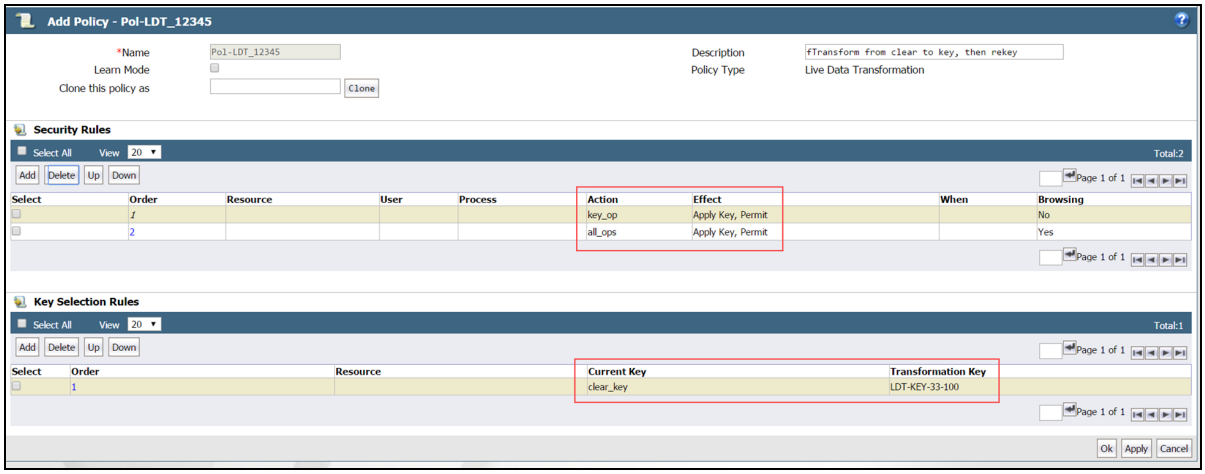
     If you set the "**When**" clause of key_op rules in CTE-LDT and a time period is set, it affects user I/O operations, not CTE-LDT I/O operations. CTE-LDT starts rekeying the GuardPoint regardless of the value in the key_op rule's When clause. CTE-LDT performs rekey operations according to the QoS schedule assigned to the host.
   - **Key Selection Rules, Current Key**: Indicates the state of the data before the CTE-LDT policy is applied to the GuardPoint. The data might be clear text, or it might already have been encrypted using a non-versioned key.

     Specify *clear_key* in Current Key if the data has not yet been encrypted. Otherwise, specify the name of the non-versioned key that was used to encrypt the data into its current state.
   - **Key Selection Rules, Transformation Key**: The **Transformation Key** is the versioned key that is applied to the data for its initial transformation. After the initial transformation, it rotates every time its life span ends. When the key rotates, all of its properties, including the key name and cryptographic algorithm, remain unchanged, except for the key material for encryption. Specify the versioned key name in the **Transformation Key** column. You can enter `clear_key` to transform your data to clear using CTE-LDT. `clear_key` is an exception to a non-versioned key in Transformation Key.
   - **Key Selection Rules, Resource**: Allows you to add or edit a resource set. A resource set is a named collection of directories, files, or both, to which a user or process is permitted or denied access.

     > **Note:** Thales strongly recommends limiting the number of resource sets in CTE-LDT policies to 50 resources or less.

The following example shows a simple CTE-LDT policy that encrypts clear-text files using a versioned key named LDT-Key-33-100, as shown in the Key Selection Rules panel. The example `key_op` and `all_ops` actions in the Security Rules panel grant user and application access to files at all times, including during initial CTE-LDT and subsequent key rotations.

Figure 4-1: CTE-LDT Policy that encrypts clear-text files



Quality of Service

CTE-LDT runs in real time, while users actively interact with applications. This could impact performance. However, CTE-LDT is designed to not adversely affect application or system performance.

Purpose of QoS

Quality of Service (QoS) provides tools for an administrator to minimize the effect of CTE-LDT on system and application performance. It provides a set of parameters that administrators can set to control CTE-LDT use of system resources, primarily CPU and I/O bandwidth. When the QoS parameters are set appropriately, CTE-LDT stays within the defined boundaries to ensure that critical user applications are not adversely affected by CTE-LDT operations.

Manage CTE-LDT Impact

Administrators can pause or resume CTE-LDT operations to manage and control CTE-LDT impact to application workload. When data transformation occurs, either during initial or subsequent transformations, it requires substantial host CPU and I/O resources. This can cause contention for resources between the applications simultaneously running on the protected host. The administrator specifies QoS settings on each host, or at a host group level, that is using CTE-LDT. When CTE-LDT is running, QoS monitors CPU or rekey/scan rate on the host and enforces the QoS settings. QoS can also monitor and enforce an administrator imposed limit on the volume of data undergoing rekey per second. The QoS settings enable you to strike a balance between completing an CTE-LDT process and not interfering with host application performance.

Monitor and Control CPU Usage

QoS monitors and controls the use of host system resources during CTE-LDT, specifically, CPU usage and rekey/scan rate.

> **Note**
> You can control CPU usage or rekey/scan I/O rate, but not both. The CPU usage and rekey/scan I/O rate options are mutually exclusive.

Monitor and Control Rekey/Scan I/O Rate

Starting with VTE version 6.1.2 and onwards, you can choose Rekey I/O Rate as a threshold to control the CTE-LDT processing rate. When this threshold is entered, the Quality of Service continuously monitors CTE-LDT transformation and enforces the specified amount of data during:

- **Rekeying**—CTE-LDT is transforming the data on active GuardPoints based on the new key version.
- **Scanning**—CTE-LDT is analyzing files in GuardPoints. Scanning occurs:
  - Before initial transformation (Linux only)
  - Before a rekey (Linux only)
  - Following an interrupted rekey, such as a reboot on Linux or Windows, and also a directory rename or directory deletion on Windows

The Rekey I/O Rate is either 0, or greater than 0. By specifying 0, Quality of Service resets the previously specified Rekey I/O Rate, if any, and then stops monitoring the LDT rekey/scan processing rate.

When the Rekey I/O Rate is set, Quality of Service ignores the CPU and/or the iow (IOWAIT) thresholds previously set. After resetting the Rekey I/O Rate to 0, the Quality of Service will resume enforcing the CPU and/or the iow thresholds, provided one or both are set.

To rekey at full throttle, you can set the Rekey I/O Rate, the CPU, and IOWAIT thresholds to 0.

> **Note**
> IOWAIT has been deprecated on Linux, and it is not supported on Windows. Instead, use Rekey IO Rate to control the CTE-LDT I/O utilization on both Linux and Windows.

With the DSM version 6.2.0 and later, you can set the Rekey Rate or CPU threshold for all of the CTE-LDT GuardPoints on a host, or for all of the GuardPoints in a given host group, on the **GuardPoints** tab in the DSM.

You can override the DSM settings locally on a single host using the `voradmin ldt ior <iorate>` command on that host. When you do so, the `voradamin` setting overrides the Rekey I/O Rate or CPU Threshold set in the DSM.

A tolerance level is associated with the Rekey I/O Rate. Together, the tolerance and Rekey I/O Rate specify a range for the CTE-LDT processing rate. The Quality of Service selects a proper tolerance for a Rekey I/O Rate provided through the `voradmin` command, and maintains the CTE-LDT processing rate at the specified Rekey I/O Rate plus or minus the tolerance. The tolerance is selected as follows:

- When the Rekey I/O Rate is less than or equal to 10MB/sec, the tolerance is 3MB/sec.
- When the Rekey I/O Rate is greater than 10MB/sec. and less than 50MB/sec, the tolerance is 4MB/sec.
- When the Rekey I/O Rate is at 50MB/sec or higher, the tolerance is 10% of the specified Rekey I/O Rate.

To set or reset Rekey I/O Rate on a single host, use the `voradmin` command as follows:

- To set the threshold of 50 MB/sec., use the following command:

  ```
  # voradmin ldt ior 50
  ```

- To reset the current threshold:

  ```
  # voradmin ldt ior 0
  ```

For more information about setting the Rekey I/O Rate using `voradmin`, see "Select and Set Rekey I/O Rate" on page 46.

Monitor and Control I/O Wait Time (Linux only)

> **CAUTION**
> **The Linux I/O wait option has been deprecated in favor of Rekey I/O Rate. Support for the I/O wait option will eventually be removed from CTE. If you are currently using I/O wait time option with QoS, please plan on switching from I/O Wait Time to Rekey I/O rate.**

You can choose to add **I/O wait time** for monitoring and control. The I/O wait time threshold refers to the system's peak I/O wait time during its production workload. For a given threshold, you can specify a tolerance level that is the amount of I/O overhead allowed by CTE-LDT for rekeying data. By specifying the I/O wait time, and the tolerance level, QoS continuously monitors your system's average I/O wait time during CTE-LDT operations. It applies the tolerance factor to the average I/O wait time to adjust CTE-LDT I/O operations. This assures that the system's average I/O wait time remains within the desired range and does not exceed the threshold.

The desired range is the specified average I/O wait time (milliseconds) plus or minus the specified tolerance. **Tolerance** provides the range in which the average I/O wait time, which includes CTE-LDT I/O operations, is maintained.

For example, you may specify a desired I/O wait time during rekey. QoS maintains the desired I/O load for rekey according to the specified I/O wait time. If the total I/O wait time increases beyond the desired I/O wait time, QOS reduces the rekey I/O load by some percentage until the desired I/O wait time is achieved or the rekey I/O load reaches down to a preset N number of rekey operations.

QoS Scheduling During Backup/Restore

QoS scheduling plays an important role when backing up/restoring data without the *Apply Key* rule applied to the backup/restore process. During backup/restore, you **must** pause CTE-LDT operations before taking backups. QoS scheduling allows the administrator to enter the schedule for QoS aligned with the backup schedule, and pause the CTE-LDT processes for the duration of the backup. The schedule specifies which days of the week, and what times of day, CTE-LDT is permitted to run. CTE-LDT cannot run at any time that is not permitted by the QoS schedule. QoS suspends CTE-LDT operations at all times outside of the schedule.

When setting a QoS schedule, consider your system and application peak demand periods during the day and week. Also consider your schedule for data backups. Schedule CTE-LDT to pause when you need all available system resources for other tasks, such as meeting peak user demand or performing data backups.

> **Note**
> On Windows, if your backup applications are using VSS, then you do not need to pause CTE-LDT on Windows.

How to Set QoS

To set QoS for a host to enforce a threshold on CTE-LDT based on CPU percentage:

1. Click **Hosts > Hosts > *hostname* > GuardPoint**.
2. Set the **Schedule** parameter:

   Choose one of the options from the drop-down list. By default, the list contains WEEKNIGHTS (rekey performed between 12 AM and 7:00 AM), WEEKENDS, or ANY_TIME. The default setting is ANY_TIME. By limiting CTE-LDT to periods of low application usage, you minimize the potential for resource contention between applications and CTE-LDT.

You can add to the choices in the Schedule drop-down by creating custom schedules. See "Creating a Custom QoS Schedule" below.

3. Enter the Rekey IO Rate threshold or percentage of CPU resources to allocate to CTE-LDT in the **Set % of available CPU usage for rekey** field:

   Choose a Rekey IO Rate threshold or the percentage of total CPU resources that CTE-LDT processes can use. CTE-LDT applies a tolerance of +/- 2% on CPU threshold settings up to 7%. For threshold over 7%, CTE-LDT applies a tolerance of +/- 4%.

   If CPU% is set to 0, QoS stops monitoring CPU usage and CTE-LDT operations run at maximum rekey rate within the available system resources. Setting CPU percentage to 0 does not affect CTE-LDT schedules, so CTE-LDT operations are suspended and resumed per QoS schedule.

   If you do not enter CPU threshold percentage, CTE-LDT applies CPU threshold of 5% capped by default.

   If CPU% is set to a very high value, such as 25%, the rekey process competes with other applications to use as many CPU cycles as it can. As a best practice, start with a setting of 10%, and increase or decrease it slowly by 5% until it reaches a reasonable level that does not adversely affect the performance of user applications. The higher the percentage, the more quickly CTE-LDT completes its processing. However, this speed causes increased competition for resources, which can significantly degrade the performance of other applications using this host.

   CAUTION
   Do not set CPU% to a very high value, or to 100%, in an attempt to force faster data transformation. This can potentially exhaust other system resources.

4. Set the **Cap CPU Allocation** for rekey.

   Check this box to specify that the CPU allowance must never exceed the percentage set in CPU%. If Cap CPU Allocation is not checked, and additional CPU resources are available on the host, CTE-LDT consumes part of the available resources for rekey above the CPU threshold. Exceeding the threshold may impact your production workload as your production CPU resource consumption fluctuates over time.

   For example, if CPU% is set to 10%, but Cap CPU Allocation is not set, the rekey process continues consuming available CPU cycles after reaching 10% CPU utilization, at which point the rekey process starts contending with applications for CPU cycles.

   Note: If you choose Rekey IO Rate threshold to control the CTE-LDT processing rate, you do not need to also set the CPU threshold for CTE-LDT.

Creating a Custom QoS Schedule

CTE-LDT provides three predefined QoS schedules: ANY_TIME, WEEKENDS, and WEEKNIGHTS. You can define additional custom schedules that can be assigned to any host on the GuardPoints tab.

1. In the DSM, choose **Hosts > QoS Schedules**.
2. Click **Add**.
3. Enter a unique name for the new schedule.

   This name displays in the **Schedule** dropdown list when you set QoS parameters.
4. Click **Add**.
5. Choose the start day/time and the end day/time for CTE-LDT operations.
6. Click **OK**.

QoS Best Practices

This section gives tips and examples to help you set QoS parameters for the best results.

General Best Practices for QoS

- Use Rekey I/O Rate threshold to limit CTE-LDT impact, if any, to your production workloads. Rekey I/O Rate approach is a simpler method for a Administrator or system administrator to enforce a limit on the volume of data that CTE-LDT should rekey per second. You can choose a threshold, in units of MBs per second, which is a small percentage of peak IOPS from your production workload.

  > **Note:** When choosing a threshold on CTE protected hosts with GuardPoints over NFS/CIFS shares, you must consider network bandwidth between your host and NFS/CIFS servers. QoS does not monitor the impact of LDT operations on network connections between your hosts and NFS/CIFS servers. However, the rate selected as LDT Rekey IO rate directly correlates to the network bandwidth to the target NFS/CIFS servers. In the follow-up discussions for selecting optimal rekey IO rate, you must monitor and collect the network traffic instead of disk IO transfers.

  > **Note:** The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

- You will see the effects of QoS settings only if the number and/or types of files in the GuardPoints stress the rekey or scan processes. On hosts with a relatively small number of files, the rekey or scan process may complete quickly without hitting a threshold and causing throttling to occur.
- Use QoS CPU parameters as an alternate method for controlling the effect CTE-LDT has on application performance.

  Set limits on CTE-LDT CPU usage whenever runtime monitoring shows that user applications are affected by CTE-LDT. Start by setting the CPU parameter to 10%, then increase or decrease in 5% intervals, as needed, to tune the CPU allocation. When an acceptable level is reached, and CTE-LDT is not noticeably affecting user applications, leave the QoS CPU parameters at a constant setting.

- Use monitoring tools.

  Monitor host CPU utilization with tools like `vmstat`, `top`, and `iotop` on Linux and `perfmon` on Windows.

  You can also monitor and obtain statistics with the `voradmin ldt stats` command.

  For more information about `voradmin ldt stats`, see "Obtaining CTE-LDT Statistics at the Command Line" on page 104.

- Select CPU resource allocation for CTE-LDT from 1% to the available limit minus 5%.

  If the monitoring tools indicate system CPU usage, without CTE-LDT, it is at N%, available CPU resource is M%, where M = 100 - N. Select a percentage within 1 - (M - 5) to allocate to CTE-LDT CPU usage. However, remember that QoS tolerates 2% - 4% leeway in the actual CPU usage, so adjust your selection by 2 - 4%.

- Do not set CPU resources to 0% or 100% in an attempt to minimize or stop CTE-LDT

  A CPU% value of 0 or 100 is reserved for disabling the QoS CPU monitoring function. This does not stop CTE-LDT or minimize its resource usage; rather the opposite. It enables CTE-LDT to run with its maximum rekey rate. Note that when CPU % is not set, and it shows 100% setting on the DSM, CTE-LDT hosts enforce a 5% CPU threshold by default. We recommend that you change the 100% setting to 5% capped on the hosts installed with VTE versions lower than v6.1.

- Cap the CPU allocation.

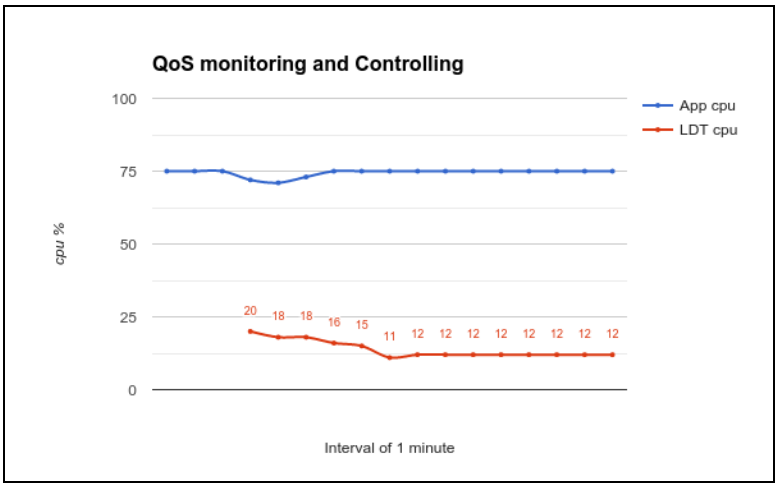
  QoS provides a **CAP CPU Allocation** parameter. Set this parameter to **True**. This ensures that CTE-LDT resource usage never exceeds the allocated percentage.

- Apply **iowait** threshold very carefully to try controlling CTE-LDT I/O load.

Example: Setting QoS before starting CTE-LDT

You can be proactive and set up QoS parameters before enabling GuardPoints that are protected with CTE-LDT policies. This ensures QoS starts monitoring and controlling CTE-LDT resource usage from the start. The following graph shows an example where 10% of the CPU is assigned to CTE-LDT. QoS makes sure that CTE-LDT is restricted to use only 10% of the CPU. There is a tolerance level of +/- 4%, so actual CTE-LDT usage can range between 5% and 15% of CPU. In the following example, applications use 75% of the CPU resources. As the graph shows, when CTE-LDT starts, application CPU utilization drops for a moment, because CTE-LDT exceeds the CPU threshold. QoS immediately reduces CTE-LDT's CPU usage to 12%, which is within tolerance levels for a 10% setting, and the application CPU share returns to normal.

Figure 4-2: QoS makes visible improvement immediately when CTE-LDT starts



The graph above was obtained on a Linux system running sysbench.

- To find the amount of CPU resources currently in use by applications, type:

  ```
  # top -n 1 -b | grep sysbench | awk 'BEGIN {cpu=0} {cpu += $9} END {print cpu}'
  ```

- To find the amount of CPU currently in use by the CTE-LDT-protected host, type:

  ```
  # top | grep Cpu
  ```

- To find the amount of CPU currently in use by CTE-LDT, type:

  ```
  # voradmin ldt stats | grep CPU
  ```

Example: Monitoring and controlling resource usage during CTE-LDT

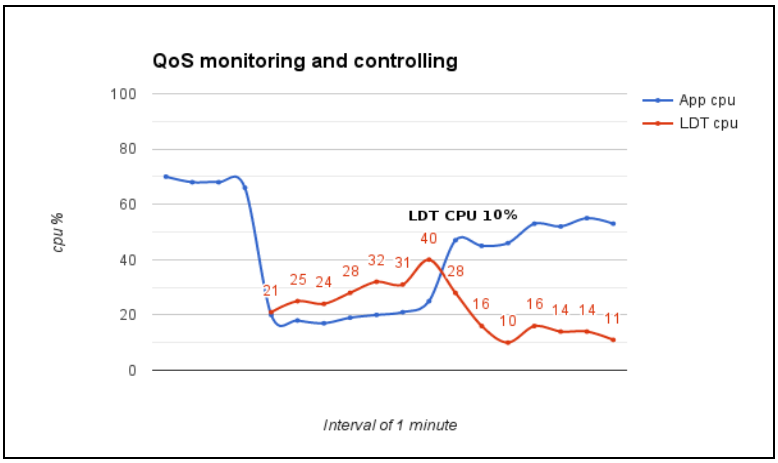
Suppose that CTE-LDT has started with CPU set to 25%, and users realize their applications are affected. For example, there might be a higher than expected level of CTE-LDT I/O operations. To return application performance to normal, reduce the CPU allocation for CTE-LDT.

1. Set the **CPU parameter** to a lower value, such as 10%.
2. Select the **Cap CPU Allocation** option.
3. Set the I/O wait parameter.

QoS restricts CTE-LDT CPU usage to 10%. The application user should monitor their application. If the application's performance is still affected, reduce the CPU parameter further, such as to 5%. Repeat this procedure until application performance returns to a satisfactory level.

The following graph shows an example where QoS is not enabled to monitor and control CTE-LDT CPU usage from the start. When CTE-LDT starts, application CPU usage drops from 65% to 20%. By setting the QoS CPU parameter to 10%, application usage is greatly improved.

Figure 4-3: CPU usage allocation before and after QoS CPU parameter is set



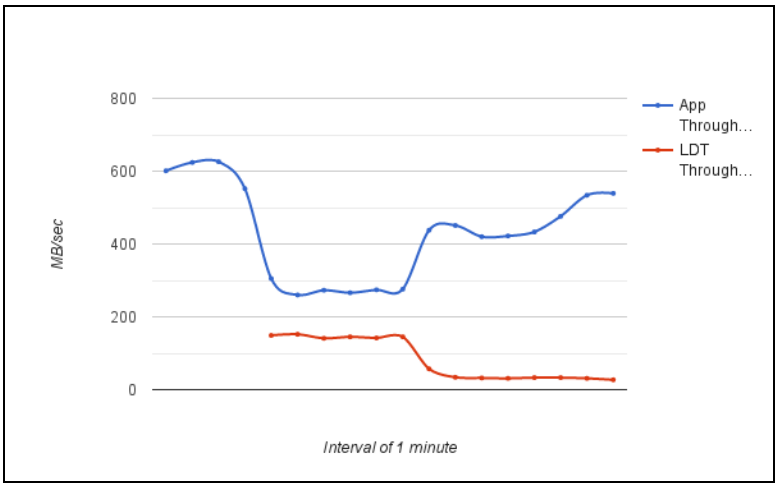
Example: How QoS CPU settings affect I/O bandwidth

Controlling CPU utilization indirectly controls I/O bandwidth. When CTE-LDT is consuming less CPU resources, it is usually performing fewer operations of all kinds, including input and output.

The following graph shows how an application's I/O throughput is affected by CTE-LDT, and how QoS can reduce this effect through monitoring and controlling the CTE-LDT CPU resource usage.

This graph is from the same system described in "Example: Monitoring and controlling resource usage during CTE-LDT" on the previous page. No QoS parameters are set at first. When CTE-LDT starts, application I/O drops from more than 600 MB per second to below 30 MB per second. By setting the QoS CPU parameter to 10%, the CTE-LDT I/O operations drop to below the application's ability to perform so application I/O operations is greatly improved.

Figure 4-4: I/O operations before and after QoS CPU limit is set



To obtain the data for this graph:

- Use `iotop` and benchmarking tools on an RHEL system to obtain application I/O throughput.
- Use `voradmin ldt stats` to obtain the CTE-LDT I/O throughput and rekey rate.

Determine and Set the I/O wait time

Monitor the average I/O wait time using `iostat` on Linux, or `perfmon` on Windows, during the production workload. Thales recommends monitoring the average I/O wait time during the peak levels of the workload.

1. Run the `iostat` command and specify the number and duration of intervals to determine `iowait` time.

   ```
   # iostat -txm 10 2
   ```

   > **Note:** For Windows, use `perfmon` to obtain: `\PhysicalDisk(_total)\Avg. Disk sec/Transfer`

2. Get the total I/O wait time from the sampled data. It is the sum of the await field of disks.

   > **Note:** Do not include await values from volumes and device mappers.

   Figure 4-5: Example of iostat data

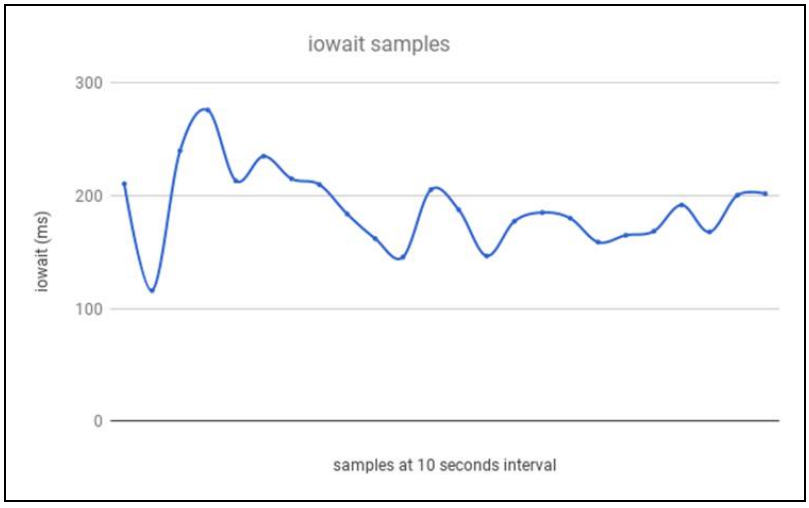
   ```
   Monday 04 September 2017 03:13:59  PDT
   avg-cpu:  %user   %nice %system %iowait  %steal   %idle
             0.18    0.00    2.52   42.03    0.00   55.27

   Device:         rrqm/s   wrqm/s     r/s     w/s    rMB/s    wMB/s avgrq-sz avgqu-sz   await  svctm  %util
   fd0               0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00    0.00   0.00   0.00
   sda               0.00     0.00    0.03    2.17     0.00     0.03    28.77     0.00    0.61   0.18   0.04
   sdb               0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00    0.00   0.00   0.00
   sdc               0.00     0.33    0.00    0.47     0.00     0.00    14.86     0.00    0.00   0.00   0.00
   sdd               0.07    17.20    0.53   12.53     0.01     0.18    29.31     1.14   88.90  76.30  99.70
   sde               0.00    75.57    0.07  839.20     0.00     4.04     9.86   141.87  169.15   1.19  99.94
   sdg               0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00    0.00   0.00   0.00
   sdh               0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00    0.00   0.00   0.00
   sdf               0.00     0.00    0.00    0.00     0.00     0.00     0.00     0.00    0.00   0.00   0.00
   dm-0              0.00     0.00    0.11    2.17     0.00     0.01     9.19     0.02    7.49   0.28   0.06
   dm-1              0.00     0.00    0.00    0.00     0.00     0.00     8.00     0.00    1.91   1.14   0.00
   dm-2              0.00     0.00    0.15    4.97     0.01     0.02    10.73     0.05    8.80   0.04   0.02
   ```

   For the above example:

   ```
   Total I/O wait = .61+ 88.9 + 169.15 = 258.66
   ```

3. Obtain multiple samples at 10 second intervals. Estimate the `iowait` time when it stabilizes.

   In the following chart, the I/O wait time is nearly 200ms when in a stable state.



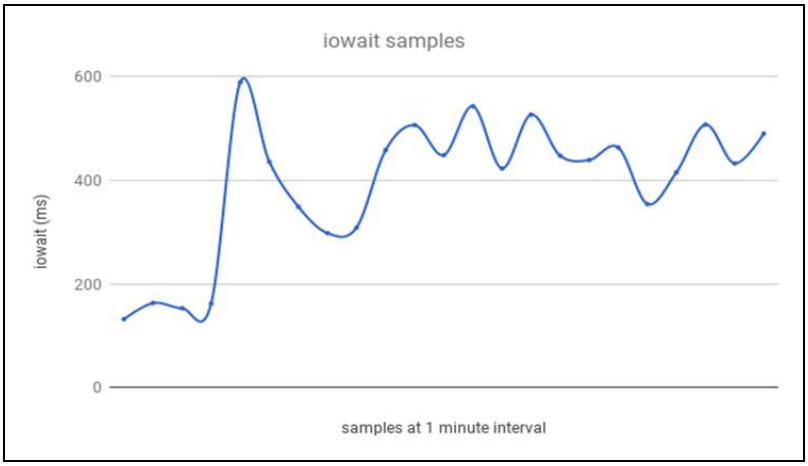
4. Estimate the `iowait` threshold by running CTE-LDT and watching the degradation in I/O wait time. Set the `iowait` threshold and/or tolerance parameters in units of milliseconds using voradmin command, type:

   `# voradmin ldt iow` *`iowait`* `[`*`tolerance`*`]`

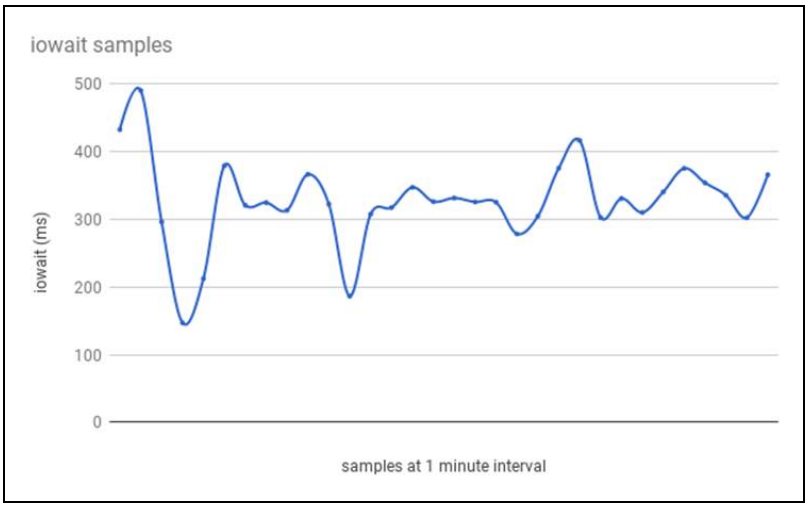
   For example, to set the iowait to 350 milliseconds and the tolerance to 10 milliseconds, you would enter:

   `# voradmin ldt iow 350 10`

In the following graph, observe that once CTE-LDT starts, the total I/O wait time jumps to 600 ms and oscillates between 400 to 500 ms. Without CTE-LDT, I/O wait time was 200 ms. If the application I/O wait time suffers because of CTE-LDT, reduce the CTE-LDT I/O usage by setting `iowait` threshold to a suitable level between 200 to 500 ms. In this example, we have chosen 350 ms as the desired/suitable level of total I/O while CTE-LDT and application workloads are running concurrently.



From the following graph, observe that QoS tried to control I/O wait time and now it oscillates near about 350 ms.



The following table explains what is monitored and controlled by QoS when CPU or I/O resource thresholds are set.

Resource Threshold Set	Monitored and Controlled by QoS
CPU (value must be greater than 0.)	CPU utilization to adjust CTE-LDT operations to maintain the set CPU threshold
I/O wait (value must be greater than 0.)	I/O average wait time to adjust CTE-LDT operations to maintain the set I/O wait threshold

Resource Threshold Set	Monitored and Controlled by QoS
CPU and I/O wait	CPU and I/O, both resources are controlled by their threshold limits
CPU and I/O wait set to 0	No resource is controlled or monitored

Select and Set Rekey I/O Rate

You can choose to set the Rekey I/O Rate to control I/O operations from CTE-LDT to minimize CTE-LDT impact to your production workload. It's assumed that you already know the maximum IOPS on your host system during your production workload. With this information, you can choose a threshold for Rekey I/O Rate and enforce the selected threshold during CipherTrust Transparent Encryption - Live Data Transformation. The work flow is as follows:

1. Set **Rekey I/O Rate** threshold using `voradmin` or in the DSM Management Console.
2. QoS retrieves the threshold and starts monitoring and controlling CTE-LDT according to the specified threshold and the tolerance factor corresponding to the threshold.
3. The selected threshold will be in effect within 2 to 4 minutes after entering the threshold.

When Rekey I/O Rate and CPU or IOWAIT thresholds are set, QoS will monitor and control the CTE-LDT processing rate based on the Rekey I/O Rate threshold. The CPU and/or IOWAIT thresholds will be ignored.

Set Rekey I/O Rate Threshold

1. Set **Rekey I/O Rate** threshold by using `voradmin`:

   ```
   # voradmin ldt ior 10
   ```

   You can also set the Rekey I/O Rate for one or more managed hosts on the QoS settings area of the **GuardPoint** tab under host configuration in the DSM (DSM version 6.2.0 or later). For more information about using the DSM method, see "How to Set QoS" on page 39.

   In the `voradmin` example above, QoS enforces the threshold of 10MB/sec with the tolerance of +/- 3MB/second. Effectively, CTE-LDT attempts to rekey the amount of data in the range of 7MBs/second to 13MB/second.

   On Linux and Windows, you can use `voradmin ldt ior` to report the current threshold setting without specifying a value for threshold:

   ```
   # voradmin ldt ior
   QoS Rekey I/O rate threshold: 10 MB/sec
   QoS Rekey I/O tolerance: 3 MB/sec
   ```

2. Be sure the threshold you enter is appropriate for your production workload. To verify this:
   a. Observe the Rekey I/O Rate for a few minutes using `voradmin`.

      On Linux, you can do this using:

      ```
      # voradmin ldt stats
      Host level statistics:
      File stats: rekeyed=202390, passed=0, created=0, removed=0
      Data stats: rekeyed=6.2GB, truncated=0.0MB
      QoS: IOR threshold=10MB/sec, tolerance=3MB/sec
      current_rekey_rate=2MB/sec, current_iow=0ms
      load_factor=50, delay_factor=0, delay_scan=0
      ```

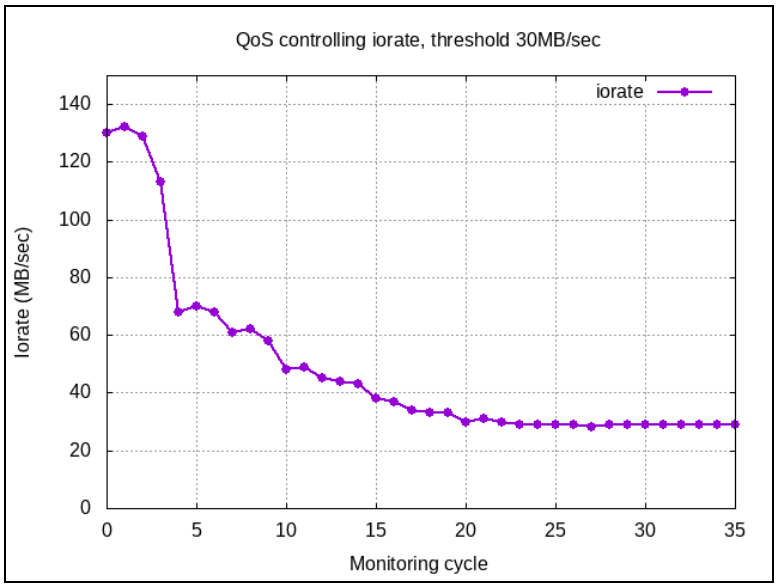
On Windows, you can do this using:

```
voradmin ldt monitor
Host Stats:
Total number of Guard Points    = 1
Rekey Status                    = Rekey done (Finished rekey on 1 out of 1 GP's)
Total files to be transformed   = 0
Total files transformed         = 0
Total files in progress         = 0
Total transformation threads    = 0

Current rekey rate              = 0 KB/s
Rekey IO rate threshold         = 1000 MB/s
Rekey IO rate tolerance         = 4 MB/s
```

b. Set an appropriate threshold. Do not set the threshold value too high, as QoS might not be able to achieve it because of other resource bottlenecks.

3. Check the QoS controlling rekey rate.

   QoS will monitor and control CTE-LDT utilization using the specified threshold. The following figure shows an example of how QoS monitors and controls CTE-LDT utilization. In this example, the threshold is 30 MB/sec. Throughput of CTE-LDT was nearly 130 MB/sec. QoS brings it down to within the range of 30 MB/second.



4. Disable QoS.

   QoS will not monitor and control resources when all the thresholds, CPU, Rekey I/O rate, and IOWAIT are set to 0. When Rekey I/O Rate and IOWAIT are not explicitly set, it is considered to be set to 90 MB/second.

   QoS continues to apply its schedules for suspending CTE-LDT operations at certain days and times regardless of what values are set for CPU, Rekey I/O Rate, and IOWAIT thresholds.

Summary of QoS Resources

The following table summarizes the available thresholds and the actions of QoS module to enforce the set thresholds:

Scenario	QoS Action
Only Rekey I/O Rate threshold is set	Monitor and control the CTE-LDT processing rate based on Rekey I/O Rate
Rekey I/O Rate and CPU threshold are set	Monitor and control the CTE-LDT processing rate based on Rekey I/O Rate. CPU threshold is ignored.
Rekey I/O Rate, CPU, and IOWAIT thresholds are set	Monitor and control the CTE-LDT processing rate based on Rekey I/O Rate. CPU and IOWAIT thresholds are ignored.

Creating a CTE-LDT GuardPoint

After you have installed the license and registered the CTE-LDT host (see Chapter 3: "Setting Up CTE-LDT" on page 21), you can create an CTE-LDT GuardPoint on the host. When you create the CTE-LDT GuardPoint, you select a Live Data Transformation policy and apply that policy with its transformation keys to that GuardPoint. CTE automatically gets the Quality of Service settings from the **GuardPoint** tab of the Host entry in the DSM.

> **Note**
> The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

This section describes the following scenarios:

- "Creating a CTE-LDT GuardPoint for a Linux Directory or NFS Share" below
- "Creating a CTE-LDT GuardPoint for a Windows Directory or CIFS Share" on the next page
- "Converting a Non-CTE-LDT GuardPoint to a CTE-LDT GuardPoint" on page 50

> **Note**
> If you intend to guard a Linux directory that is accessed through a NAS server, the directory must be accessed using the NFS version 3 or version 4 protocol. CTE for Linux does not support accessing a Linux directory using CIFS.
>
> Similarly, if you intend to guard a Windows directory through a NAS server, the directory must be accessed using the CIFS protocol. CTE for Windows does not support accessing a Windows directory using NFS.

Creating a CTE-LDT GuardPoint for a Linux Directory or NFS Share

To create a CTE-LDT GuardPoint on what was previously an unprotected/unguarded Linux directory on a local file system or NFS share, do the following:

1. Create an Live Data Transformation policy that transforms data from clear text to a versioned key. In the policy, set the **Current Key** to `clear_key` and the **Transformation Key** to the versioned key. See "Creating CTE-LDT Policies" on page 35.
2. Set, or modify, the Quality of Service (QoS) parameters to account for CTE-LDT on all GuardPoints. See "Quality of Service" on page 37.

3. Click **Guard**.
4. In the Guard File System window, set the following values:
   a. Select a Live Data Transformation Policy.
   b. Select **Directory (Auto Guard or Manual Guard)**.
   c. Enter or browse for the directory to protect in the **Path** field.

      If you are specifying an NFS share, use the notation `/mountpoint/dir-path/`. For example, `/mnt/HR-Data/`.

      > **Note:** If you are protecting the same NFS share on multiple CTE hosts, the directory path for that NFS share can be different on each CTE host as long as you encrypt the NFS share with the same LDT policy on all hosts. For more information, see "Multiple GuardPoint Pathnames" on page 81.

   d. Do **not** check Automount. CTE-LDT is not supported on automounted file systems.
5. Check **Transform Sparse Regions** if you want those regions allocated with disk blocks.

   A *sparse region* is a region within the file size that has not yet been written to. Therefore, it is not allocated with disk blocks. Any attempt to read a sparse region reads stream of zeros as data. A file may have one or more sparse regions, or an entire file may be sparse.

   If you select **Transform Sparse Regions**, CTE-LDT transforms a file without checking or skipping sparse regions. Consequently, as CTE-LDT operations transform and fill sparse regions with encrypted stream of zeros, sparse regions are allocated with disk blocks. This increases the number of disk blocks utilized in the file system.

   If you do *not* select **Transform Sparse Regions**, CTE-LDT detects and skips transforming sparse regions. Therefore, it does not change the number of blocks utilized in the file system.
6. Click **OK**.

   The **GuardPoint** tab displays again. CTE-Live Data Transformation starts immediately after you enable the GuardPoint on the host.
7. Wait a moment and check the status of the GuardPoint in the Rekey Status column of the GuardPoint tab.
8. In the Status column, click the status icon (the green circle) to get status details about this GuardPoint.

   The GuardPoint Status window displays. For an explanation of all the items in the GuardPoint Status dialog, see "Obtaining Statistics in the DSM or CipherTrust Manager with GuardPoint Status" on page 102.

   > **Note:** When you first apply a GuardPoint, the key version shown in GuardPoint Status is 0. This indicates the first version of the key.
9. Wait for CTE-LDT to complete the rekey process for the GuardPoint. When the Rekey Status column shows 100%, the GuardPoint has been rekeyed and the GuardPoint status changes to rekeyed. For information about the values in the Rekey Status column, see "Checking the Rekey Status" on page 52.

Creating a CTE-LDT GuardPoint for a Windows Directory or CIFS Share

To create a CTE-LDT GuardPoint on what was previously an unprotected/unguarded Windows directory on a local file system or CIFS share, do the following:

1. Create a Live Data Transformation policy that transforms data from clear text to a versioned key. In the policy, set the **Current Key** to `clear_key` and the **Transformation Key** to the versioned key. See "Creating CTE-LDT Policies" on page 35.
2. Set, or modify, the Quality of Service (QoS) parameters to account for CTE-LDT on all GuardPoints. See "Quality of Service" on page 37.

3. Click **Guard**.
4. In the Guard File System window, set the following values:
   a. Select an CTE-LDT Policy.
   b. Select **Directory (Auto Guard or Manual Guard)**.
   c. Enter or browse for the directory to protect in the **Path** field.

      If you are specifying a CIFS share, use the notation `\\host\dir-path\`. For example, `\\myhost.com\share\HR-Data\`.

      When you enter \\ in the **Path** field, the DSM displays the **CIFS Share Credential** drop-down that lets you specify the CIFS credentials CTE should use to log into the CIFS share. You can enter multiple shares in the **Path** field but you can only enter one set of CIFS credentials. So the credentials you specify must work for all of the shares specified in the **Path** field.
   d. If you want to create a Secure Start GuardPoint, check **Secure Start**. For details on Secure Start, see the *CTE Agent for Windows Advanced Configuration and Integration Guide*.
   e. Check **Transform Sparse Regions** if you want those regions allocated with disk blocks.

      A *sparse region* is a region within the file size that has not yet been written to. Therefore, it is not allocated with disk blocks. Any attempt to read a sparse region reads stream of zeros as data. A file may have one or more sparse regions, or an entire file may be sparse.

      If you select **Transform Sparse Regions**, CTE-LDT transforms a file without checking or skipping sparse regions. Consequently, as CTE-LDT operations transform and fill sparse regions with encrypted stream of zeros, sparse regions are allocated with disk blocks. This increases the number of disk blocks utilized in the file system.

      If you do *not* select **Transform Sparse Regions**, CTE-LDT detects and skips transforming sparse regions. Therefore, it does not change the number of blocks utilized in the file system.
5. Click **OK**.

   The **GuardPoint** tab displays again. CipherTrust Transparent Encryption - Live Data Transformation starts immediately after you enable the GuardPoint on the host.
6. Wait a moment and check the status of the GuardPoint in the Rekey Status column of the GuardPoint tab.
7. In the Status column, click the status icon (the green circle) to get status details about this GuardPoint.

   The GuardPoint Status window displays. For an explanation of all the items in the GuardPoint Status dialog, see "Obtaining Statistics in the DSM or CipherTrust Manager with GuardPoint Status" on page 102.

   > **Note:** When you first apply a GuardPoint, the key version shown in GuardPoint Status is 0. This indicates the first version of the key.
8. Wait for CTE-LDT to complete the rekey process for the GuardPoint. When the Rekey Status column shows 100%, the GuardPoint has been rekeyed and the GuardPoint status changes to rekeyed. For information about the values in the Rekey Status column, see "Checking the Rekey Status" on page 52.

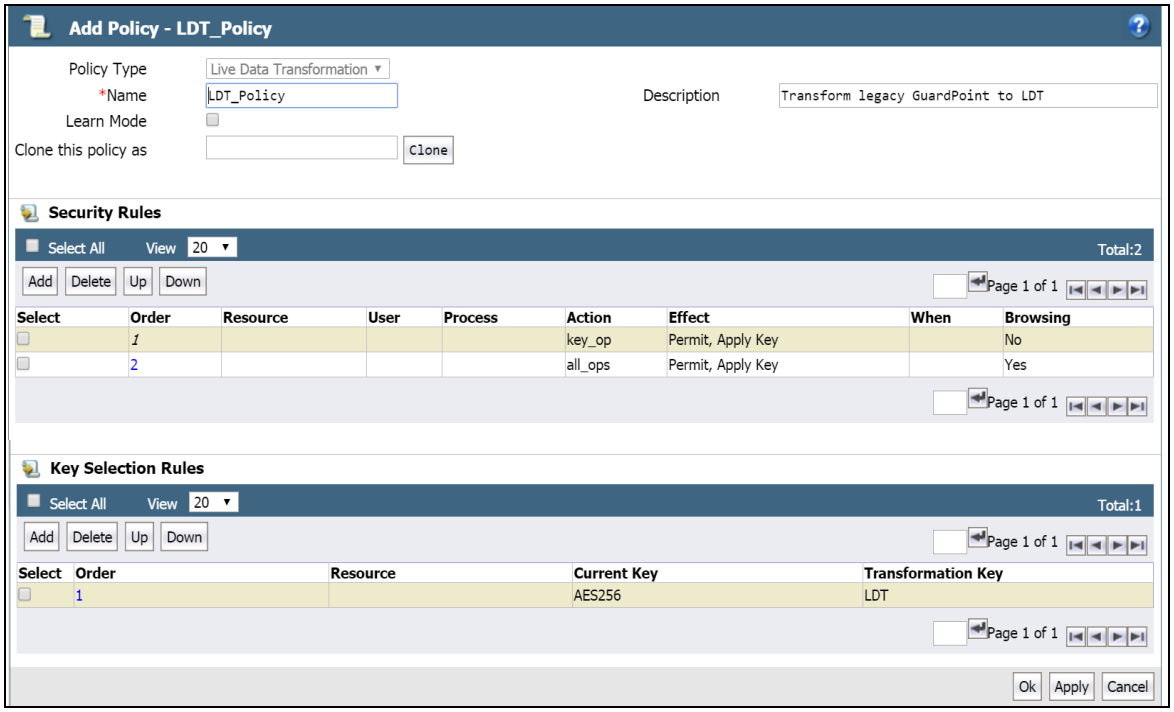
Converting a Non-CTE-LDT GuardPoint to a CTE-LDT GuardPoint

If you have an earlier version of CipherTrust Transparent Encryption installed, you most likely have set up GuardPoints. These legacy GuardPoints are protected with policies that do not make use of CipherTrust Transparent Encryption - Live Data Transformation. This means that you must stop all users from accessing the files in these GuardPoints while the policy performs encryption.

After installing CTE 7.0.x or higher and setting up CTE-LDT on the hosts, you can change legacy GuardPoints to CTE-LDT GuardPoints. CTE-LDT GuardPoints provide the advantage of allowing users to access all files in the GuardPoints while encryption is occurring. There is no downtime for the user except for the time needed to apply the GuardPoint.

1. Write a new CTE-LDT policy that transforms data from the non-CTE-LDT/non-versioned key used in the existing GuardPoint, to an CTE-LDT versioned key. See "Creating CTE-LDT Policies" on page 35.

   The following example shows an CTE-LDT production policy that transforms data encrypted by a non-versioned key (AES256) to data encrypted with a versioned key (LDT).



2. Make sure there is no application activity within the GuardPoint.

   CAUTION
   This step is critical. Do not skip it. Make sure there is no application activity within the GuardPoint.

3. Click **Unguard** to unguard the GuardPoint.
4. Guard the directory again using the new CTE-LDT policy. Use the steps in "Creating a CTE-LDT GuardPoint for a Linux Directory or NFS Share" on page 48, but choose the policy that starts from the non-versioned/non-LDT key rather than a policy that starts from `clear_key`.

Rotating Encryption Keys (Rekey)

This following sections describe procedures related to encrypting an existing GuardPoint with a new key (also called rekeying).

> **Note**
> If the GuardPoint is shared among multiple hosts using NFS (Linux) or CIFS (Windows), key rotation will not start until the GuardPoint is only guarded on a single host.

> **Note**
> The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

Manual Key Rotation

If you need to rekey your GuardPoint prior to the expiration of the current key version, you can manually start a rekey process.

On Windows, you cannot rotate a key until all current data transformations that use the same key have completed, whether those transformations are an initial data transformation or a rekey. On Linux, you can rotate the key at any time. For details, see "Rotating Encryption Keys While a Rekey is in Progress (Relaunch)" on page 55.

1. To manually generate a new version of a key, open the DSM Management Console and choose **Keys > Agent Keys > Keys.**
2. Click the name of the key you want to rotate.

   Note the hex value shown next to **Key Hash** before and after you rotate the key. The hash value represents the current key version associated with the key. After rotating the key, CTE-LDT generates a new version of the key, and the hash value changes.
3. Click **Rotate**.
4. After clicking Rotate, a Rotate Key pop-up displays. Select the *Rotation Reason* and click **Rotate**. This initiates the rekey process on all GuardPoints using the key.
5. Check the status of the GuardPoint in the Rekey Status column of the GuardPoint tab. For example, just after clicking Rotate, the status might be “Rekeying 4%”.

   Once CTE-LDT transforms the data, you are finished.

Checking the Rekey Status

During a rekey, you can check the progress in the UI on the GuardPoint status page on the DSM. GuardPoint status is not relayed to the DSM in real time. A delay of several minutes before the DSM displays events on the host is likely. When the number of GuardPoints on the managed host is high, for example 100+, the delay in relaying GuardPoint status is due to delays in scheduling and the execution of CTE-LDT operations on the managed host for GuardPoints.

When the host and DSM are configured for one-way communication, the delay is longer because CTE-LDT statistics are sent to the DSM once per hour, or each time a GuardPoint enable or disable status changes.

To check when the DSM last received a status update from the host, look at the Last Status Update time stamp in the GuardPoint status dialog. For information about how to display this dialog, see "Obtaining Statistics in the DSM or CipherTrust Manager with GuardPoint Status" on page 102. For the most up-to-date statistics, inspect the host itself. To find out whether your host is configured for one-way communication, open the DSM Management Console, choose **Hosts > Hosts**, click the name of the host, and look for the FS Agent One Way Communication check box.

To see the progress of a rekey and the estimated completion time:

1. Open the DSM Management Console.
2. Choose **Hosts > Hosts**.

3. Click the name of the host you want to check.
4. Click the **GuardPoint** tab.
5. The Rekey Status column shows the current status. The status is one of the following:

Rekey Status	Description
Error (Windows only)	Runtime error occurred. Check for CTE-LDT-ALERT messages. Typically occurs when Low disk space occurs or a volume got dismounted.
Exited (Windows only)	Exited an operation like renaming folder. CTE-LDT will be restarted in the next ~20 secs or as soon as the operation completes. On restart, it will go into scan phase if file/folder deleted/renamed.
Incomplete (Windows only)	CTE-LDT did not finish. The administrator must intervene to remove the obstacle so the rekey can finish. You can run the `voradmin ldt skip` command to obtain more information.
In-Progress	CTE-LDT is transforming each file one by one.
Not Started	CTE-LDT has not started. It will go into this state when a GuardPoint is added or a key is rotated or changed.
Paused	CTE-LDT is paused on the GuardPoint, because of the QoS schedule or a direct request from administrator.
Queued	When a key rotation is in progress and another rotation is initiated.
Rekeyed	GuardPoint has been transformed to the latest key version.
Rekeying	GuardPoint transformation is in process of transforming the data with the latest key.
Scanning	Linux: CTE-LDT started, it is scanning the GuardPoint and counting bytes, files to transform in the GP before starting transformation. Required for statistics and estimation. Windows: CTE-LDT is analyzing files in the GuardPoint before initial transformation, before a rekey, or following an interrupted rekey. The scan is required for statistics and estimation. CTE-LDT can scan multiple times on completing a traversal of the guarded directory. The scanning status does not include a percent complete estimate.
Starting	GuardPoint transformation is ready to begin, but has not yet begun.
Suspended	When CTE-LDT is suspended by user or due to QoS schedule.
Unguard Required	Another CTE host is waiting for this CTE Linux or Windows host to disable this GuardPoint over NFS (CTE Linux) or CIFS (CTE Windows). As soon as this host disables the GuardPoint, the other CTE host will launch LDT on the GuardPoint.
Unknown	GuardPoint is not active on the host, so status could not be determined.
Waiting to Launch Rekey	CTE-LDTfor this NFS (Linux)/CIFS (Windows) GuardPointis ready to start on this host as soon as the GuardPointis disabled on all other CTE hosts that share this GuardPoint.

Obtaining Information About Keys Applied to Files

Key Report Option

In the following command, you can use the `report` option of the `voradmin` command to obtain information about all of the keys in use on the GuardPoint. The report lists all keys used in the GuardPoint. For each key, it gives the key name and key version number. It lists each unique key name and version combination only once, no matter how

many files use the key.

The following example shows three keys used in the GuardPoint `/oxf-fs1/gp1`:

```
# voradmin ldt key report /oxf-fs1/gp1
LDT_KEY1,1
LDT_KEY2,2
LDT_KEY3,5
```

For an overview of `voradmin ldt`, see "CTE-LDT Command-Line Administration: voradmin command" on page 92.

Key Map Option

In the `voradmin ldt key [report|map] <key_name> <guard path>` command, you can use the `map` option to obtain information about which files in a GuardPoint were transformed with a specific key, where:

- `<key_name>` is the name of the key.
- `<guard path>`is the path of the GuardPoint where the key was used.

For example, to view information about the key LDT_KEY2 in the GuardPoint /oxf-fs1/gp1, you would enter:

```
# voradmin ldt key map LDT_KEY2 /oxf-fs1/gp1
/oxf-fs1/gp1/file12345.dat10
/oxf-fs1/gp1/file12345.dat10
/oxf-fs1/gp1/file12345.dat10
/oxf-fs1/gp1/file12345.dat10
/oxf-fs1/gp1/file12345.dat10
```

Keys without a version number are used by files in an exclusion key rule or files that have yet to undergo initial key rotation. Use `voradmin ldt key map` in conjunction with `voradmin ldt attr get` to determine if a file using a key without a version number is part of an exclusion key rule or awaiting initial key rotation.

Showing GuardPoints During Rekey (Linux)

Use the following command to display a list of known CTE-LDT metadata stores and any associated GuardPoints currently undergoing transformation.

```
# voradmin ldt list all
MDS_1:  type=file, nguards=0, name=/::vorm:mds::
            Guard Table: version 1 nentries 0
MDS_2:  type=file, nguards=0, name=/disk3/::vorm:mds::
            Guard Table: version 1 nentries 0
MDS_3:  type=file, nguards=0, name=/disk4/::vorm:mds::
        Guard Table: version 1 nentries 0
```

For an overview of `voradmin ldt` commands, see "CTE-LDT Command-Line Administration: voradmin command" on page 92.

Suspending and Resuming Rekey and/or Scan Phase

The QoS schedule specifies certain time windows when CTE-LDT operations must be stopped temporarily. However, you can also suspend or resume CTE-LDT at the host/client level at any time.

To suspend CTE-LDT through the DSM Management Console:

1. Go to **Hosts** > **Hosts**.
2. Click the name of the host where you want to suspend or resume rekey.
3. Click the **GuardPoint** tab.
4. Click **Suspend/Resume Rekey** to suspend/resume CTE-LDT operations on the host.

Considerations

- Disabling, or unguarding, an CTE-LDT GuardPoint where a rekey is in progress automatically suspends the CTE-LDT rekey operation.
- You can also use the `voradmin suspend/resume` command in Windows Powershell to suspend and resume the rekey. However, if you use the `voradmin ldt suspend` command to pause the rekey process, the CTE-LDT suspended state is not retained when the system reboots. If you want to retain the state after reboot, suspend the rekey from the DSM or CipherTrust Manager.
- On Linux, you can pause CTE-LDT during the scan phase of CTE-LDT. When paused during scan, CTE-LDT suspends operations that traverse through the namespace of GuardPoints in the scan phase of transformation. Suspending CTE-LDT during scan stops file lookup operations of CTE-LDT. This eliminates performance impact to I/O intensive production workloads, such as file serving type workloads, that access large number of files.

Automatic Suspend and Resume of CTE-LDT Operations Due to Insufficient Disk Space

CTE-LDT requires adequate storage space headroom to perform CTE-LDT operations such as rekeying on GuardPoints. On Linux, if available storage space drops below the threshold required for CTE-LDT operations to continue, the CTE-LDT operations are automatically suspended. On Windows, CTE-LDT will be suspended if there is less than 3 GB free space in the GuardPoint. Once additional storage space is available in the file system, CTE-LDT operations automatically resume.

As available space approaches the threshold for automatic suspension, CTE-LDT sends an alert to the DSM or CipherTrust Manager to notify you that you should free up more storage space before CTE-LDT operations are suspended. The alert on the DSM or CipherTrust Manager is:

```
Low space on guard point [GuardPoint], increase free space or CTE-LDT will be suspended.
```

Behavior of Automatic Suspend and Resume of CTE-LDT Operations on ext4 File Systems

By default, ext4 file systems reserve a portion of the storage space for use only by privileged processes to prevent running out of storage space in file systems. In this situation, non-privileged processes are automatically blocked from writing to the file system until the free disk space level reaches the minimum threshold. As CTE-LDT operates in privileged mode, CTE-LDT operations continue without blockage even if the free disk space threshold is low. Because of this ext4 feature, CTE-LDT operations on ext4 file systems may not be suspended due to low available storage space, even when the `df` command reports storage is 100% allocated.

Rotating Encryption Keys While a Rekey is in Progress (Relaunch)

On Linux, if a key is rotated (either manually or automatically when a key version expires) while CTE-LDT is in progress on a GuardPoint, the key rotation is processed and queued, and the GuardPoint is marked for relaunch. Relaunch indicates the need to restart CTE-LDT after the current transformation completes. If the GuardPoint has been rekeyed and is flagged for relaunch, CTE-LDT launches as soon as the GuardPoint is enabled.

When this event occurs, the following message appears in the log file: "`LDT: Deferred key rotation on GuardPoint [GuardPoint] until after completion of current transformation.`"

When the new key rotation request is queued, files can be in one of three states: Undergoing rekey to a previous key, scheduled for rekey to a previous key, or rekeyed to a previous key.

- Files already undergoing transformation to a previous key when the new rekey request is queued are rekeyed to the key version already in progress.
- Files that start transformation after the new key rotation request is queued are rekeyed to the newest key version.
- Files that have already been rekeyed to a previous key remain in that state until all files undergoing rekey or scheduled for rekey have been processed. Once the current CTE-LDT process for the GuardPoint is complete, CTE-LDT automatically relaunches to transform any files that are not rekeyed to the latest key version. This includes any files that were rekeyed before the new key rotation request was queued or that were undergoing transformation when the new key request was queued.

For example:

```
# voradmin ldt attr get /oxf-fs1/gp1
LDT stats: version=2, rekey_status=rekeying,relaunch
     Number of times rekeyed:                1 time
     Rekey start time:                       2020/03/18 16:54:00
     Last rekey completion time:             2020/03/18 16:53:38
     Estimated rekey completion time:        0 days 0 hours 6 minutes
     Policy key version:                     344
     Data stats:
           total=9.8GB, rekeyed=0.0MB
           truncated=0.0MB, sparse=0.0MB
     File stats:
            total=1000, rekeyed=0, failed=0
            passed=0, skipped=0, created=0, removed=0, excluded=0
```

> **Note**
> Relaunch is supported on Linux only. It is not supported on Windows.

File System Operations

The following sections describe the file system operations that may require attention from the Administrator .

Renaming Files and Directories

On both Linux and Windows hosts, you can rename files during initial transformation, or rekey operations.

- On Linux, you can now, as of v7.1.1, rename directories in a GuardPoint during a rekey operation.
- On Windows with VTE version 6.1.0 or later, CTE-LDT stops if it is transforming the contents of a folder and a user attempts to rename/move that folder.

  You can change the stopping behavior of CTE-LDT using the configuration parameter `oxf_stop_on_rename`.

  Using the Registry Editor, or the Windows command line, add a registry entry in `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Vmmgmt\Parameters` for `oxf_stop_on_rename`.

Registry Name	Values	Comments
`oxf_stop_on_rename`	0 - Disabled	CTE-LDT **does not** stop if there is contention.
	1 - Enabled	CTE-LDT **stops** if there is contention. Default: Enabled

  For the target folder in rename operation, CTE-LDT is never stopped.

Renaming Directories on Linux

Prior to CTE-LDT v7.1.0, you could not rename a directory within a CTE-LDT GuardPoint, during the rekeying process. However, starting in CTE-LDT v7.1.1, users and applications can now rename directories within an CTE-LDT GuardPoint, during the rekeying process.

When a directory is renamed, CTE-LDT starts performing single file rekey jobs, on files in that renamed directory. A single file rekey job is a background process that rekeys a specific file, independent of the rekey process occurring on the entire GuardPoint. This process is similar to how files marked with a lazy rekey flag are rekeyed to the latest key version upon access to the marked file.

You can view rekey jobs in a GuardPoint using the command `voradmin ldt list all`. Rekey operations for files marked as lazy, as part of CipherTrust Intelligent Remediation, and for directory rename, are listed as separate single file rekey jobs. In the example below, the file `/ldt/renamed/file.0` is undergoing rekey as a single file rekey job, as the result of changing the pathname to the file.

```
# voradmin ldt list all

MDS_1:  type=file, nguards=9, name=/ldt/::vorm:mds::

        Guard Table: version 1 nentries 9

        Guard 0: type=GP, state=REKEYING DIRTY, flags=GP LOCKED, gp=/ldt

                File List: count 51

        Guard 1: type=FILE, state=REKEYING DIRTY, flags=FILE LOCKED, gp=/ldt/renamed_dir/file.0

                File List: count 1
```

CTE-LDT can run a maximum of eight concurrent file rekey jobs for directory renaming. This limit is shared across directory renames for all of the GuardPoints on a system.

Furthermore, while directory rename is in progress on a target GuardPoint, the GuardPoint will not transition into the rekeyed state, even if CTE-LDT has completed on the rest of the files in the GuardPoint. As soon as the single file rekey jobs associated with the GuardPoint complete, the GuardPoint transitions into a rekeyed state.

Caveats

The following caveats exist when renaming directories that are in GuardPoints:

1. A directory rename operation changes path names to an unknown number of files. As such, CTE-LDT cannot accurately estimate completion time on the target GuardPoint. In such a situation, the rekey progress reflected on the key manager may show that the GuardPoint is in 100% rekeyed status, and the status remains in the 100% rekeying status, until the files in the renamed directories are rekeyed to completion.
2. Directory rename operations also affect the accuracy of LDT statistics. Those files, in a renamed directory that are not rekeyed when the directory is renamed, are no longer accessible as part of rekeying the GuardPoint. However, those files will be rekeyed as part of rekeying the renamed directory. As the GuardPoint level rekey operations fail to find those non-rekeyed files, LDT stats update to reflect those files as 'passed' and their size is added to the 'truncated' stat. When those files are rekeyed, as part of the renaming directory rekey process, the 'rekeyed' status will be correctly updated.
3. The net effect of renaming a directory during a rekey will be reflected in an inaccurate 'passed' count, and the amount of data truncated for each file not rekeyed in the renamed directory.

Example

If there are 1024 files totaling 1 GB in the directory `/ldt/gp/dir`, within the GuardPoint `/ldt/gp` when rekey starts, the stats should appear as follows:

```
# ls -1 /ldt/gp/dir | wc -l

1024

# du -h /ldt/gp/dir

1.0G    /ldt/gp/dir

# voradmin ldt stats /ldt/gp

LDT stats on /ldt/gp: version=3, rekey_status=rekeying

        Number of times rekeyed:                3 times

        Rekey start time:                       2021/06/14 23:53:56

        Last rekey completion time:             2021/06/14 21:26:09

        Estimated rekey completion time:        N/A

        Data stats:

                total=13.8GB, rekeyed=726.0MB, truncated=0.0MB

        File stats:

                total=1043, rekeyed=0, failed=0

                passed=0, skipped=0, created=0, removed=0, excluded=0
```

Upon renaming of the directory `/ldt/gp/dir`, any files that have not yet been processed by a GuardPoint level rekey, is added to the **passed** and **truncated** values in the stats:

```
# voradmin ldt stats /ldt/gp

LDT stats on /ldt/gp: version=3, rekey_status=rekeying

        Number of times rekeyed:                3 times

        Rekey start time:                       2021/06/14 23:53:56
```

```
Last rekey completion time:             2021/06/14 21:26:09

Estimated rekey completion time:        N/A

Data stats:

        total=13.8GB, rekeyed=10.6GB, truncated=1.0GB

File stats:

        total=1043, rekeyed=21, failed=0

        passed=1024, skipped=0, created=0, removed=0, excluded=0
```

Considerations

Consider the following issues when renaming directories during a rekey:

- If a directory rename operation occurs during the scan phase of CTE-LDT, the GuardPoint will be marked for relaunch and CTE-LDT will be launched again at the completion of the current rekey cycle. This is because it is not possible to begin rekey jobs while CTE-LDT is in the scan phase.
- When CTE-LDT is suspended on a GuardPoint with a renamed directory, additional single file rekey jobs will not be launched until rekey resumes. However, ongoing single file rekey jobs continue until each single file rekey is completely rekeyed. Consequently, suspending CTE-LDT while single file rekey operations are in progress only suspends rekey operations at the GuardPoint level, while CTE-LDT operations on single rekey files continues without disruption. As a result, stopping CTE service, which requires suspending CTE-LDT and disabling GuardPoints, may be delayed until the files currently undergoing rekey, due to directory rename, are completed.
- Ensure that CTE-LDT is suspended at the GuardPoint level and there is no single rekey file in progress before starting operations, such as a backup, which requires suspending CTE-LDT operations. Failure to do so can result in unexpected problems.
- If files in a renamed directory are moved out of the renamed directory, those files are processed as separate rekey jobs. Those rekey jobs cannot be suspended until the rekey is completed.
- A system crash during a rekey for a directory rename, also affects how CTE-LDT performs a recovery. Normally, CTE-LDT requires that, after the crash, the recovery completes for all of the affected files in the GuardPoint, before resuming rekey operations. However, with a directory rename, it is possible for the CTE-LDT recovery to skip files that are in the renamed directory because the path name to those files has changed since the scan phase of the key rotation. Instead, recovery for these files is deferred until the next access by CTE-LDT, or a user application, when a GuardPoint is enabled.
- CTE-LDT recovery log messages are not logged to CTE-LDT recovery log files, if the recovery log messages become necessary in the event of a failed recovery attempt. In the event of an error, CTE-LDT logs error messages in syslog, blocks access to the file that could not be recovered, and marks the file in rekey error. Files in rekey error must be restored from a backup. Additionally, as files that cannot be found for recovery are now treated as being deferred (only when the directory rename occurs prior to the crash), any orphaned files that might have been moved to the lost& found directory, in the target file system, will not be discovered as orphaned files and consequently skipped by CTE-LDT. CTE-LDT cannot discern if the orphaned files are missing or were linked to a renamed directory.

Deleting a File

When a file is removed before it is rekeyed, the file is not included in the total number of files transformed on the GuardPoint **Status** page. Any discrepancy between total number of files to transform and those transformed is due to the removal of files from GuardPoints during CTE-LDT.

The `voradmin` command provides more detailed file level statistics related to rekey operations on the host. You can run `voradmin` to get file level statistics:

```
# voradmin ldt stats <guard_path>
```

File Handling (Windows Only)

It is critical that you understand how the CTE-LDT process handles read-only, binary (executable), NTFS encrypted and NTFS compressed files.

The CTE-LDT process is subjected to all of the File System policies and attributes set on the files. In some cases, this prevents CTE-LDT from encrypting a file. If users or applications are accessing files while CTE-LDT is in progress, CTE-LDT cannot change the attributes of the files and encrypt the file. It is critical that you understand how CTE-LDT handles various types of files:

- **NTFS Encryption and Compression**

  If NTFS encryption and compression is enabled on a file or folder, the CTE-LDT process cannot encrypt these files. To maintain the data coherency, CTE-LDT skips the encryption of the these files. These files display as “passthrough” files in the CTE-LDT statistics.

- **Read-Only Files**

  When CTE-LDT encounters read-only files, it rekeys the file by resetting the read-only attribute and then setting the attributes back again when the rekey completes. If a file is open, CTE-LDT skips this file.

  1. If the file is not opened, CTE-LDT changes the attributes of the file and stores the original attributes in the file metadata.
  2. CTE-LDT starts Rekey on this file.
  3. If a user requests to open a file for writing while rekey is in progress, access is denied. User can only open files for reading.
  4. CTE-LDT restores the attributes once rekey is done.

- **Executable Files**

  If an executable is running, or files are exclusively locked by the application, the CTE-LDT process cannot encrypt those files as it is unable to acquire the required locks on the files. CTE-LDT skips these files and changes to the INCOMPLETE state.

Enabling GuardPoints in Read-Only mounted file systems (Linux)

Access to a GuardPoint enabled in a read-only mounted file system is restricted to read operations unless the GuardPoint is on an NFS share. You cannot modify data in such file systems, therefore, you cannot perform CTE-LDT operations on GuardPoints in read-only file systems. For this reason, CTE-LDT automatically suspends operations when GuardPoints are enabled on read-only file systems. CTE-LDT ignores all attempts to resume CTE-LDT operations until the underlying file system is remounted with read/write access.

WARNINGS

- You must disable a GuardPoint before changing the read/write mount options of the underlying file system. After changing the mount options, you can re-enable the GuardPoint. CTE-LDT operations adapt to the read/write options of the underlying file system when you enable the GuardPoint. Changing the mount options while a GuardPoint is enabled is unsupported and may result in unexpected errors.
- CTE-LDT will reject any request to enable GuardPoints on an NFS share mounted in read-only mode. Access to read-only NFS GuardPoints is not supported.

Copying Files Into a GuardPoint

If you copy a file into a GuardPoint without an Apply Key rule, make sure that the file was previously copied from the same GuardPoint, or a GuardPoint protected with the same policy/versioned key. A copy operation, without an Apply Key rule, is the same as a backup or restore of a file from/to an CTE-LDT protected GuardPoint.

CTE enforces key rules of an CTE-LDT policy while a GuardPoint is enabled. CTE cannot enforce the key rules while a GuardPoint is disabled. Modifying or adding data/files inside a disabled GuardPoint is not only unsupported, but it also results in unrecoverable data corruption.

WARNING
Do not add new files or modify existing files inside CTE-LDT protected GuardPoints while the GuardPoint is not enabled. This results in unrecoverable data corruption and/or files that cannot be accessed when the GuardPoint is enabled.

Behavior of Hard Links Inside and Outside of GuardPoints (Windows)

When using hard links on Windows, all the hard links to a file must be within the boundary of a GuardPoint and must use the same key. The following scenarios provide additional details:

- If hard links to the same file are inside a GuardPoint and outside a GuardPoint, the effect on the file depends on what process accesses which hard link first. If the hard link within the GuardPoint is opened first, the file is transformed. If the hard link outside the GuardPoint is opened first, the file won't be transformed.
- If hard links to the same file exist in different GuardPoints with different keys, the file will be corrupted.
- If hard links to the same file exist in the same GuardPoint but with different keys, such as if folder-based rules are used, there will be a conflict in the key.

Excluding Files or Directories from Rekey

You can exclude files or directories from the initial transformation and subsequent rekeys with exclusion key rules. For example, you can exclude a subset of non-secret files from a larger set of files that are encrypted with an Live Data Transformation policy.

You set up exclusion key rules on a DSM version 6.3.0 and higher. For details, see the *DSM Administration Guide*.

Examples of Exclusion Key Rules

This section describes some examples of how to use an exclusion key rule.

Note
If you are using exclusion key rules with a GuardPoint on a local file system, you will need to be very careful if you ever need to copy the encrypted files to an NFS share directory and then re-protect them within a new CTE-LDT NFS GuardPoint. Copying the encrypted files from the local file system to the NFS share does *not* preserve the LDT attributes or and/or the IV attribute from the policy that was used to guard the local directory. Because these attributes are not present, guarding the previously-encrypted files within an NFS CTE-LDT GuardPoint will lead to data corruption. You will need to decrypt such files before you copy them to the NFS share and then re-encrypt them with the new NFS CTE-LDT policy.

Encrypt Files With Exclusion Property Using a Non-Versioned Key

The following exclusion key rule applies the non-versioned key, `Key_TextFiles`, to any *new* files that are created with a `*.txt` extension.

> **Note**
> Existing `*.txt` files in the GuardPoint during the initial CTE-LDT data transformation process are assumed to be already encrypted with the same key that you specify in the exclusion key rule. These existing files are *not* transformed during the initial encryption or during any subsequent rekeys. The key you specify in the exclusion key rule is applied to new `*.txt` files only.

In the Policy, do the following:

1. Add a Key Rule.
2. In the **Resource** field, select a resource set that specifies `*.txt` in the **File** field.
3. Select the **Exclusion Rule** check box. The DSM changes to show only a single **Key** field.
4. In the **Key** field, select the key `Key_TextFiles`.
5. Add any other key rules or exclusion key rules to the policy that you want. You must add at least one non-exclusion key rule that specifies the current encryption key and the versioned encryption key that you want to use with this Live Data Transformation policy.

   For example, you could add another exclusion key rule that specifies all **.doc** files should be encrypted with the key `Key_DocFiles`, or one that specifies all `.zip` files should be left unencrypted by specifying the key `clear_key`.

Exempt Excluded Files from Encryption (Set to clear_key)

The following exclusion key rule sets all files in the resource set `/oxf-fs1/gp1/Clear_Files_Folder` (Linux) or `\oxf-fs1\gp1\Clear_Files_Folder` (Windows) to `clear_key` (in other words, not encrypted). Files in other directories that match other key rules in the same policy may be encrypted. This could allow unrestricted access to the files in `/oxf-fs1/gp1/Clear_Files_Folder` (Linux) or `\oxf-fs1\gp1\Clear_Files_Folder` (Windows) while access may be restricted to files in parallel directories.

- Linux Exclusion Key Rule: The Resource Set **Directory** field should contain `/oxf-fs1/gp1/Clear_Files_Folder`, and the **Key** should be `clear_key`.
- Windows Exclusion Key Rule: The Resource Set **Directory** field should contain `\oxf-fs1\gp1\Clear_Files_Folder`, and the **Key** should be `clear_key`.

Requirements for Exclusion Key Rules

Keep in mind the following requirements when configuring exclusion key rules:

- Before adding an exclusion key rule to an existing policy, you must disable all GuardPoints protected with the policy. Log on to the DSM or CipherTrust Manager to disable the GuardPoint.
- You cannot choose a versioned key for the key in an exclusion key rule. Only non-versioned keys or `clear_key` (no encryption) are valid for exclusion key rules.
- Policies with exclusion key rules:
  - Can be added to Live Data Transformation policies created in the DSM 6.3 or higher.
  - Can be applied to GuardPoints on protected hosts running VTE version 6.2 or higher.

    Do *not* add an exclusion key rule to an existing Live Data Transformation policy if that policy is being used

to protect hosts running VTE version 6.1.x or earlier.

- All exclusion key rules must be *above* all CTE-LDT transformation key rules in the **Key Selection Rules** area in the policy.

Usage Notes and Limitations for Configuring Exclusion Key Rules

Keep in mind the information in the following sections when configuring an exclusion key rule.

Adding an Exclusion Key Rule to an Existing Policy with Versioned Keys (Linux)

When adding an exclusion key rule to an existing policy, the exclusion rule only applies to newly created files. Existing files that match the exclusion key rule remain encrypted with the same versioned key(s) specified in the non-exclusion key rule in the policy and will be rekeyed to the key in the exclusion key rule when the versioned key (s) rotates.

To force an existing file that matches an exclusion key rule to be transformed to the key in the exclusion key rule (non-versioned in this example), use one of the following methods:

- Rotate the versioned key specified in the policy to initiate rekey operations on the GuardPoint.
- Copy the existing file within the GuardPoint. The new file will be associated with the resource set in the exclusion key rule and will be encrypted with the non-versioned key. You can then delete the original file.

To perform a similar conversion on Windows, see "Changing a Folder or Files from Versioned to Non-Versioned Key (Windows)" on the facing page.

Adding an Exclusion Key Rule That is Part of an Active GuardPoint (Linux)

To edit and/or add an exclusion key rule to an CTE-LDT policy, all GuardPoints using the policy must first be disabled before the new key rule can be added. See “Add Key Selection Rules” in the “Policies” chapter of the *DSM Administration Guide*.

Changing an Exclusion Key Rule That is Part of an Active GuardPoint (Windows)

Changes that you make to an exclusion key rule that is part of an existing policy in an active GuardPoint do not take effect until the GuardPoint that the exclusion key rule is part of is disabled and enabled again.

Conflicting Keys as the Result of Rename Operations

Do not attempt to move or rename a file encrypted with a versioned key to a name associated with an exclusion key rule with `clear_key`. If you attempt such a move or rename, the original file is unaffected but following error is output on Linux systems and a log entry is created on the DSM:

```
<command name>: setting attribute ‘user.::secfs:xattr:’ for ‘user.::secfs:xattr:’:
Invalid argument
<command name>: failed to close ‘<filename>’: Invalid argument
```

No error is displayed on Windows systems. The target moved or renamed file is corrupted and should be deleted. The target file is also flagged with the `xattr_error` flag on Linux and `Rekey Status Excluded` on Windows. This flag prevents subsequent read/write access to the file. You can check the CTE-LDT attributes for the presence of this flag. See "About the Exclusion Attribute for Files Matching an Exclusion Key Rule" on page 66.

Also, a log entry is sent to the DSM or CipherTrust Manager on Linux systems when this occurs. For example, if you moved the versioned file `/gp/foo.txt` into the GuardPoint `/gp/subdir/foo.txt` with an exclusion key rule that excludes matching files with the clear_key, the following log message would be created on the DSM or CipherTrust Manager:

```
[CGA] [WARN ] [29261] [CGS3268W] LDT-ALERT: encrypted data detected in filename
[foo.txt] inode [35720037] in guard point [/gp] under clear exclusion key rule
```

Overlapping Exclusion Key Rules

Multiple exclusion key rules in the same policy may overlap each other. For example on Linux, if the non-versioned key Key_A is associated with resource `/oxf-fs1/gp1/Folder_Enc_With_KEY_A` and the non-versioned key Key_B is associated with resource `*.txt`, placement of the file `/oxf-fs1/gp/Folder_Enc_With_KEY_A/foo.txt` overlaps the two key rules. In such a case, the first rule in the policy is enforced on `/oxf-fs1/gp/Folder_Enc_With_KEY_A/foo.txt` when the file is created and in subsequent file access.

On Windows, if the non-versioned key Key_A is associated with resource `c:\oxf-fs1\gp1\Folder_Enc_With_KEY_A` and the non-versioned key Key_B is associated with resource `*.txt`, then they would overlap on the file `c:\oxf-fs1\gp\Folder_Enc_With_KEY_A\foo.txt`. In such a case, the first rule in the policy is enforced on `c:\oxf-fs1\gp\Folder_Enc_With_KEY_A\foo.txt` when the file is created and in subsequent file access.

Caution About Applications That Create Temporary Files (Windows)

Some applications on Windows create a temporary file version of the original file when you open and modify a file. This behavior can affect how you implement exclusion key rules.

If you have an exclusion key rule that uses a file extension to exclude files that may be opened and modified by such an application, exclude the temporary file name extension also. If you don't exclude the temporary file, the temporary file may be encrypted by another policy that matches the temporary file extension. Then the original file, which is copied from the temporary file, will be unreadable. Also keep in mind that other applications that may create temporary files with the same file extension and consider what policies should affect those temporary files.

This situation can happen with Microsoft Office files such as the `.docx` files used by Microsoft Word. When you open and modify a `.docx` file, Word creates a `.tmp` file version of that file. So for Microsoft Word you should add `*.tmp` to the exclusion key rule resource set if you add `*.docx`.

Rename Operations Crossing Key Rules (Linux)

On Linux, when a rename operation crosses key rules, the rename operation copies the source file to a new file using the new name and removes the original file. If the source file is flagged for exclusion key rule property, the target new file is disassociated with the exclusion key rule if the new name no longer matches the resource set associated with exclusion key rule. For more information, see "About the Exclusion Attribute for Files Matching an Exclusion Key Rule" on page 66.

Using Linked Files with Exclusion Key Rules (Linux)

On Linux, do not create multiple hard links to the same target file such that the pathname of each hard link is associated with a resource set of an exclusion key and the key rules have different keys. Accessing the file through the pathname of each hard link results in a different key to be applied to the file resulting in data corruption in the target file due to application of multiple keys to the same data.

If you must create hard links, be sure the pathnames of hard links and the target file are within the same resource set.

Changing a Folder or Files from Versioned to Non-Versioned Key (Windows)

Exclusion key rules provide a way to convert a subset of guarded files or the contents of a folder from being encrypted by a versioned key to being encrypted by a non-versioned key. After this conversion, the excluded files or

directory contents will be encrypted at the last version of the versioned key but the encryption keys for those items will not rotate to a new version when the keys for other non-excluded files are rotated. In other words, the excluded files or folder contents will be encrypted by a non-versioned key.

Follow these steps to change selected files or folder contents to a non-versioned key:

1. In the DSM or CipherTrust Manager, clone the versioned key that is used in the policy that you plan to edit. Cloning a key creates a non-versioned copy of the existing version of the versioned key.
2. In the DSM or CipherTrust Manager, disable the GuardPoint. Disabling the GuardPoint is required before modifying a policy applied in that GuardPoint.
3. Configure one or more resource sets to match the files that you want to exclude. For example, the resource set `star-dot-text` could contain `*.txt` and the resource set `sales-folder` could contain `\sales\*`.
4. Add one or more exclusion key rules to convert matching files to a non-versioned key. In the following example, files matching `*.txt` and files in the `sales` folder (as defined in resource sets) will be converted from the current version of the versioned key `AES256_versioned` to `AES256_clone`, assuming `AES256_clone` is a clone of `AES256_versioned`.

Order	Resource	Current Key	Transformation Key	Exclusion Rule
1	star-dot-txt	AES256_clone	AES256_clone	Y
2	sales-folder	AES256_clone	AES256_clone	Y
3		clear_key	AES256_versioned	N

   Exclusion key rules must be ordered before other rules.
5. On the command line, run the following command to remove the CTE-LDT metadata from the files that you want to convert from versioned key to non-versioned key encryption:

   ```
   voradmin ldt attr delete <path_to_files>
   ```

   To recursively delete the CTE-LDT metadata from all files matching a pattern in all subfolders, use the following form, including `*` as a wildcard where needed:

   ```
   voradmin ldt attr delete <path_to_files> -filter <filename>.<extension>
   ```

   > **Note:** Be careful when deleting the CTE-LDT metadata from files. If you delete the metadata from a file that does not match an exclusion key rule policy, the file will be unreadable after the next rekey.

   Given the example exclusion key rule described in step 4, you would need to run this command on all files with the extension `.txt` (first example below) and on the files in `\sales` (second example below).

   ```
   voradmin ldt attr delete c:\gp1 -filter *.txt
   voradmin ldt attr delete c:\gp1\sales
   ```

   For more information about using `voradmin` on CTE-LDT metadata, run `voradmin ldt attr /?` on the command line.
6. In the DSM or CipherTrust Manager, re-enable the GuardPoint that includes the new exclusion key rule.
7. In the DSM or CipherTrust Manager, rotate the key for the policy that includes the new exclusion key rule.

Using the example exclusion key rule in step 4, after completing this procedure, files matching `*.txt` and files in the `sales` folder will have the exclusion attribute set and will be excluded from rekeying (see "About the Exclusion Attribute for Files Matching an Exclusion Key Rule" on the facing page). Files not matching the exclusion key rule will be rekeyed to the next version of the `AES256_versioned` key.

To perform a similar conversion on Linux, see "Adding an Exclusion Key Rule to an Existing Policy with Versioned Keys (Linux)" on page 63.

About the Exclusion Attribute for Files Matching an Exclusion Key Rule

Files matching an exclusion key rule have the status `rekey_excluded` in the CTE-LDT attribute. For more information about CTE-LDT attributes, see "CTE-LDT Metadata in Extended Attributes" on page 72. To check for the exclusion attribute on a file, see "Determining if a File is Included in an Exclusion Key Rule" below.

The Exclusion Attribute is Persistent

Exclusion from rekey is a persistent property. A file excluded from rekey is not rekeyed regardless of changes that may seem to disassociate the file from the exclusion key rule. For example, if you rename a file to a new name within the same GuardPoint that no longer matches the exclusion key rule that the original name matched, the file with the new name retains the encryption type (non-versioned key or clear_key) of the exclusion key rule. To remove the exclusion attribute from a file you must copy the file rather than move or rename it (see "Removing the Exclusion Attribute From a File in a Local Directory GuardPoint" below).

Determining if a File is Included in an Exclusion Key Rule

Use the `voradmin ldt attr get <path to file>` command to check whether a file is associated with an exclusion key rule.

- On Linux, an excluded file will include the status `rekey_excluded` in the `voradmin` output. For example:

  ```
  # voradmin ldt attr get /oxf-fs1/gp1/foo.txt
  CTE-LDT attributes: rekeyed_size=0, rekey_status=rekey_excluded
  ```

- On Windows, an excluded file will include the attribute `Rekey Status Excluded` in the `voradmin` output. For example:

  ```
  C:\> voradmin ldt attr get c:\gp1\foo.txt
  CTE-LDT attributes:
  Rekey Status Excluded
  Initial Rekeyed Size 9 Bytes
  Data Transformed 0 Bytes

  Key:
  Current Key Name/Version (Clear_Key)
  New Key Name/Version (Clear_Key)
  ```

Removing the Exclusion Attribute From a File in a Local Directory GuardPoint

To disassociate a file from an exclusion key rule, you must copy the file to a new file not associated with the resource set of the same or another exclusion key rule as the source file. You can then remove the original file. The new file is created under the default key rule of whatever policy applies to the new file. Moving or renaming a file does not disassociate a file from an exclusion key rule (see "The Exclusion Attribute is Persistent" above).

For example, assume the following exclusion key rule where Key_TextFiles is a non-versioned key:

Exclusion Key Rule: Resource set =`*.txt`, Key = `Key_TextFiles`

If you copy the file `test1.txt` to `test1.foo`, `test1.foo` is created with whatever key is specified in the policy that matches the new file. The key for the new file could be a non-versioned key, versioned key, clear_key, or, no key at all if the new file is outside of a GuardPoint. The original file `test1.txt` remains unchanged and encrypted with the Key_TextFiles non-versioned key because the file remains in the exclusion key rule.

Removing the Exclusion Attribute From a File in an NFS Share GuardPoint

For a CTE-LDT NFS share GuardPoint, in order to disassociate a file from an exclusion key rule you need to migrate from the existing LDT policy to a new LDT policy that does not contain the exclusion key rules and that encrypts the previously-excluded files using the same encryption key as the the rest of the files in the GuardPoint.

For example, if the existing LDT NFS GuardPoint `/mnt/HR-Data/` has the following exclusion key rules:

- Resource set =`*.log`, Key = `clear_key`
- Resource set =`*.txt`, Key = `Key_TextFiles`

You would do the following to remove all exclusion key rules:

1. Stop all applications or clients from accessing the LDT NFS GuardPoint `/mnt/HR-Data/`.
2. In the DSM, unguard the LDT NFS GuardPoint `/mnt/HR-Data/`.
3. Create a new policy that defines the same encryption key for all files without any exclusions. In this example, you would specify:
   a. Key Rule 1: Resource set =`*.log`, Current Key = `clear_key`, Transform key= `ldt_key2`
   b. Key rule 2: Resource set =`*.txt`, Current Key = `Key_TextFiles`, Transform key=`ldt_key2`
   c. Key rule3: Current Key= `clear_key`, Transform key=`ldt_key2`
4. On the host, remove the embedded attributes on the existing files in the NFS directory.

   ```
   # voradmin ldt attr delete /mnt/HR-Data/
   ```

   Make sure you verify that this command has completed successfully for all files in the GuardPoint before you continue with this procedure. You must remove the embedded attributes from the files before you remove the embedded attributes from the GuardPoint itself.
5. On the host, after `voradmin ldt attr delete` has completed, remove the embedded attributes on the existing NFS directory.

   ```
   # voradmin ldt rmldt /mnt/HR-Data/
   ```
6. On the DSM, create a GuardPoint for `/mnt/HR-Data/` using the newly created policy that has no exclusion key rules.
   - For an auto-guarded GuardPoint, the NFS guarded directory will automatically go through data transformation once the CTE Agent receives the policy push from the DSM.
   - For a manual GuardPoint, enable the GuardPoint using `secfsd -guard <gp>` to begin data transformation to convert all existing exclusion files to the new `ldt_key2`.

     ```
     # secfsd -guard /mnt/HR-Data/
     ```
7. Verify that none of the files in the CTE-LDT NFS GuardPoint are being excluded using the `voradmin ldt attr get` command.

   For example, let's say there is an excluded file called `example-file.txt` in the NFS share `/mnt/HR-Data/`.

   Before the procedure, when the file is still excluded:

   ```
   # voradmin ldt attr get /mnt/HR-Data/example-file.txt
   LDT attributes: rekeyed_size=100003840, rekey_status=rekey_excluded
           Key:    name=clear_key, version=none
   ```

   After the procedure, when the file is no longer excluded:

   ```
   # voradmin ldt attr get /mnt/HR-Data/example-file.txt
   LDT attributes: rekeyed_size=100003840, rekey_status=none
           Key:    name=ldt-key2, version=5
   ```

Rename and Restore Operations (Linux)

The effect of rename or backup/restore operations involving files associated or not associated with exclusion key rules is mixed and somewhat confusing. In some cases, the operations to restore or rename cause conflicts between the key associated with the source file and the key associated with the target file. For example, the result of a rename operation involving a source file not associated with an exclusion key rule and target file name associated with a resource set of an exclusion key rule is different from the result of the same operation when target file name is associated with the resource set of an exclusion key rule with clear_key.

> **Note**
> Be sure to review the effect of the operations below and avoid administrative operations that cross associations of files across multiple resource sets with conflicting key rules.

Backup/Restore

The table below illustrates the status and the key effect of restore operations involving mixed keys associated with source files from backup image and existing target files inside GuardPoint.

Key Applied to File in Backup	Key Applied to existing file in GuardPoint	Key Effect of Restore Operation on Existing file
Versioned	Versioned	Versioned
Versioned	Exclusion key	xattr_error (failed)
Versioned	Exclusion clear key	xattr_error (failed)
Exclusion key	Versioned	Exclusion key
Exclusion key	Exclusion key	Exclusion key
Exclusion key	Exclusion clear key	xattr_error (failed)
Exclusion clear key	Versioned	Exclusion clear key
Exclusion clear key	Exclusion key	xattr_error (failed)
Exclusion clear key	Exclusion clear key	Exclusion clear key

The table below illustrates the status and the key effect of restore operations involving mixed keys associated with source files from backup image and new target files not present inside GuardPoint.

Key Applied to File in Backup	Key Applied to existing file in GuardPoint	Key Effect of Restore Operation on Existing file
Versioned	Versioned	Versioned
Versioned	Exclusion key	xattr_error (failed)
Versioned	Exclusion clear key	xattr_error (failed)
Exclusion key	Versioned	Exclusion key
Exclusion key	Exclusion key	Exclusion key
Exclusion key	Exclusion clear key	xattr_error (failed)
Exclusion clear key	Versioned	Exclusion clear key

Key Applied to File in Backup	Key Applied to existing file in GuardPoint	Key Effect of Restore Operation on Existing file
Exclusion clear key	Exclusion key	xattr_error (failed)
Exclusion clear key	Exclusion clear key	Exclusion clear key

Rename Operation

The table below illustrates the status and the key effect of rename operations for different combinations of versioned, exclusion key, and exclusion clear key.

Key Applied to File in Backup	Key Applied to existing file in GuardPoint	Key Effect of Restore Operation on Existing file
Versioned	Versioned	Versioned
Versioned	Exclusion key	Exclusion key
Versioned	Exclusion clear key	xattr_error (failed)
Exclusion key	Versioned	Exclusion key
Exclusion key	Exclusion key	Exclusion key
Exclusion key	Exclusion clear key	Exclusion key
Exclusion clear key	Versioned	Exclusion clear key
Exclusion clear key	Exclusion key	Exclusion clear key
Exclusion clear key	Exclusion clear key	Exclusion clear key

Listing All Files Included in an Exclusion Key Rule (Linux)

Determining the files that match an exclusion key rule involves two steps. You list all the keys in a GuardPoint, choose the key that you're interested in, and then run a command to list the files that match that key. This process works for both standard key rules and exclusion key rules.

1. Decide on the GuardPoint that you want to check for excluded files. For example, for the GuardPoint `/oxf-fs1/gp1`, type the following on the command line:

   ```
   # voradmin ldt key report /oxf-fs1/gp1
   LDT_KEY1,1
   LDT_KEY2,2
   LDT_KEY3,5
   NON_VERSIONED_KEY
   ```

   The number after the comma is the key version number. See "Key Report Option" on page 53 for more information about the `voradmin` key report.

2. From the key report output, choose the key rule for which you want to list matching files. For example, to see the files associated with the NON_VERSIONED_KEY in the previous step, you would type:

```
# voradmin ldt key map NON_VERSIONED_KEY /oxf-fs1/gp1
/oxf-fs1/gp1/file1.dat10
/oxf-fs1/gp1/file2.dat10
/oxf-fs1/gp1/file3.dat10
/oxf-fs1/gp1/file4.dat10
/oxf-fs1/gp1/file5.dat10
```

   See "Key Map Option" on page 54 for more information about the `voradmin` key map report.

Listing All Files Included in an Exclusion Key Rule (Windows)

Determining the files that match an exclusion key rule involves two steps. You list all the keys in a GuardPoint, choose the key that you're interested in, and then run a command to list the files that match that key. This process works for both standard key rules and exclusion key rules.

1. Decide on the GuardPoint that you want to check for excluded files. For example, for the GuardPoint `c:\gp1`, type the following on the command line:

```
C:\> voradmin ldt key report c:\gp1
Keys used for GP, c:\gp1 :
clear_key,0
CS1-CTE-LDT-AES256,11
```

   The number after the comma is the key version number. The clear_key key is not versioned, so the version number is 0.

2. From the key report output, choose the key rule for which you want to list matching files. For example, to show the files associated with the clear_key listed above, you would type:

```
C:\> voradmin ldt key map c:\gp1 clear_key,0
Files rekey with key (clear_key,0)-
c:\gp1\file1.txt
c:\gp1\file2.txt
c:\gp1\file3.txt
c:\gp1\file4.txt
c:\gp1\file5.txt
```

Using CTE-LDT with SAP HANA Fibre Channel Systems (Linux Only)

You must add additional CTE commands to the HANA administrator entry. To do so, edit the `/etc/sudoers` with a text editor and add entries for `/usr/bin/voradmin` and `/usr/bin/vmsec`: For example:

```
# hanadm ALL=NOPASSWD:
/usr/bin/secfsd,/usr/bin/voradmin,/usr/bin/vmsec,/sbin/multipath,/sbin/multipathd,/etc/i
nit.d/multipathd,/usr/bin/sg_
persist,/bin/mount,/bin/umount,/bin/kill,/usr/bin/lsof,/sbin/vgchange,/sbin/vgscan
```

If you are using an ext3 file system, you must mount it with extended attributes. To do so, edit the storage section of the `global.ini` file using a text editor and add the following lines:

```
partition_*_data__mountOptions = -o user_xattr
partition_*_log__mountOptions = -o user_xattr
```

Note
See the *CTE Agent for Linux Advanced Configuration and Integration Guide* or the *CTE Agent for Windows Advanced Configuration and Integration Guide* for more information about SAP HANA.

Chapter 5: CTE-LDT Administration

This chapter contains the following topics:

CTE-LDT Metadata in Extended Attributes

An *extended attribute* is a name/value pair permanently associated with a file or directory stored in a file system. CipherTrust Transparent Encryption (CTE) creates and maintains its own user extended attributes on CTE-LDT GuardPoint directories and files. The extended attributes are used to store metadata related to each file or directory that is protected using an CTE-LDT policy.

On Linux, CTE-LDT sets extended attributes on GuardPoint directories in local file systems, and creates CTE protected file for each GuardPoint over NFS shares. (For more information, see "CTE-LDT Private Space Directory for NFS Shares" on page 77.)

> **Note**
> The single node CTE-LDT for NFS/CIFS feature is a preview with DSM. Multiple node CTE-LDT CIFS/NFS support will only be available on CipherTrust Manager. Customers will need to start fresh on CipherTrust Manager, migration will not be possible.

The CTE-LDT attribute of an CTE-LDT GuardPoint stores the following metadata:

- Current key version.
- Rekey status.
- Rekey start and end times.
- Estimated completion time.
- Total amount of data transformed.
- Total number of files transformed.
- Current key signature and applied key signature.

On both Linux and Windows, CTE-LDT sets extended attributes on files in local file systems, and embeds the LDT metadata for each file in the beginning of files in GuardPoints over NFS/CIFS shares. The CTE-LDT attribute of a file stores the following metadata:

- Name of the current key.
- Name of the versioned key.
- Version number of the versioned key.
- CTE-LDT rekey status of the file.

In most cases, the current and new key names are the same. The exception is during initial transformation from a legacy policy to an CTE-LDT policy, when the file has been encrypted with the current key and is being transformed to the current version of the transformation key.

> **Note**
> Before you set up a GuardPoint for CTE-LDT, ensure that there is sufficient disk space available in your file system for CTE-LDT metadata. The amount of disk space you need depends on the number of files in your GuardPoint. For more information about the disk space requirement, see "Planning for CTE-LDT Attribute Storage" on page 78.

The state of a file changes during CTE-LDT operations. The extended attributes are continually updated to reflect the current file status, which falls into one of the following categories:

- Rekeyed to the current version of the key.
- Rekeyed to the previous version of the key, or the initial key state (before the first CTE-LDT rekey has been performed).
- Partially rekeyed, where some regions of the file are rekeyed to the new key version and other regions are still keyed to the previous key version or the initial key.

LDT Metadata Management Over NFS/CIFS Shares

CTE-LDT manages LDT metadata as extended attribute in local file systems for each file and GuardPoint directory. Because the NFS/CIFS protocol does not support extended attributes, CTE-LDT embeds LDT metadata in files over NFS/CIFS shares during the initial data transformation as each file is transformed. The size of the LDT metadata is 4096 bytes, so the size of each file in the NFS/CIFS share is larger by 4096 bytes. The presence of LDT metadata and larger file sizes are not visible to users and applications as long as the GuardPoint remains enabled. The same file stored in a CTE-LDT protected GuardPoint, over NFS/CIFS and a non-protected directory, is identical to other users and applications despite the presence of the embedded LDT attribute and the additional 4096 bytes in the size of the file.

As the Linux example below illustrates, the presence of the attributes embedded during the initial transformation of the file is invisible to users and applications. Note that the size of the file after transforming the file appears unchanged until the GuardPoint is disabled.

```
# ls /nfs-oxf-fs1/gp1
# cp /etc/hosts /nfs-oxf-fs1/gp2
# ls -l /nfs-oxf-fs1/gp2/hosts
-rw-r--r--. 1 root root 241 Jan  3 16:14 hosts
# secfsd -guard /nfs-oxf-fs1/gp2
secfsd: Guard point initialization in progress
# voradmin ldt attr get /nfs-oxf-fs1/gp2/hosts
LDT attributes: rekeyed_size=0, rekey_status=none
        Key:    clear_key
# ls -l /nfs-oxf-fs1/gp2/hosts
```

```
-rw-r--r--. 1 root root 241 Jan  3 16:15 /nfs-oxf-fs1/gp2/hosts
# secfsd -unguard /nfs-oxf-fs1/gp2
secfsd: Path is not guarded
# ls -l /nfs-oxf-fs1/gp2/hosts
-rw-r--r--. 1 root root 4337 Jan  3 16:15 /nfs-oxf-fs1/gp2/hosts
# voradmin ldt attr get /nfs-oxf-fs1/gp2/hosts
LDT attributes: rekeyed_size=4096, rekey_status=none
        Key:    name=LDTNFS_KEY_1, version=1482
```

As noted, CTE-LDT also manages the LDT attribute for each GuardPoint directory as an extended attribute. For GuardPoints over NFS/CIFS shares, CTE-LDT stores the LDT metadata for each GuardPoint in the LDT Attribute File associated with the directory. In general, there are multiple metadata files that CTE-LDT manages for GuardPoints over NFS. Those metadata files are stored in the LDT Private Space Directory inside each GuardPoint directory. The directory name of LDT Private Space is `vorm_ldtprivspace`. The directory is created inside the GuardPoint directory at the time of initial transformation. The LDT metadata file for GuardPoint is also created in `vorm_ldtprivspace` directory of GuardPoint at the time of initial transformation. The name of the file is `::vorm:ldtxattr::`, similar to the MDS file which is also protected against user modification or deletion. The size of the LDT metadata file is 4096 bytes. For example:

```
# ls -l /nfs-oxf-fs1/gp1/vorm_ldtprivspace/
total 4
-rwxr-xr-x. 1 root root 4096 Jan  4 12:13 ::vorm:ldtxattr::
```

WARNING
The LDT Attribute file is protected, and it can only be manually removed using `voradmin ldt rmldt delete`. For details, see "Deleting CTE-LDT Metadata (Linux)" on page 99.

Listing Extended Attributes

You can list extended attributes of files in local file systems by using native operating system commands, or system calls. As part of GuardPoint administration, CTE can modify or delete extended attributes.

> **Note**
> This functionality is only available for local file systems. It is *not* supported for files in NFS Share GuardPoints.

In Linux, CTE-LDT attributes are set on GuardPoint directories and regular files in GuardPoint directories protected with CTE-LDT policies. The CTE extended attribute name is `::secfs:xattr:` on local file systems.

The following examples use the native Linux operating system command `attr` to display the CTE-LDT attribute for the GuardPoint `/oxf-fs1/gp1` and the file `/oxf-fs1/gp1/File_1.txt`. In this example, `/oxf-fs1/gp1` is a local file system.

Example Getting File Attributes

```
# attr -l /oxf-fs1/gp1/File_1.txt
Attribute "::secfs:xattr:" has a 1044 byte value for /oxf-fs1/gp1/File_1.txt
Attribute "selinux" has a 37 byte value for /oxf-fs1/gp1/File_1.txt
```

Example Getting GuardPoint Attributes

```
# attr -l /oxf-fs1/gp1
Attribute "::secfs:xattr:" has a 1044 byte value for /oxf-fs1/gp1
Attribute "selinux" has a 37 byte value for /oxf-fs1/gp1
```

Example of `voradmin ldt attr get` for Linux File Attributes

In the following example, the file `/oxf-fs1/gp1/File_1.txt` has the same name for current and new keys at the same key version. In the following example, if the versioned key LDT_KEY is at version 755, the file is rekeyed to the latest key version under the CTE-LDT policy.

```
# voradmin ldt attr get /oxf-fs1/gp1/File_1.txt
CTE-LDT attributes: rekeyed_size=4096, rekey_status=none
 Key:    name=LDT_KEY, version=755
```

Example of `voradmin ldt attr get` for Linux GuardPoint Attributes

The following is example of an CTE-LDT attribute on a GuardPoint directory on Linux:

```
# voradmin ldt attr get /oxf-fs1/gp1
LDT stats: version=1, rekey_status=rekeying
       Number of times rekeyed:         3 times
       Rekey start time:                2018/08/04 16:24:45
       Last rekey completion time:      2018/07/04 16:24:04
       Estimated rekey completion time: N/A
       Policy key version:              2043
       Data stats:
          total=3.3GB, rekeyed=1.5GB, truncated=0.0MB
       File stats:
          total=4307, rekeyed=1181,
          passed=2, skipped=0, created=0, removed=0
```

Example of `voradmin ldt attr get` for Linux NFS Share GuardPoint Attributes

The following example shows how to use the `voradmin ldt attr get` command to view the CTR-LDT attribute on GuardPoint directories over NFS shares:

```
# secfsd -unguard /nfs-oxf-fs1/gp2
secfsd: Path is not guarded
# voradmin ldt attr get /nfs-oxf-fs1/gp2
LDT stats: version=3, rekey_status=rekeyed
        Number of times rekeyed:                1 time
        Rekey start time:                       2021/01/04 08:19:02
        Last rekey completion time:             2021/01/04 08:19:03
        Estimated rekey completion time:        N/A
        Policy key version:                     0
        Policy ID:
                23785
        Data stats:
                total=0.0MB, rekeyed=0.0MB
                truncated=0.0MB, sparse=0.0MB
        File stats:
                total=3, rekeyed=1, failed=0
                passed=0, skipped=0, created=0, removed=0, excluded=0
```

Example of `voradmin ldt attr get` for Windows GuardPoint Attributes

The attribute for the GuardPoint `c:\GP 1` contains the status (rekeyed) and statistics specific to the GuardPoint and CTE-LDT. Following is sample output of voradmin command on Windows for statistics on a file:

```
C:\> voradmin ldt attr get c:\GP\Test.txt
LDT attributes:
    Rekey Status           Rekeyed
```

```
    Initial Rekeyed Size   10 Bytes
Key:
    Key Name/Version       (LDT_KEY, 28)
```

The attribute for GuardPoint C:\GP contains the status (rekeyed) and statistics specific to the GuardPoint and CTE-LDT:

```
C:\> voradmin ldt attr get c:\gp\
LDT Stats
------------------------------
    Rekey Status                                LDT_ST_REKEYED
    Last rekey completion time        10/2/2017 4:26:50
    Rekey Start time                            10/2/2017 4:26:17
    Estimated rekey completion time   000:00:00

File Stats:
    Total      444
    Rekeyed    444
    Skipped    0
    Errored    0
    Passed     0
    Removed    0

Data Stats:
    Total      11 GB (12649143417 Bytes)
    Rekeyed    11 GB (12649143417 Bytes)
    Truncated  0 Bytes
```

Example of `voradmin ldt attr get` for Windows CIFS Share GuardPoint Attributes

The following example shows how to get the CTE-LDT attributes for the CIFS GuardPoint `\\myhost\share\HR-Files\`.

```
C:\>voradmin ldt attr get \\myhost\share\HR-Files\

Live Data Transformation Stats
------------------------------

    Rekey Status                      LDT_ST_REKEYED
    Last rekey completion time        2/24/2021 13:42:40
    Rekey Start time                  2/24/2021 13:36:47
    Estimated rekey completion time   000:00:00

File Stats:
    Total      19087
    Rekeyed    19087
    Skipped    0
    Errored    0
    Passed     0
    Removed    0
    Excluded   0

Data Stats:
    Total      1 GB (1083187108 Bytes)
    Rekeyed    958 MB (1005006756 Bytes)
    Truncated  0 Bytes
```

Example of `voradmin ldt attr get` for Windows CIFS Share File Attributes

The following example shows how to get the CTE-LDT attributes for the file `employees.doc` on the CIFS share `\\myhost\share\HR-Files\`.

```
C:\>voradmin ldt attr get \\myhost\share\HR-Files\employees.doc

LDT attributes:
    Rekey Status           Rekeyed
    Initial Rekeyed Size   0 Bytes

Key:
    Key Name/Version       (AES_256_LDTKey_CBC,15)
```

MDS File (Linux)

In addition to CTE-LDT attributes, the CTE-LDT process on Linux requires persistent storage for additional metadata related to encrypting, or rekeying, files in GuardPoints. CTE-LDT allocates the storage space as soon as the CTE-LDT process starts on a GuardPoint. It maintains this storage space during the entire transformation process, until the GuardPoint is completely transformed.

Storage for this metadata is allocated and managed in a special file, called the MDS (metadata store) file. The MDS file resides inside a GuardPoint directory so each GuardPoint has its own MDS file.

The MDS file is a CTE protected file with the name `::vorm:mds::`. For example:

```
# ls -l /oxf-fs1/gp1/::vorm:mds::
-rwxr-xr-x. 1 root  root 31754474496 Dec  8 09:09 /oxf-fs1/::vorm:mds::
# du -B 1024 /oxf-fs1/gp1/::vorm:mds::
25056   /oxf-fs1/::vorm:mds::
```

As shown above, the MDS file is sparse. In the example, the file size is approximately 30GB, however the file is allocated with approximately 25MB of disk storage. CTE-LDT automatically creates the MDS file the first time the CTE-LDT process starts on any GuardPoint in the file system namespace. It populates the MDS file with all of the metadata for the GuardPoint at the beginning of the CTE-LDT process. Disk space allocated to the MDS file is freed and the MDS file in the GuardPoint directory is removed when the CTE-LDT process completes on the GuardPoint.

> **WARNING**
> **The MDS file is protected. You cannot remove it unless the administrator runs the voradmin command to manually remove the MDS file once it is no longer needed. See "Deleting CTE-LDT Metadata (Linux)" on page 99 for more information.**

CTE-LDT automatically allocates and deallocates disk space for the MDS file as part of the CTE-LDT process. Deallocation of disk space for a GuardPoint does not change the MDS file size, although it frees the disk blocks. MDS files are sparse and very large in size. The MDS file is automatically removed from GuardPoints when the files have been successfully rekeyed.

CTE-LDT Private Space Directory for NFS Shares

CTE-LDT also manages the LDT attribute for each GuardPoint directory as an extended attribute. For GuardPoints over NFS shares, CTE-LDT stores the LDT metadata for each GuardPoint in the LDT Attribute File under the GuardPoint directory. In general, there are multiple metadata files that CTE-LDT manages for GuardPoints over NFS. Those metadata files are stored in the LDT Private Space Directory inside each GuardPoint directory. The directory name of LDT Private Space is `vorm_ldtprivspace`. The directory is created inside the GuardPoint directory at the time of initial transformation.

LDT Metadata File

The LDT metadata file holds the same metadata information that CTE-LDT maintains for GuardPoint directory in local file systems. For GuardPoints over NFS, the metadata file is created in the `vorm_ldtprivspace` directory under the GuardPoint directory pathname at the time of initial transformation. The name of the metadata file is `::vorm:ldtxattr::`, similar to an MDS file. Like MDS files, the metadata file are fully protected against user modification or deletion. The size of LDT metadata file is 4096 bytes. For example:

```
# ls -l /nfs-oxf-fs1/gp1/vorm_ldtprivspace/
total 4
-rwxr-xr-x. 1 root root 4096 Jan  4 12:13 ::vorm:ldtxattr::
```

WARNING
The Metadata file is protected. You cannot remove it unless the administrator runs the voradmin command to manually remove the metadata file once it is no longer needed. See "Deleting CTE-LDT Metadata (Linux)" on page 99 for more information.

CTE-LDT Host Tag Files

CTE maintains the list of which CTE clients have enabled a GuardPoint over NFS through *tag files*. A tag file is an empty file that CTE creates in the LDT Private Space directory of the GuardPoint when CTE enables the GuardPoint. Each CTE host creates its own tag file. If multiple tag files appear in the LDT Private Space directory, that means that the GuardPoint is enabled on multiple CTE hosts.

The tag file name includes the host name of the CTE protected host on which the GuardPoint is enabled. For example, if a GuardPoint is enabled on `host1`, the tag file name could be `tag_host1.i.mydomain.com`.

Each time a GuardPoint is enabled on a client, CTE adds a new tag file for that client the LDT Private Space directory. Similarly, each time a GuardPoint is disabled on a client, CTE removes the corresponding tag file from the LDT Private Space directory.

Note
Because these tag files are required, CTE requires read/write access to the NFS share. Read-only access for an NFS GuardPoint is *not* supported.

In the following example, the hosts `host1` and `host2` have the GuardPoint `/nfs-oxf-fs1/gp1` enabled:

```
# ls -l /nfs-oxf-fs1/gp1/vorm_ldtprivspace/
total 4
-rwx------. 1 root root 0 Jan 4 20:26 tag_host1.i.mydomain.com
-rwx------. 1 root root 0 Jan 4 20:25 tag_host2.i.mydomain.com
-rwxr-xr-x. 1 root root 4096 Jan 4 12:13 ::vorm:ldtxattr::
```

Planning for CTE-LDT Attribute Storage

Before a GuardPoint is enabled for CTE-LDT, make sure that there is sufficient free disk space in the file system to which the GuardPoint belongs. Free space is required for CTE-LDT attributes (CTE extended attributes or embedded attributes over NFS/CIFS shares) and (in Linux) metadata in the MDS file. CTE-LDT attributes are created during the initial encryption and are never freed until the GuardPoint is permanently unguarded and removed from the protection of an CTE-LDT policy. In contrast, disk space for metadata in the MDS file is temporary, kept only during the live transformation process.

When planning how much free disk space to reserve for CTE-LDT on a GuardPoint, consider the following:

- Number of files in the GuardPoint
- (Linux) Average length of absolute pathnames of files in the GuardPoint

The CTE-LDT process pre-allocates disk space for the Linux MDS file based on a minimum of 200K files with an average pathname of 1024 bytes per GuardPoint. The minimum space amounts to 325MB of disk space for the MDS file for each GuardPoint, even if file count is very low. (In Windows, CTE-LDT reserves the space when the file is rekeyed.)

Using voradmin To Estimate Disk Space Required for CTE-LDT (Linux)

In Linux, you can use the `voradmin ldt space` command to estimate the amount of disk space required for CTE-LDT attributes and the MDS file. The result is rounded to the nearest MB. The syntax of the command is:

```
# voradmin ldt space <guard path>
```

The following example shows how to estimate the required disk space to perform CipherTrust Transparent Encryption - Live Data Transformation on 1501 files in the GuardPoint `/oxf-fs1/gp1`. Estimate the disk space before protecting `/oxf-fs1/gp1` using an CTE-LDT policy. For example:

```
# voradmin ldt space /oxf-fs1/gp1
/oxf-fs1/gp1: found 1501 files without CTE-LDT extended attributes
CTE-LDT disk space requirements: total 169MB (CTE-LDT attributes=6MB, MDS=163MB)
```

The voradmin command reports that 1501 files in `/oxf-fs1/gp1` without CTE-LDT attributes. These files are new to CTE-LDT. 6MB of space is required for the CTE-LDT extended attributes and 163MB for metadata in the MDS file.

The following example shows the output of the same command after encryption completes. This estimates the additional disk space needed for the next CTE-LDT rekey operation. For example:

```
# voradmin ldt space /oxf-fs1/gp1
/oxf-fs1/gp1: found 0 files without CTE-LDT extended attributes
CTE-LDT disk space requirements: total 163MB (CTE-LDT attributes=0MB, MDS=163MB)
```

The voradmin command still reports on the same 1501 files, but because encryption has occurred using CTE-LDT, the files all have their CTE-LDT attributes. No additional space is required for CTE-LDT extended attributes on the next run. However, since the MDS file is transient and deletes after it finishes encrypting the GuardPoint, it requires additional space for the next key rotation.

> **Note**
> Windows estimates space using the following equations:
> - Permanent Space = Number of Files * 4K
> - Temporary Space = 128K * (Number of CPU * 2)

Displaying Metadata

Use the following command to see CTE-LDT file attributes or GuardPoint attributes:

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

For an overview of `voradmin ldt`, see "CTE-LDT Command-Line Administration: voradmin command" on page 92.

Verifying Metadata (Windows only)

Use the following command to verify if the metadata is corrupt:

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

For an overview of `voradmin ldt`, see "CTE-LDT Command-Line Administration: voradmin command" on page 92.

DFSR and Replication (Windows)

The Distributed File System Replication (DFSR) service is a multi-master replication engine used to maintain synchronized folders on multiple servers. Replicating data to multiple servers increases data availability and provides users in remote sites with fast, reliable access to files.

If you are creating GuardPoints in a DFSR environment, you must first add the DFSR private folder guard path to the exclusion list in the Windows Registry. CTE-LDT should not attempt data transformation on this read-only directory.

1. Using the Registry Editor, or the Windows command line, add a registry entry in: `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Vmmgmt\Parameters` named "LDTExclusionGPList" of type REG_MULTI_SZ.
2. Add the `DfsrPrivate` directory path to the `LDTExclusionGPList`.

   For example, if the DFSR private directory path is `D:\data\DfsrPrivate`, add this string in LDTExclusionGPList.
3. Reboot the system to make the change take effect.

> **Note**
> CTE-LDT does not perform a rekey on the DFSR private directory. Its rekey status is always ”N/A”.

If an application, or user, is performing a rename or a delete folder operation inside a GuardPoint with an CTE-LDT policy, this may restart the rekey process. Files which are already rekeyed will not be rekeyed again.

- If a rename or a delete operation is already performed on rekeyed files/folder, then the rekey process does not restart.
- If a rename or delete operation is performed in a folder where a rekey is in progress, CTE-LDT needs to stop the rekey process and restart the rekey again.

GuardPoint Management Over an NFS Share

Starting with CTE 7.1.0, you can enable GuardPoints over NFS shares using LDT policies. Full support for LDT over NFS is planned over two phases. In the first phase, you can enable GuardPoints over NFS on multiple CTE protected NFS clients while the GuardPoint is not undergoing transformation. Upon key rotation, CTE-LDT will begin transforming affected GuardPoints as long as each GuardPoint is enabled only on one of the NFS clients.

When a group of CTE protected NFS clients receives a key rotation notification from the DSM or CipherTrust Manager , each client will attempt to launch LDT on the affected GuardPoints on those hosts. When a client succeeds in launching LDT on a particular GuardPoint, that client takes over control of that specific GuardPoint and will conduct the data transformation on that GuardPoint as soon as other CTE clients disable the GuardPoint. During live data transformation, the CTE client authorized to access and/or enable the GuardPoint is the client that succeeded with the take-over until live data transformation completes on the entire GuardPoint.

The relationship between NFS clients and GuardPoints is many to one: a single NFS client can be responsible for rekeying many GuardPoints, but a single GuardPoint can only be rekeyed by a single NFS client.

The NFS client that ends up taking over the GuardPoint performs the following functions:

1. The client sends a message to the security server to announce its take over. An alert is sent every 30 seconds while the client waits for the GuardPoint to be disabled on the other clients.
2. The client delays starting LDT operations until the GuardPoint has been disabled on all other NFS clients. GuardPointsconfigured for Auto Guard must be disabled on the DSM. GuardPoints configured for Manual Guard must be disabled on each NFS CTE client using the `secfsd -unguard` command.. The client determines this through the use of tag files, as described in "CTE-LDT Host Tag Files" on page 78.

   > **Note**
   > The status of the client taking over the GuardPoint will change to "Waiting to Launch Rekey" while the status of all other clients sharing the GuardPoint will change to "Unguard Required" until the client taking over the GuardPoint is the sole owner of the GuardPoint. (NFS clients must be connected to the DSM for the status changes to take effect.)
   >
   > Messages will also be added to the log indicating that one client has taken over the GuardPoint and that the other clients need to unguard the GuardPoint so that data transformation can begin.

3. After the owning NFS client is the only NFS client with that GuardPoint enabled, the owning NFS client starts the LDT operation on the GuardPoint.
4. After LDT has completed successfully, the owning NFS client informs the key manager that data transformation has completed and the standard Rekey Successful message is posted in the log. At this point, the Administrator can reenable the GuardPoint on the other NFS clients that need to share access to the GuardPoint.

As described in "CTE-LDT Host Tag Files" on page 78, CTE maintains the list of CTE clients that enable a GuardPoint through tag files created in the LDT Private Space Directory of the GuardPoint by CTE when CTE enables the GuardPoint.

In the event of a host failure when the tag file of the host is present, you must manually remove the tag file of the failed host if the host does not recover or if the host is not rebooted to re-enable the GuardPoint. You can remove the tag file of a host by running the `voradmin ldt rmtag` command.

Multiple GuardPoint Pathnames

If you have multiple CTE-LDT hosts guarding and sharing the same CTE-LDT GuardPoint directory with different local pathnames, the pathname that will be associated with CTE-LDT operations is the pathname used on the CTE-LDT host that takes over the GuardPoint directory for live data transformation.

When you are entering `voradmin ldt` commands, you must use the host's local pathname for the GuardPoint when you specify the GuardPoint parameter. In response, voradmin displays the pathname used on the host that has taken over the GuardPoint for CTE-LDT operations. If a different host subsequently takes over the transformation of the GuardPoint, CTE-LDT changes the the pathname of the GuardPoint to the local pathname on the new CTE-LDT host.

For example, let's say you have two hosts (`LDT_Host_1` and `LDT_Host_2`) with the same directory in an NFS share mounted on different mount points on each host:

- GuardPoint on LDT_Host_1: `/nfs-oxf-fs1-host1/gp`
- GuardPoint on LDT_Host_2: `/nfs-oxf-fs1-host2/gp`

When the GuardPoint is first added to `LDT_Host_1`, that starts the initial data transformation:

```
# secfsd -guard /nfs-oxf-fs1-host1/gp
secfsd: Guard point initialization in progress
```

MDS has the GuardPoint configured for rekey at `/nfs-oxf-fs1-host1/gp`.

```
# voradmin ldt list all
MDS_1:  type=file, nguards=0, name=/nfs-oxf-fs1-host1/gp/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING DIRTY, flags=GP LOCKED, gp=/nfs-oxf-fs1-
host1/gp
                File List: count 4
```

Disable the GuardPoint on `LDT_Host_1` while rekey in progress:

```
# secfsd -unguard /nfs-oxf-fs1-host1/gp
secfsd: Path is not guarded
```

On `LDT_Host_2`, guard the same directory mounted at the different path:

```
# voradmin ldt list all

# secfsd -guard /nfs-oxf-fs1-host2/gp
secfsd: Path is guarded
```

MDS now has the GuardPoint configured for rekey at `/nfs-oxf-fs1-host2/gp` on `LDT_Host_2`.

```
# voradmin ldt list all
MDS_1:  type=file, nguards=0, name=/nfs-oxf-fs1-host2/gp/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING SUSPENDED (qos, flags=GP LOCKED, gp=/nfs-oxf-
fs1-host2/gp
                File List: count 4
```

LDT Configuration and Operations in a Multi-Node Cluster Environment

This section describes what to expect at runtime when CTE-LDT operates in a multi-node cluster environment with local file systems. It does *not* apply to GuardPoints over an NFS share.

CTE-LDT Behavior on Failover

In a multi-node cluster, CTE-LDT runs in active-passive mode. A GuardPoint is only enabled on the active node, and CTE-LDT only runs on the active node. On the DSM Management Console, you can only view statistics for the active host.

If the active host fails over, and the standby host takes control, CTE-LDT also fails over to the second host. On the DSM Management Console , CTE-LDT statistics now appear on the other host.

Backing Up and Restoring CTE-LDT GuardPoints

This section describes procedures and considerations related to backing up and restoring data in CTE-LDT GuardPoints in local file systems. For back up and restore over an NFS share, see "Linux-Specific Requirements" on page 22.

In addition to the files in a GuardPoint, CTE-LDT stores metadata in extended file attributes. These CTE-LDT attributes contain information that is required for decrypting the files and for the proper operation of CTE-LDT. Therefore, it is critical that your backup application also backs up the extended attributes of the files along with file data.

Clear Text Backup and Restore

A policy that applies a Security Rule with the Apply Key effect on backup/restore operations does not require any special rules for data access by backup applications. Under such a policy, backup applications always read clear data and store clear data in backup media. The backup application is not required to back up the CTE-LDT extended attributes, and a QoS schedule is not required to suspend CTE-LDT during backups.

Encrypted Backup and Restore

When a policy does not enforce a Security Rule with the Apply Key effect on backup/restore operations, the policy does not decrypt data on I/O operations from that backup application. Under such a policy, the backup application stores encrypted data and the CTE-LDT extended attributes of the file on the backup media.

In Linux, CTE-LDT operations must be suspended during backup. Suspending CTE-LDT completes the ongoing rekey operations on regions of files before starting the backup. During live transformation, CTE-LDT first preserves those regions of a file to be rekeyed in the MDS file. Then it updates some of the metadata in order to update the status of the data preserved in the MDS file in preparation for the rekey. Then it starts rekeying and updating those regions in the underlying file.

Suspending CTE-LDT waits for ongoing rekey operations to complete, and saves the metadata in the CTE-LDT extended attribute section of the MDS file. Suspending CTE-LDT ensures that the rekey status stored in the CTE-LDT extended attribute accurately reflects the rekey status of the data in the entire file during backup.

> **Note**
> This requirement does not apply to CTE-LDT for Windows.

The following table summarizes the state of the data in files in backup media:

CTE-LDT State of File	Security Policy	Backup Metadata or Alternate Data Streams Along with File data	Effect
Not rekeyed	Permit	Yes	Data in backup may be in clear format or encrypted with older key version.
Rekey in progress	Permit	Yes	File in backup storage is partially rekeyed. Some parts are in clear format or encrypted with older key version, and other parts are encrypted with current key version.
Rekey complete	Permit	Yes	File in backup storage is in the encrypted format with new key version.

CTE-LDT Policy for Encrypted Backup and Restore

Suppose you have an CTE-LDT policy allowing a backup user, or a backup process, to perform an encrypted backup. The backup user, or the backup process, reads encrypted data from files and stores the same encrypted data in backup media.

For example, suppose you wanted to back up GuardPoint `/oxf-fs1/gp1` protected by `My_LDT_Policy_1`. The key version before the key rotation is 8. The following steps occur:

1. Key version 8 expires and is rotated to version 9.
2. Rotating the encryption key triggers CTE-LDT to start running on the data under the GuardPoint.
3. The QoS schedule suspends CTE-LDT because a backup process is running.

4. The backup process begins, runs and later ends.
5. The QoS schedule resumes CTE-LDT.
6. CTE-LDT on the GuardPoint completes.

> **Note**
> Although the backup/restore scenarios and examples described in the rest of this section are specific to CTE-LDT on Linux platform, the concepts also apply to CTE-LDT on the Windows platform.

Consider the state of the three files in the GuardPoint during the CTE-LDT process, right after CTE-LDT is suspended in preparation for the backup. You can obtain the state of each file through the `voradmin ldt attr get` command, which examines the CTE-LDT state of each file captured in the CTE-LDT extended attributes of those files. (For an overview of `voradmin CTE-LDT`, see "CTE-LDT Command-Line Administration: voradmin command" on page 92).

- *File_1.dat* is rekeyed/encrypted to completion. The applied and new key versions are at version 9 of the key.

```
# voradmin ldt attr get /oxf-fs1/gp1/data_files /file_1.dat
 LDT attributes: rekeyed_size=1440743424, rekey_status=none
     Key:    name=LDT_AES256_KEY, version=9
```

- *File_2.dat* is partially rekeyed/encrypted. The applied key version is at version 8 and the transformation key is at version 9.

```
# voradmin ldt attr get /oxf-fs1/gp1/data_files/file_2.dat
 LDT attributes: rekeyed_size=1440743424, rekey_status=rekeying,state_saved
     Current Key:    name=LDT_AES256_KEY, version=8
     New Key:        name=LDT_AES256_KEY, version=9
```

- *File_3.dat* has not been rekeyed/encrypted. The applied key and transformation key are both at version 8, which is the version before the current key rotation.

```
# voradmin ldt attr get /oxf-fs1/gp1/data_files/file_3.dat
 LDT attributes: rekeyed_size=1440743424, rekey_status=none
     Key:    name=LDT_AES256_KEY, version=8
```

While CTE-LDT is suspended on the GuardPoint, the backup process starts and archives these three files, including extended attributes, in the backup media.

```
# my_backup --preserve=xattr /oxf-fs1/gp1/data_files/file_1.dat \
        /backup-media/oxf-fs1/gp1/data_files/file_1.dat
# my_backup --preserve=xattr /oxf-fs1/gp1/data_files/file_2.dat \
                /backup-media/oxf-fs1/gp1/data_files/file_2.dat
# my_backup --preserve=xattr /oxf-fs1/gp1/data_files/file_3.dat \
                /backup-media/oxf-fs1/gp1/data_files/file_3.dat
```

> **Notes**
> - On Linux platforms, you must perform backups on a GuardPoint during periods when CTE-LDT is not actively transforming data in the GuardPoint. You must schedule backups in conjunction with CTE-LDT schedules or during periods when CTE-LDT is paused. Make sure to coordinate with the backup administrator and schedule accordingly.
> - If the preferred method of backup is to create device level snapshots of the file system on the managed host or in the storage array subsystem, you must ensure that the schedule for creating snapshots pauses CTE-LDT before a snapshot is created.

Backup/Restore of Metadata Store File (MDS) in GuardPoints Undergoing Rekey

Backing up an entire GuardPoint using commands such as, `cp`, `tar`, or `rsync`, or even commercial backup products that sweep the file system namespace for backing up files is supported. This method of backup performed while live transformation is in progress over GuardPoint poses some challenges when files in GuardPoint are backed up in encrypted format. The challenge with this method of backup is existence of the CTE-LDT Metadata Store file (MDS) in GuardPoint during backup or restore process and strict protection enforced on MDS files preventing deletion or modification to MDS files. Second, your backup utility must backup and restore extended attributes of files and the GuardPoint directory, and because CTE-LDT extended attributes are also protected similar to MDS files, your backup must be enabled to restore extended attribute of files. To overcome both restrictions, your policy on GuardPoint must include a security rule without Apply Key Effect to enable your backup utility to replace MDS file and CTE-LDT extended attribute of files as part of backup/restore operations. Additionally, your backup utility must be signed with a signature set from the DSM or CipherTrust Manager for added security, and be executed with the option to preserve extended attribute options available with the standard Linux utilities.

> **Note**
> Support for backup/restore of entire GuardPoints is not currently supported for GuardPoints on an NFS share. This support will be added in a future CTE release.

The following table lists the supported backup utilities, required options to preserve extended attributes, and supported versions of the utilities.

Command	Required options	Supported version
`cp`	`--preserve=all`	OS default
`tar`	`--xattrs`	OS default
`rsync`	`-vapIXWP --inplace`	OS default
netbackup	overwrite existing files	v7.6.1 and v7.7.3

You can also backup and restore GuardPoint data, including the MDS file, if the requirements listed above on the backup utility are satisfied. GuardPoints associated with an MDS file located outside of GuardPoint namespace cannot be backed up or restore using this method of backup.

> **Note**
> You must suspend CTE-LDT on GuardPoint before performing backup or restore.

You can check your GuardPoint to determine if you can use this method of backup. If the directory of your GuardPoint is a mount point, MDS files reside inside your GuardPoint and this method of backup can be used for backing up or restoring your GuardPoint. However, GuardPoint directories below file system mount points must be checked and verified to use this method of backup. To verify, you can run the `voradmin` command to determine the MDS file associated your GuardPoint.

For example:

```
# voradmin ldt list all
MDS_1:  type=file, nguards=1, name=/oxf-fs1/gp1/::vorm:mds::

Guard Table: version 1 nentries 1
Guard 0: type=GP, state=REKEYING SUSPENDED (vadm, flags=GP LOCKED, gp=/oxf-fs1/gp1
File List: count 4308
```

The report on `/oxf-fs1/gp1` GuardPoint indicates association of the MDS file `/oxf-fs1/gp1/::vorm:mds` with the GuardPoint. As the MDS file resides inside the GuardPoint directory, you can use this method of backup to backup
and restore this GuardPoint.

Before restoring the GuardPoint, you must suspend CTE-LDT operations on the GuardPoint if live transformation is in progress. Failure to do so will fail to restore the MDS file in the backup image. Failure to restore MDS file affects partially rekeyed files restored to the GuardPoint. In such a situation the restored data is invalid, and you must remove all the files restored to the GuardPoint and repeat the restore operation after suspending CTE-LDT.

Restoring a GuardPoint from a Backup

To properly restore GuardPoint along with the MDS file, the following steps must be done in order:

1. Verify if live transformation is in progress on the GuardPoint and suspend the CTE-LDT operations. (Skip this step if no live transformation is occurring.)
2. With the same tool that was used for backup, restore to the GuardPoint.
3. Once restore is complete, GuardPoint needs to be disabled and enabled again. Run the `voradmin` command as shown below to determine how the GuardPoint was suspended at the time of backup. For manual

   GuardPoint, run the `secfsd` command as shown below. For auto-guards, disable and enable GuardPoints on DSM or CipherTrust Manager.

   ```
   # secfsd -unguard <GuardPoint>
   # secfsd -guard <GuardPoint>
   ```

4. Since GuardPoint was in suspended state during the time of backup, it will be restored in suspended state. You must resume CTE-LDT to complete CTE-LDT on the restored GuardPoint. You must resume CTE-LDT on the DSM or CipherTrust Manager if CTE-LDT was suspended on the DSM or CipherTrust Manager at the time of backup, otherwise you will resume CTE-LDT using `voradmin` command.
   Run the `voradmin` command below to determine if CTE-LDT must be resumed on DSM or CipherTrust Manager or through `voradmin`. If the tag string next to SUSPENDED state of GuardPoint is `vadm`, run `voradmin` command to resume CTE-LDT, otherwise the tag value is `qos` and CTE-LDT must be resumed on DSM or CipherTrust Manager. In the example below, GuardPoint `/oxf-fs1/gp1` was suspended using `voradmin`, and it must be resumed after restore using `voradmin` command.

   ```
   # voradmin ldt list all
   MDS_1:  type=file, nguards=1, name=/oxf-fs1/gp1::vorm:mds
           Guard Table: version 1 nentries 1
           Guard 0: type=GP, state=REKEYING SUSPENDED (vadm), flags=GP LOCKED, gp=/oxf-
   fs1/gp1
                   File List: count 4308
   # voradmin ldt resume /oxf-fs1/gp1
   ```

5. Wait for CTE-LDT completion on GuardPoint.
6. We strongly recommend that you disable and re-enable the GuardPoint once more.

The restore is now complete and files in your GuardPoint can be accessed.

The CTE Agent sends the following alert message to DSM or CipherTrust Manager if the restore operation is rejected:

```
LDT-ALERT: Restore of LDT protected file <GuardPoint> not allowed by policy
```

Potential Warnings During Restore Operation

When using `cp` for backup/restore, the `cp` command may report a failed attempt to preserve permissions on the Metadata Store File (MDS) during a restore. If you encounter the below message, continue to proceed with the restore steps since this will not affect the MDS file or dataset that is being restored.

```
cp: preserving permissions for '/oxf-fs1/gp1/::vorm:mds::': Permission denied
```

When using `rsync` for backup/restore, the `rsync` command may report a failed attempt to set extended attribute when restoring the MDS file on system with selinux configured in enforced mode. If user encounters the below message, continue to proceed with the restore steps since this will not affect the MDS file or dataset that is being restored.

```
rsync: copy_xattrs: lsetxattr(""/oxf-fs1/gp1/::vorm:mds::"","security.selinux") failed:
Permission denied (13)
rsync: rsync_xal_set: lsetxattr(""/oxf-fs1/gp1/::vorm:mds::"","security.selinux")
failed: Permission denied (13)
```

Restore an Encrypted Backup

This section illustrates restoration of the three files from the backup media to the same GuardPoint, */oxf-fs1/gp1*. The files are restored to a different directory under the GuardPoint.

Restore a File Fully Rekeyed to the Latest Key Version

Recall that *file_1.dat* was archived in the backup media when it was fully rekeyed to version 9 of the key. As the current version of the key is also 9, *file_1.dat* is restored from backup without any changes. After restoring the file, the state of the restored file and its applied and current key versions remain unchanged, as compared to the original file that was backed up.

```
# my_backup --preserve=xattr /backup-media/oxf-fs1/gp1/data_files \
/file_1.dat /oxf-fs1/gp1/restored_files/file_1.dat

# voradmin ldt attr get /oxf-fs1/gp1/restored_files/file_1.dat
  LDT attributes: rekeyed_size=1440743424, rekey_status=none
     Key:    name=LDT_AES256_KEY, version=9

# voradmin ldt attr get /oxf-fs1/gp1/data_files/file_1.dat
  LDT attributes: rekeyed_size=1440743424, rekey_status=none
     Key:    name=LDT_AES256_KEY, version=9
```

Restore a Partially Rekeyed/encrypted File

Recall that *file_2.dat*, archived in the backup media, was partially rekeyed between versions 8 and 9. As the current version of the key was 9 at the time of backup, *file_2.dat* is restored to the GuardPoint with the same version of the key from the time of backup. The file is flagged for *lazy rekey*, meaning that a background rekey operation is scheduled to transform the file to the latest key version the next time an application tries to access the file.

At the completion of restoration, the file is fully transformed to the key version (v9). The key version is also the latest one in the policy. Although the file is flagged for lazy rekey (LAZY_RK), the file does not need to be transformed to the latest key version because it's already there. Had the file been partially rekeyed from version 7 to version 8 of the key at the time of backup, the restored file would have completed rekeying to version 8 at the end of the restoration. Therefore, the LAZY_RK flag would initiate a background transformation to update the key version to the latest key version when the file is accessed.

If this file is not accessed by any application, the file remains unchanged in the GuardPoint. It is not transformed to the latest key version. To trigger a rekey, either re-push the CTE-LDT policy from the DSM or CipherTrust Manager, or access the file with an application, such as a file explorer.

```
# my_backup --preserve=xattr /backup-media/oxf-fs1/gp1/data_files/  \
file_1.dat /oxf-fs1/gp1/restored_files/file_2.dat

# voradmin ldt attr get /oxf-fs1/gp1/restored_files/file_2.dat
LDT attributes: rekeyed_size=1440743424, rekey_status=lazy_rekey
     Key:    name=LDT_AES256_KEY, version=9

# voradmin ldt attr get /oxf-fs1/gp1/restored_files/file_2.dat
 LDT attributes: rekeyed_size=1440743424, rekey_status=none
     Key:    name=LDT_AES256_KEY, version=9
```

Restore a File Not Rekeyed/encrypted with an Older Key Version

Recall *file_3.dat* was archived in the backup media when it was keyed to version 8 of the key, one version below the latest version at the time of backup. At completion of the restoration, *file_3.dat* is restored from backup to the same version, version 8, that it was keyed to when it was backed up.

However, the file is flagged for lazy rekey. After restoring the file, it is keyed to version 8 and flagged for lazy rekey (LAZY_RK). The file is rekeyed to the latest key version, version 9, as soon as an application accesses the file. If this file is not accessed by any application, the file remains unchanged in the GuardPoint.

```
# my_backup --preserve=xattr /backup-media/oxf-fs1/gp1/data_files \
/file_3.dat /oxf-fs1/gp1/restored_files/file_3.dat

# voradmin ldt attr get /oxf-fs1/gp1/restored_files/file_3.dat
LDT attributes: rekeyed_size=1440743424, rekey_status=lazy_rekey
     Key:    name=LDT_AES256_KEY, version=8

# sum /oxf-fs1/gp1/restored_files/file_3.dat
39994 1406976

# voradmin ldt attr get /oxf-fs1/gp1/restored_files/file_3.dat
LDT attributes: rekeyed_size=1440743424, rekey_status=none
     Key:    name=LDT_AES256_KEY, version=9
```

Restoring Non-CTE-LDT Backup Data to an CTE-LDT GuardPoint

This section describes how to restore data encrypted with a non-versioned key to an CTE-LDT GuardPoint.

If the backup was performed with the Apply Key effect, the backup files are in clear text. Simply restore the clear text files to the CTE-LDT GuardPoint with the Apply Key effect. All files will be encrypted with the versioned key.

If the backup of the non-CTE-LDT GuardPoint was performed without the Apply Key effect, the backup is encrypted, and you must do the following:

> **Note**
> The following example is for a manual guarding. The steps may differ slighly if your GuardPoint is configured for auto guard.

1. Create a temporary directory for restoring the files, type:

   ```
   # mkdir -p /oxf-fs1/tmp_restore
   ```

2. Restore the encrypted backup files into the temporary directory, type:

   ```
   # cp -pr /backup-media/oxf-fs1/gp1/data_files/* /oxf-fs1/tmp_restore
   ```

3. Create a Standard Policy with the Apply Key effect for all operations, using the same key as the policy applied on the GuardPoint at the time of backup.

4. Create and enable a new GuardPoint for the temporary directory using the Standard Policy just created.

   ```
   # secfsd -guard /oxf-fs1/tmp_restore
   ```

5. Ensure that the temporary GuardPoint and CTE-LDT GuardPoint are both enabled.

   ```
   # secfsd -status guard
   GuardPoint             Policy      Type       ConfigState  Status       Reason
   ----------             ------      ----       -----------  ------       ------
   /oxf-fs1/gp1           LDT_AES256  manual     guarded      guarded      N/A
   /oxf-fs1/tmp_restore   AES256      manual     guarded      guarded      N/A
   ```

6. Move the restored files from the temporary folder to the GuardPoint enabled with the CTE-LDT policy. The CTE agent encrypts the files in the CTE-LDT GuardPoint using the current key version in effect for the CTE-LDT policy.

   ```
   # mv /oxf-fs1/tmp_restore/* /oxf-fs1/gp1
   ```

7. Disable the temporary GuardPoint and remove the temporary restore directory.

   ```
   # secfsd -unguard /oxf-fs1/tmp_restore
   # rm -fr /oxf-fs1/tmp_restore
   ```

8. Delete the temporary GuardPoint on the DSM or CipherTrust Manager.

Using fsfreeze (Linux only)

If you use the `fsfreeze` command to quiesce access to the file system before creating a snapshot, refer to the fsfreeze section in CTE Admin Guide in the Linux Utilities chapter on how to run the fsfreeze command to quiesce access to both the file system and the GuardPoint(s).

> **WARNING**
> **Do not use CTE-LDT schedules and do not pause CTE-LDT to align backup schedules with CTE-LDT. Instead, use the `fsfreeze` command.**

- Running `fsfreeze -f` on the GuardPoint directory pauses CTE-LDT operations in-progress and freezes access to both the GuardPoint and the underlying file system.
- Running `freeze -u` reverses `freeze -f`, allowing access to underlying file system and resuming CTE-LDT operations.

CTE-LDT Backups Using a File System or Storage-Level Snapshot Tool

You can make a file system snapshot using a Logical Volume Manager service or mirroring/splitting storage level LUNs of a file system inside the storage subsystem. CTE-LDT does not have requirements for where and how you create a file system snapshot. However, it is required that you **suspend CTE-LDT processes before you take the file system snapshot**. Suspending CTE-LDT ensures data and metadata consistency between files and CTE-LDT extended attributes.

You may choose to suspend CTE-LDT manually on the managed host using `voradmin ldt suspend` command or `fsfreeze -f`, or suspend CTE-LDT on the DSM or CipherTrust Manager.

> **Note**
> Be aware that suspending CTE-LDT on the DSM or CipherTrust Manager suspends CTE-LDT on the entire host.

After creating a file system snapshot, you can resume CTE-LDT processes on the GuardPoint using `voradmin ldt resume`, or `fsfreeze -u`, or resuming CTE-LDT on the DSM or CipherTrust Manager. Do not mix the use of `fsfreeze` and `voradmin ldt suspend` to pause and resume CTE-LDT. CTE suspends or resumes CTE-LDT processes during live transformation when freezing or unfreezing GuardPoint access using `fsfreeze -f` or `-u` option. See the *CTE Agent for Linux Advanced Configuration and Integration Guide* on the use of the `fsfreeze` command on a GuardPoint.

> **Note**
> You can make sure CTE-LDT is suspended at backup time by setting the QoS schedule.

You can mount a file system snapshot for data recovery. Configuration for GuardPoints must be duplicated over the mount point of the snapshot file system. Make sure to use the same CTE-LDT policy. Enabling GuardPoints over or under the snapshot mount point provides access to the protected files for recovery. You can choose to manually resume key rotation on the GuardPoints of the snapshot file system, although this is not necessary.

Following is an example of the `fsfreeze` command used to freeze access to the file system `/oxf-fs1` in order to create a snapshot of the file system device. This examples illustrates three GuardPoints enabled inside the file system namespace, `/oxf-fs1/gp-1`, `/oxf-fs1/gp-2`, and `/oxf-fs1/gp-3`. Executing the command `fsfreeze -f` targets any of the GuardPoints in the `/oxf-fs1` mount point and suspends CTE-LDT processes on all of the GuardPoints. Then it freezes access to the file system.

```
# fsfreeze -f /oxf-fs1/gp-1
# voradmin ldt list all
MDS_1:  type=file, nguards=1, name=/oxf-fs1/gp-3/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING SUSPENDED (vadm, flags=GP LOCKED,
gp=/oxf-fs1/gp-3
                           File List: count 4308

MDS_2:  type=file, nguards=1, name=/oxf-fs1/gp-2/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING DIRTY, flags=GP LOCKED, gp=/oxf-fs1/gp-2
                        File List: count 4308

   MDS_3:  type=file, nguards=1, name=/oxf-fs1/gp-1/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING SUSPENDED (vadm, flags=GP LOCKED, gp=/oxf-
fs1/gp-1
                File List: count 4308
```

After the file system snapshot is created, executing the `fsfreeze -u` command on any of the GuardPoints in the file system namespace unfreezes access to the file system and resumes CTE-LDT processes on all of the GuardPoints.

```
# fsfreeze -u /oxf-fs1/gp-1
# voradmin ldt list all
MDS_1:  type=file, nguards=1, name=/oxf-fs1/gp-3/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING DIRTY, flags=GP LOCKED, gp=/oxf-fs1/gp-3
                File List: count 4308
```

```
MDS_2:  type=file, nguards=1, name=/oxf-fs1/gp-2/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING DIRTY, flags=GP LOCKED, gp=/oxf-fs1/gp-2
                              File List: count 4308

     MDS_3:  type=file, nguards=1, name=/oxf-fs1/gp-1/::vorm:mds::
        Guard Table: version 1 nentries 1
        Guard 0: type=GP, state=REKEYING DIRTY, flags=GP LOCKED, gp=/oxf-fs1/gp-1
                File List: count 4308
```

NFS Server Backups

As described in "Backup Requirements" on page 22, CTE-LDT requires suspending all data transformation during the backup window. Suspending CTE-LDT enforces data and metadata consistency in files undergoing data transformation. It's important to align the schedules for CTE-LDT and backup/restore to ensure consistency of the data and the metadata of affected files in the backup image.

Windows Backup and Snapshots

On Windows, most backup applications use Volume Snapshot Service (VSS) for the backup. Using VSS is required for backing up files in CTE-LDT GuardPoints. VSS service provides a consistent view of the data to backup and restore applications by taking a snapshot of the volume. Windows Backup uses the snapshot volume, while other applications can continue using the original volume for normal I/O operations. VSS snapshot volume uses a “Copy on write” mechanism to provide a consistent view of the data. Some of the high-level steps of the Windows backup process are:

1. Backup application takes a snapshot of the volume using the VSS service.
2. Backup application mounts this VSS volume to read the data to be backed up.
3. The CTE Agent uses the same policy as that of the original volume to protect these snapshots.
4. The CTE Agent applies the policy and rules to all the I/O requests coming from the backup and restore applications.

Restoring ESXi VM Snapshots of a Protected Host

Some organizations may use ESXi virtual machines for protected hosts and create VM snapshots to back up sensitive data. If you are restoring an older ESXi VM snapshot of a protected host, that contains an previous encryption key, follow these steps:

1. Restore the VM to update to the latest version of the encryption key.
2. In the DSM Management Console, choose **Hosts** > **Hosts**, then click the name of the VM host.
3. Click the **GuardPoint** tab.
4. Click **Re-Push Policies**.

CTE-LDT Backup and Restore Troubleshooting

Restored files to a GuardPoint protected with conflicting key rules

When restoring an encrypted file from backup media to an CTE-LDT protected GuardPoint without the Apply Key effect, and the file in the backup media does not have an CTE-LDT extended attribute, the file restored to the GuardPoint is set with an CTE-LDT extended attribute that specifies the current key version of the key in the policy associated with the data in the restored file. As the key and key version in the policy do not match the key that was applied to the data at the time of backup, the file restored to the CTE-LDT protected GuardPoint is unreadable.

When restoring an encrypted file from backup media to an CTE-LDT protected GuardPoint without Apply Key effect, and the key specified in the CTE-LDT extended attribute of the file in backup media conflicts with the key rules of the policy on the GuardPoint, the restore operation fails and flags the restored file in error. You can only remove the file, or truncate it, to clear the error status on the file. Access to such files, except remove or truncate, fail with an EINVAL error.

CTE-LDT Command-Line Administration: voradmin command

Use the `voradmin` utility to gather statistics and administer CTE-LDT on a host/client . It has slightly different syntax and command capabilities depending on whether the host/client is running Linux or Windows.

> **Note**
> Refer to the `voradmin` man page for `voradmin` usage in Linux.

To use `voradmin`:

1. Log in to the host running the CTE Agent with CTE-LDT enabled.
2. At the command line, type `voradmin`.
3. Follow the usage outputs onscreen to find the available commands and their syntax. The general syntax is:
   ```
   # voradmin ldt <command> [args]
   ```

Command	Description
`voradmin ldt attr get|delete <guard path>|<object path>`	Get or delete attributes for a GuardPoint or Object path.
`voradmin ldt key <report|map <key> > <guard path>`	Creates a report, or map, containing statistics after each key rotation on a GuardPoint.
`voradmin ldt list all`	(Linux only) List all MDS files and GuardPoints undergoing CTE-LDT processing.
`voradmin ldt monitor [interval]`	(Windows only) Monitor CTE-LDT progress. If an interval is specified (in units of seconds), it continually updates the monitoring output at the specified interval.
`voradmin ldt rekey report <guard path> [<output file>]`	Generate a rekey report manually for a Guardpath.
`voradmin ldt resume <guard path>|all`	Resume guarding the GuardPoint directory.

Command	Description
`voradmin ldt rmldt <guard path>` `voradmin ldt rmstore <guard path>`	Both the `rmstore` and `rmldt` options remove the LDT metadata information remaining in a GuardPoint directory after permanent unguarding of the specified GuardPoint. The `rmstore` option removes the MDS file, if it exists. The `rmldt` removes the MDS file and the files in LDT Private Directory for GuardPoints over NFS shares.
`voradmin ldt rmtag <guard path>`	Removes the LDT NFS tag file for a CTE-LDT host from the specified GuardPoint. Not specifying a host will delete the tag file for the current host on which the `voradmin` command is executed. **Important:** Do *not* remove the tag file for a host that still has the specified GuardPoint guarded.
`voradmin ldt space <guard path>`	Estimates disk space needed for metadata and mds store information.
`voradmin ldt stats [<guard path>] [<interval>]`	Obtains transformation CTE-LDT statistics such as: • Current rekey status • Start time • Estimated completion time • Percentage completed • Total data • Amount of data transformed • Total files • Number of files transformed • Number of files skipped • Number of files remaining for rekey

Upgrading or Downgrading Agent Software On an CTE-LDT Host

Consider the following when upgrading or downgrading software on protected hosts.

Upgrading

If a host contains an agent software version lower than 6.0, and you upgrade to version 6.0 or later, the CTE-LDT feature does not automatically enable on all upgraded hosts. You must:

1. Upgrade the DSM.
2. Install the CTE-LDT license.
3. Restart the 6.x agent, or reboot the host on which the CTE Agent is installed.
4. Select the CTE-LDT option to enable it, once the CTE-LDT option displays in the host's menu.

After upgrading the DSM, the keys and policies on the DSM remain unchanged. They are standard keys and policies not usable with CTE-LDT. To use CTE-LDT, create versioned keys and policies set to type: CipherTrust Transparent Encryption - Live Data Transformation.

On the DSM, select the Live Data Transformation option on the **Edit > Host** page to enable CTE-LDT on target host.

After upgrading a protected host to 6.x from an earlier version, existing policies, keys, and GuardPoints on the managed host remain unchanged and enforced.

> **Note**
> In v6.1, you can automatically register CTE-LDT with the DSM installation.

For more information, see Chapter 3: "Setting Up CTE-LDT" on page 21.

> **Note**
> If you are using CTE-LDT with the Hadoop Distributed File System (HDFS), see the HDFS chapter in the *CTE Agent Installation and Configuration Guide* for more information.

Downgrading

To roll back from 6.0 (or greater) to a version before 6.0, it is not sufficient to uninstall the 6.0 software and reinstall the earlier version. You must migrate GuardPoints protected with CTE-LDT encryption policies to a non-CTE-LDT policy first, reverse the CTE-LDT host's registration with the DSM, then re-register the host after installing the earlier software version. See "Removing CTE-LDT and Security Encryption" on page 96.

Migrating a GuardPoint to a Different CTE-LDT Policy

To change the CTE-LDT policy that an CTE-LDT GuardPoint uses, complete the following steps to ensure that the GuardPoint is migrated properly from one CTE-LDT policy to another CTE-LDT policy.

> **Note**
> This procedure is for the migration of local file system GuardPoints only. For information about migrating NFS GuardPoints to clear_key, see "Migrating GuardPoints over NFS From or To an LDT Policy" on the next page.

Scenario

The GuardPoint is currently attached to LDT-Policy-1, which rekeys from `clear_key` to `LDT-Key-1`. The objective is to migrate the data in the GuardPoint to another versioned key, `LDT-Key-2`. Migration to `LDT-Key-2` requires detaching the GuardPoint from `LDT-Policy-1`, and then attaching it to `LDT-Policy-2` , assuming `LDT-Key-2` is the versioned key specified in `LDT-Policy-2`. To do so:

1. Clone the latest version of the key `LDT-Key-1` to a *non-versioned* key such as `LDT-Key-1-Clone`.
   a. In the DSM Management Console, select **Keys > Agent Keys > Keys**.
   b. Click the name of the versioned key that you want to clone.
   c. Click **Clone** to clone the key.
2. Identify or create a new *versioned* key that you want CTE-LDT to use to re-encrypt the data. For example, `LDT-Key-2`.
3. Create a new Live Data Transformation policy that specifies `LDT-Key-1-Clone` as the Initial Key and `LDT-Key-2` as the Transformation Key.
4. Make sure that all data transformation has completed on the GuardPoint. To verify this, use the `voradmin ldt attr delete <GuardPoint>` command.
5. In your key manager, unguard the GuardPoint.
6. On the host, remove the existing CTE-LDT attributes on the GuardPoint using the `voradmin ldt attr delete <GuardPoint>` command.

```
# voradmin ldt attr delete /oxf-fs1/gp1
LDT metadata has been removed from all files in GuardPoint /oxf-fs1/gp1
```

7. Guard the directory using the new LDT policy.
   - If you have selected Auto Guard, data transformation begins as soon as the host gets the new policy information from the key manager.
   - If you have selected Manual Guard, use the `secfsd -guard <GuardPoint>` command on the host to begin data transformation.

Migrating GuardPoints over NFS From or To an LDT Policy

GuardPoints over NFS can be guarded using policies with key rules specifying a mix of CBC or CBC-CS1 keys.

> **Note**
> This section applies to NFS GuardPoints only. For information about migrating local file system GuardPoints, see "Migrating a GuardPoint to a Different CTE-LDT Policy" on the previous page.

You can migrate to an LDT policy from clear text (no existing policy) or from a standard CTE policy. If the standard policy uses:

- **CBC keys**: The CBC keys are transformed during the initial data transformation when CTE-LDT embeds the LDT metadata in the beginning of each file. CTE-LDT shifts the existing data in the files by 4096 bytes to make room for the LDT metadata.
- **CBC-CS1 keys** The IV attribute is already embedded in the protected files. CTE-LDT transforms the files in those GuardPoints without shifting the existing data because the required IV attribute already exists.

Migration out of an LDT policy is only *partially* supported because of the shift to the existing data that was done to accommodate the LDT metadata. You can only migrate from a Live Data Transformation policy that uses CBC or CBC_CS1 keys to a different Live Data Transformation policy that uses CBC or CBC_CS1 keys. You cannot remove the Live Data Transformation policy from a guarded directory, and you cannot migrate from a Live Data Transformation policy to a Standard CTE policy. The migration support matrix is shown in the following table.

Source Policy Type	Target Policy Type	Supported?
Live Data Transformation using CBC or CBC_CS1 keys	Live Data Transformation using CBC or CBC_CS1 keys	Yes
Live Data Transformation	No policy (unguarded directory)	No
Live Data Transformation	Standard CTE policy	No
Standard Policy using CBC or CBC_CS1 keys	Live Data Transformation using CBC or CBC_CS1 keys	Yes

The only way you can migrate an NFS GuardPoint from LDT to clear-text or to a CTE standard policy that uses CBC or CBC_CS1 keys is to do the following:

1. Perform a full backup of files in the NFS GuardPoint in clear text. Make sure that you disable the security rule for the backup process in the LDT policy if the security rule skips Apply Key as part of backup operation.
2. Upon completion of full backup, unguard the directory on the key manager and then remove the LDT Private Space directory (`ldtprivspace`) in the NFS GuardPoint using the `voradmin ldt rmldt <GuardPoint>` command.
3. Remove the remaining files inside the NFS GuardPoint directory and restore the full backup of the files in clear-text over the NFS GuardPoint directory. If you are migrating to a standard CTE policy, you can now proceed with re-guarding the NFS GuardPoint directory using the standard policy.

Removing CTE-LDT and Security Encryption

If you want to stop using CTE-LDT on a GuardPoint or on a whole host/client , follow the instructions in the following sections.

Migrating a GuardPoint Out of CTE-LDT

Migrating a GuardPoint from CTE-LDT removes the security encryption. It also provides an Administrator with the flexibility to relax the compliance requirement, when strict compliance for frequent key rotation on specific data is no longer mandatory. The following sections describe how to migrate a GuardPoint from CTE-LDT to a non-CTE-LDT policy, or to remove encryption protection from it.

Converting a GuardPoint from an CTE-LDT Policy to a non-CTE-LDT Policy

If you want to do more than just change the policy on a GuardPoint from an CTE-LDT policy to a non-CTE-LDT policy, see "Deleting CTE-LDT Metadata (Linux)" on page 99 or "Deleting CTE-LDT Metadata (Windows)" on page 100.

> **Note**
> Converting GuardPoints from Live Data Transformation policies to Standard CTE policies is not supported. Similarly, CTE-LDT protected GuardPoints cannot be migrated to clear_key.

1. Clone the versioned key associated with the CTE-LDT GuardPoint to a non-versioned key.

   The clone function creates a new key with the same cryptographic encryption material as the current version of the cloned versioned key.

   This allows CTE-LDT to use the cloned key in a non-CTE-LDT policy to convert the GuardPoint from an CTE-LDT to a non-CTE-LDT managed policy.

   a. In the DSM Management Console, select **Keys > Agent Keys > Keys**.

   b. Click the name of the versioned key that you want to clone.

   c. On the **General** tab, click **Clone** to clone the current version of the key.

   By default, the DSM creates a name for the cloned key that combines the name of the key and the current version number. You can specify a different name if desired. For example, if you are cloning version 5 of a key named `LDT-Key-1`, the name of the cloned key would default to `LDT-Key-1__5`.

2. Click **Hosts > *host_name* > GuardPoint** tab.

3. Select the GuardPoint and then click **Disable**.

4. After CTE-LDT disables the GuardPoint, click **Unguard** to delete the GuardPoint.

5. (Linux only) Ensure that the GuardPoint is removed on the managed host and the GuardPoint directory does not appear in the output of the `secfsd -status guard` command:

   ```
   # secfsd -status guard
   ```

6. CTE-LDT creates extended attributes for every file under the GuardPoint as well as the GuardPoint directory. Now that the CTE-LDT policy does not manage the GuardPoint, you must remove the extended attributes for every file in the GuardPoint, type:

   ```
   # voradmin ldt attr delete <GuardPoint>
   ```

   The command may take some time depending on the number of files in the GuardPoint.

   > **Note:** For all of the file system mount points that contain an CTE-LDT protected GuardPoint, you must clean up the metadata first. See "Deleting CTE-LDT Metadata (Linux)" on page 99.

7. If the GuardPoint is on an NFS/CIFS share, you must remove the LDT Private Space Directory using the `voradmin ldt rmldt <guardpoint>` command.

   ```
   # voradmin ldt rmldt <GuardPoint>
   ```

8. Create a non-CTE-LDT policy using the cloned key you created earlier in this procedure.
   a. Click **Policies > Manage Policies > Manage Policies**.
   b. Click **Add**.
   c. In Policy Type, choose **Standard**.
   d. In Key Selection Rules, click **Add**.
   e. In Key, choose the cloned key1.
   f. Click **Select Key**.
   g. Finish creating the policy as usual for a non-CTE-LDT policy.
9. Apply the non-CTE-LDT policy to the GuardPoint.

   CAUTION
   Make sure that you have removed all of the CTE-LDT metadata from the GuardPoint before applying the non-CTE-LDT policy.

   a. Click **Hosts > *host_name* > GuardPoint** tab.
   b. Click **Guard**.
   c. In the *Guard File System* window, select the Policy that you just created.
   d. Select *Type*: **Directory (Auto Guard)** or **(Manual Guard)**, and enter or browse to the directory to protect.

Remove Protection from a GuardPoint

When compliance may no longer require protecting data in a GuardPoint, you may choose to unprotect/decrypt it. Before removing protection from your GuardPoint, you must decrypt the data in your GuardPoint by setting it to clear. You have two options to decrypt your data:

- While a GuardPoint is protected and enabled under an CTE-LDT policy, you can use copy or backup/restore commands to save files in your GuardPoint to a location outside of your GuardPoint.
- Use the dataxform command to transform your GuardPoint to clear in an offline transformation process.

For GuardPoints over NFS, you must backup the entire GuardPoint before you unguard the GuardPoint. Then you can restore the files from backup over the GuardPoint directory after you remove the protection on the GuardPoint.

Copying Files to Decrypt Them

If you choose to copy your files, you must create a directory outside of the GuardPoint and then copy the files into the GuardPoint directory. After finishing copying, complete the following steps:

1. Click **Hosts > *host_name* > GuardPoint** tab.
2. Select the GuardPoint and then click **Disable**.
3. After CTE-LDT disables the GuardPoint, click **Unguard** to delete the GuardPoint.
4. (Linux only) Ensure that the GuardPoint is removed on the managed host:

   ```
   # secfsd -status guard
   No GuardPoints configured
   ```

This completes removal of the GuardPoint under an CTE-LDT policy. You can now remove the original files and data within the GuardPoint namespace.

Using Dataxform Command to Transform the Files

If you choose to use the dataxform command to transform data in your GuardPoint to clear, use the `voradmin` command to verify that earlier versions of the versioned key are not in use on your GuardPoint. Complete the following steps to clear all metadata in your GuardPoint. Then, transform your GuardPoint to clear.

1. Clone the key.
   a. In the DSM Management Console, select **Keys > Agent Keys > Keys**.
   b. Click the name of the versioned key that you want to clone.
   c. Click **Clone** to clone the key.

   By default, the DSM uses the name of the versioned key, and the current version number of the versioned key, for naming the cloned key. You can choose a different name, if desired.
2. Create an offline dataxform policy to transform your GuardPoint from the cloned key created in step 1 to `clear_key`. See the *DSM Administration Guide* for more information.
3. In the DSM Management Console, click **Hosts > Hosts > *Hostname***.
4. Click the **GuardPoint** tab.
5. Click **Disable** to disable the GuardPoint.
6. After CTE-LDT disables the GuardPoint, click **Unguard** to delete the GuardPoint.

   Before you apply the offline data transformation policy to your GuardPoint, you must clean up the CTE-LDT metadata from your GuardPoint. CTE-LDT creates extended attributes for every file under the GuardPoint, as well as the GuardPoint directory. Now that the CTE-LDT policy does not manage the GuardPoint, you can remove the extended attributes for every file in the GuardPoint.
7. Remove the extended attributes of files in a GuardPoint, type:

   ```
   # voradmin ldt attr delete <GuardPoint>
   ```

   The command may take some time depending on the number of files in the GuardPoint. After metadata deletion is complete, you can apply the offline transformation policy on the GuardPoint.
8. On the DSM, guard and enable the GuardPoint with the offline dataxform policy you created earlier.
9. After enabling your GuardPoint, run the dataxform command on the managed host to transform the GuardPoint to a `clear_key`, type:

   ```
   # dataxform --rekey --gp /<GuardPoint>/ --preserve_modified_time --preserve_access_
   time --cleanup_on_success
   ```
10. After completion of dataxform, unguard the GuardPoint.
    a. Click the **Hosts > *Hostname* > GuardPoint** tab.
    b. Select the GuardPoint and click **Disable**.
    c. After CTE-LDT disables the GuardPoint, click **Unguard** to delete the GuardPoint.
11. Remove the GuardPoint from the dataxform policy on the DSM.
    a. Click **Policies > *PolicyName***.
    b. Select the **GuardPoint**.
    c. Click **Delete**.
    d. Click **Apply**.

Deleting CTE-LDT Metadata (Linux)

To remove the metadata associated with the GuardPoint, you must run the voradmin command on the managed host to remove the CTE-LDT metadata associated with the GuardPoint.

> **WARNING**
> **Before you attempt to remove an MDS file, make sure that no CTE-LDT-protected GuardPoints remain configured under the file system mount point.**

In the following example:

- `/oxf-fs1` is the mount point
- `/oxf-fs1/gp1` is the CTE-LDT-protected GuardPoint

To delete the metadata associated with the GuardPoint:

1. Ensure that the GuardPoint is not enabled on the host. Run the command below and verify that the GuardPoint pathname does not appear in the output of the secfsd command, type:

   ```
   # secfsd -status guard
   ```

2. Run the `voradmin` command to remove CTE-LDT attributes on the GuardPoint, type:

   ```
   # voradmin ldt attr delete <GuardPoint path>
   ```

   For example:

   ```
   # voradmin ldt attr delete /oxf-fs1/gp1
   LDT metadata has been removed from all files in GuardPoint /oxf-fs1/gp1
   ```

If the last transformation on your GuardPoint completed with version 6.0.2 of the VTE Agent (renamed the CTE Agent in release 7.0.0), you may have to remove the MDS file that is located in the mount point directory of the file system in which GuardPoint resides. Before you remove the MDS file, make sure that the MDS file is not in use with other GuardPoints under the same file system mount point. For example, if you removed the metadata for GuardPoint `/oxf-fs1/gp-1` where `/oxf-fs1` is the mount point of the mounted file system, make sure that `/oxf-fs1` does not appear in the output of the `secfsd -status guard` command, and then proceed:

1. Remove the MDS file from the mount points or GuardPoints. Starting with VTE version 6.1 and continuing with all CTEversions, CTE-LDT always creates and manages the MDS file inside the GuardPoint directory. Earlier versions created and managed the MDS file inside mount point directory while sharing the MDS file with all the GuardPoints under the mount point. Type:
   ```
   # voradmin ldt rmstore <mount_point>
   ```
   For example:

   ```
   # voradmin ldt rmstore /oxf-fs1
   Enter YES if /oxf-fs1 does not include any GuardPoints associated with an CTE-LDT
   policy ->YES
   MDS file /oxf-fs1/::vorm:mds:: has been removed.
   ```

2. Verify that the metadata store has been removed from the secfs mount points, type:

   ```
   # ls -altr <file_system_mount_point>
   ```

   For example:

   ```
   # ls -altr /oxf-fs1
   ```

   You should not see `/oxf-fs1/::vorm:mds` listed.

Deleting the LDT Private Space Directory for NFS Shares

As described in "LDT Metadata Management Over NFS/CIFS Shares" on page 73, CTE-LDT creates and manages LDT NFS metadata in the LDT Private Space Directory inside the GuardPoint. To remove the LDT Private Space Directory and the files it contains, use the `voradmin ldt rmldt <GuardPoint>` command. For example:

```
# voradmin ldt rmldt /nfs-oxf-fs1/gp1
Enter YES if GuardPoint /nfs-oxf-fs1/gp1 is no longer associated with an LDT policy ->
YES
LDT metadata on /nfs-oxf-fs1/gp1 has been removed.
```

Deleting CTE-LDT Metadata (Windows)

Use the following command to delete the CTE extended attribute for a file or a GuardPoint. This is useful when removing a GuardPoint from under an CTE-LDT policy (see "Migrating a GuardPoint Out of CTE-LDT" on page 96).

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

For an overview of `voradmin ldt`, see "CTE-LDT Command-Line Administration: voradmin command" on page 92.

Removing CTE-LDT from a Host

Once you have registered a host and enabled CTE-LDT, you cannot disable the CTE-LDT feature by unchecking the CTE-LDT box. You must unregister the host from the DSM or CipherTrust Manager, then register it again without CTE-LDT. When you remove the CTE-LDT feature from a host entirely, the host's CTE-LDT license becomes available for use on another host.

1. Stop all applications from accessing data in CTE-LDT GuardPoints on the host.
2. Migrate data in every CTE-LDT GuardPoint using the steps described in the section "Remove Protection from a GuardPoint" on page 97.

   WARNING
   Potential data loss. Ensure that you have decrypted the data and, optionally, copied it out of the GuardPoint. Once the CTE Agent software is removed, access to data is no longer controlled by CTE. If the data was encrypted, it remains encrypted, and there is no way to read it.

3. Remove the GuardPoints on the host from the DSM or CipherTrust Manager.
   a. Remove the CTE-LDT metadata from those GuardPoints.
   b. Remove the MDS files associated with those GuardPoints, if necessary. See "Deleting CTE-LDT Metadata (Linux)" on the previous page for more information.
4. Remove the host from the DSM or CipherTrust Manager. For details, see the *CTE Agent for Linux Advanced Configuration and Integration Guide* or the *CTE Agent for Windows Advanced Configuration and Integration Guide*.
5. Re-install the agent on the host.
6. Register the host with the DSM or CipherTrust Manager. This time, do not select the CipherTrust Transparent Encryption - Live Data Transformation option. See "Enabling CTE-LDT on a Protected Host" on page 31.

Uninstalling the Agent while CTE-LDT is Rekeying GuardPoints

You cannot uninstall the CTE Agent if CTE-LDT is rekeying any GuardPoints on that host/client .

If you want to uninstall the CTE Agent from a host/client that contains CTE-LDT GuardPoints, you must first unencrypt the files on each CTE-LDT GuardPoint using the steps noted in the previous sections.

Chapter 6: Troubleshooting CTE-LDT

The CTE-LDT administrator needs to monitor and respond to runtime statistics and alerts.

Monitoring and Statistics

This section describes how to find the runtime status of CTE-LDT and how to get statistics about its operation.

> **Note**
> When you check GuardPoint status on the DSM or CipherTrust Manager, be aware that the status is not relayed in real time. There can be a delay of several minutes before the DSM or CipherTrust Manager becomes aware of events on the host. When you configure the host and v for one-way communication, the delay is longer, up to an hour, because CTE-LDT sends statistics to the DSM or CipherTrust Manager only once per hour.

To find out when the DSM or CipherTrust Manager last received a status update from the host, check the Last Status Update timestamp in the GuardPoint Status dialog (for information about how to display this dialog, see "Obtaining Statistics in the DSM or CipherTrust Manager with GuardPoint Status" below). For the most up-to-date statistics, inspect the host/client . Run the `voradmin` command on the managed hosts to retrieve status and statistics pertaining to GuardPoints.

To find out whether your host is configured for one-way communication, open the DSM Management Console, choose **Hosts > Hosts**, click the name of the host, and look for the FS Agent One Way Communication checkbox.

Obtaining Statistics in the DSM or CipherTrust Manager with GuardPoint Status

During and after data transformation, you can view statistics about any GuardPoint on any host in your system. Either use the DSM Management Console, as described in this section, or use the `voradmin` command, as described in "Obtaining CTE-LDT Statistics at the Command Line" on page 104.

To display the GuardPoint status in the DSM Management Console:

1. In the DSM Management Console, choose **Hosts > Hosts**.
2. Click the name of the host you want to inspect.
3. Click the **GuardPoint** tab.
4. In the Status column, click the icon.

   The GuardPoint Status window opens. The window displays the following values:

GuardPoint Status Field	Description
Reason	If the key rotation was initiated manually on an CTE-LDT host, the requester selected a reason such as “normal maintenance” or “key compromised.”
GuardPoint Status	Status of the GuardPoint.
GuardPoint State	State of the GuardPoint.
Usage	Displays the reference count for the GuardPoint.
Guarded on	Date when the policy went into effect on this GuardPoint.
Policy Name	Name of the policy.
Policy Version	The current policy version. If you make any changes to a policy after creating it, the policy version number increases by 1. The first version is number 0.
Last Status Update	Time stamp that displays when the DSM last received status information from the agent on the protected host. There can be a delay between status updates from agent to DSM. If it is imperative that you see the most recent data, go to the agent and use the voradmin command to inspect it. For an overview of voradmin CTE-LDT, see "CTE-LDT Command-Line Administration: voradmin command" on page 92.
Policy Aggregate Key Version	Indicates the version of the key used to transform data.
Policy ID	A unique policy ID within the target domain in the Vormetric Data Security Manager (DSM)
Transformation Status	A one-word summary of the current (or most recent) CipherTrust Transparent Encryption - Live Data Transformation. The same value is shown in the Rekey Status column on the host’s GuardPoint tab. For details about each status code, see "Checking the Rekey Status" on page 52.
Last Transformation Completion Time	Timestamp from the last time that transformation completed successfully.
Last Transformation Start Time	Timestamp showing the last time transformation was started. If this time is later than the Last Transformation Completion Time, the transformation did not complete normally. The Transformation Status should reflect this, such as with Incomplete.
Estimated Rekey Completion Time	If transformation is underway, this field displays the rekey duration.
Total Number of Files Deleted	Number of files in the GuardPoint that were deleted by users while data transformation was underway.
Total Files to Be Transformed	Number of files found during the initial scan of the GuardPoint. For more information, see "CTE-LDT Runtime Flow" on page 14.
Total Bytes to Be Transformed	Size of all of the files found during the initial scan of the GuardPoint.
Total Bytes Transformed	Number of bytes in the GuardPoint that were successfully transformed in the most recent CTE-LDT operation. If this does not match Total Bytes to Be Transformed, look for Total Files Skipped or Total Files Errored.

GuardPoint Status Field	Description
Total Files Transformed	Number of files in the GuardPoint that were successfully transformed in the most recent CTE-LDT operation. If this does not match Total Files to Be Transformed, look for Total Files Skipped or Total Files Errored.
Total Files Skipped	Total number of files that CTE-LDT did not transform in this pass. The files were intended for transformation, but were not transformed for some reason. For example, (on Windows) the file is read-only, or it is an executable file that is currently running. On a Windows host, transformation continues on the next available file in the GuardPoint, and CTE-LDT maintains a list of the files that it skipped.
Total Files Errored	Total number of files that triggered alerts. See "Alerts Playbook" on page 107.

Obtaining CTE-LDT Statistics at the Command Line

You can obtain CTE-LDT statistics from the command line using the following command:

```
# voradmin ldt stats <guard path> [interval]
```

This command displays transformation statistics for a GuardPoint, or for all GuardPoints, if none are specified. If you specify an interval (in units of seconds), the command continually updates the statistics on the given interval. If no GuardPoint is specified, the command returns the aggregate statistics at the host level, including specific statistics related to QoS. If a GuardPoint is specified, statistics include:

- Current rekey status
- Rekey Start time
- Last rekey completion time
- Estimated Rekey completion time
- Data Statistics including the amount of data transformed
- File statistics including:
  - Number of files transformed
  - Number of files skipped
  - Number of files remaining for rekey

Obtaining a Rekey Report

CTE-LDT generates a report automatically after completion of a key rotation. The report lists the files in the GuardPoint and the key and version of the key applied to each file. This kind of report may be a common compliance requirement.

The administrator can also request a rekey report during rekey, if the need to see the partial transformation results so far.

About the rekey report

On Linux hosts, CTE-LDT writes rekey reports to a local file on the agent host in `/var/log/vormetric/`. The file name begins with `ldaudit-log-`. It is followed by the file system directory name, GuardPoint directory name, and a timestamp. For example, for a GuardPoint at `/oxf-fs2/gp2` with rekey completed on November 19, 2016, just after 2:00 p.m. (hour 14), the rekey report file name would be `ldtaudit-log-_oxf-fs2_gp2-2016111914917`.

The report includes:

- Total number of files in the GuardPoint
- Number of files transformed
- Rekey start and end times
- List of all files transformed
- Applied key and key version for each file (for example, files in different resource sets might have used different keys)

Manually generating a rekey report

To generate a rekey report manually, use the following command:

```
# voradmin ldt rekey report <GuardPoint> [<logfile>]
```

In `<GuardPoint>`, type the GuardPoint path. In `<logfile>`, you can optionally direct the output to a file. If no logfile is specified, the report displays on `stdout`.

Monitoring Ongoing CTE-LDT Operations at the Command Line (Windows only)

Use the following syntax to monitor CTE-LDT progress. If an interval is specified (in units of seconds), it continually updates the monitoring output at the specified interval.

```
C:\> voradmin ldt monitor [interval]
```

For an overview of `voradmin ldt`, see "CTE-LDT Command-Line Administration: voradmin command" on page 92.

Protecting CTE-LDT GuardPoints against Failure in Underlying File Systems (Linux)

CTE-LDT Recovery Challenges

The main challenge with CTE-LDT recovery is a failure to access files or specific blocks in files inside a GuardPoint. Before CTE-LDT runs the recovery process on a GuardPoint, the underlying file system was recovered before file system was mounted. File system recovery may create orphan files in the `lost+found` directory inside the mount point. If an orphan file belongs to the GuardPoint, CTE-LDT cannot recover the file as the file is no longer in the same directory as it was prior to the crash while it was undergoing rekey.

CTE-LDT Recovery Enhancement

CTE-LDT has been improved for handling inconsistencies in the underlying file system during execution of CTE-LDT recovery. When enabling a GuardPoint after a system crash, CTE-LDT performs consistency checks on the files undergoing rekey at the time of crash before enabling GuardPoint. If the CTE-LDT recovery process is unable to recover any of the affected files, the GuardPoint will not be enabled and CTE-LDT will send the following alert to the DSM or CipherTrust Manager:

```
[CGS3266E] LDT-ALERT: Cannot enable guard point [GuardPoint] due to inconsistencies in
underlying file system encountered during LDT recovery.
```

You can check the status of a GuardPoint and the reason for failing to enable a GuardPoint using the `secfsd -status guard` command. For example, the GuardPoint `/mnt/gp` failed to guard because `Guard point needs LDT recovery` as reported in the `Reason` column for the following example GuardPoint.

```
GuardPoint Policy Type   ConfigState Status      Reason
---------- ------ ----   ----------- ------      ------
/mnt/gp    LDT220 manual guarded     not guarded Guard point needs LDT recovery
```

Details on the specific issues encountered, including any files that could not be recovered, are reported to a log file in the `/var/log/vormetric` directory. The log file name is in the format:

```
ldt_recovery_log:<guardpoint_name>:<timestamp>.txt
```

The log file name starts with `ldt_recovery_log`, followed by the GuardPoint pathname and the date and time of the recovery attempt. The GuardPoint pathname and date and time are separated with "`:`". For example:

```
/var/log/vormetric/ldt_recovery_log:_mnt_gp:2018-12-08-14:6:12.txt
```

Refer to the issues listed in the log file, and resolve those issues before enabling the GuardPoint again. A GuardPoint cannot be enabled in subsequent guard attempts until those issues are resolved. Each attempt generates a new log file. Check the CTE-LDT log file for missing files whose inode numbers match inode numbers of orphan files in `lost+found`. If there is a match, restore the file from `lost+found` to the pathname specified in the CTE-LDT recovery log file. Be sure to run the `mv` command to restore the file to the original location, do not run the `cp` command. After resolving all or some of the reported issues, you must run the command `voradmin ldt recover <GuardPoint>` to repeat the recovery process.

```
voradmin ldt recover <GuardPoint>
```

Running this command resolves the problems that can be corrected and clears the failed recovery status on the GuardPoint, allowing the GuardPoint to enable automatically within 30 seconds. If the GuardPoint does not enable, you can enable it using `secfsd -guard` command, if manual guard, or enable it on the DSM or CipherTrust Manager, if auto-guard. Note that if you are unable to resolve all of the reported issues, you accept the loss of some data or files, as reported in the latest recovery log file for the GuardPoint, when you run the command `voradmin ldt recover <GuardPoint>`.

Recovery Alerts

CTE-LDT sends two alerts to DSM or CipherTrust Manager if CTE-LDT fails to resolve the issues encountered during recovery.

The alert below is an error report sent to the DSM or CipherTrust Manager whenever CTE-LDT encounters failure during recovery prior to enabling GuardPoint. This alert also reports the GuardPoint specified in the message is not enabled:

```
[CGS3266E]  LDT-ALERT: Cannot enable guard point [GuardPoint] due to inconsistencies in
underlying file system encountered during LDT recovery.
```

The alert below is a warning report sent to the DSM or CipherTrust Manager only if `voradmin ldt recover` encounters errors and continues anyway. This alert reports the GuardPoint specified in the message has been enabled with some files in error status.

```
[CGS3267W]  LDT-ALERT: LDT manual recovery on guard point [GuardPoint] completed with
[#] errors.
```

The full message body does not currently appear in the logs on the DSM or CipherTrust Manager. At this time, the DSM or CipherTrust Manager shows the message ID without the message body.

Alerts Playbook

This section lists the CTE-LDT alerts and describes what to do in each case. CTE-LDT generates alerts when issues arise that require attention from the Administrator . Without prompt attention, alerts can delay the rekey process or cause the process to remain incomplete.

Failure to Enable GuardPoint Due to Incorrect Policy

The following message indicates that there was an attempt to apply an LDT policy to a GuardPoint that was previously guarded with a different LDT policy:

- **LDT-ALERT: Cannot enable GuardPoint due to wrong policy applied to [GuardPoint]**

  The GuardPoint status changes to "incorrect policy".

Solution: Unguard the GuardPoint and re-guard with the correct policy or, if you want to change the policy associated with the GuardPoint, clean up the LDT metadata and the re-guard. For details on cleaning up the GuardPoint, see "Deleting CTE-LDT Metadata (Linux)" on page 99.

Failure to Suspend or Resume CTE-LDT Operation

- **LDT-ALERT: Failed to suspend rekey on GuardPoint [GuardPoint]**

  Solution: An I/O error is the most common cause of failure when updating the persistent state of a GuardPoint. For I/O errors, you must fix the problem at the host OS or storage level.

  The appropriate corrective action depends on when this problem occurs and the reason for the suspension. If the suspend request was at the host level as part of a QoS schedule, or it was initiated by a user on the DSM or CipherTrust Manager, the failure occurred at the host level rather than the specified GuardPoint.

  Otherwise, the source is probably the initiation of a suspension on a specific GuardPoint before backup. If the backup operation is in progress when this alert message occurs, you must fix the cause of the suspension failure and then restart the backup. If the backup has already completed when the alert message occurs, the backup image on the GuardPoint may have inconsistent data and CTE-LDT metadata. Discard this backup image and do a fresh backup.

  If you cannot find and fix the host OS or storage issue, contact Thales Support for troubleshooting and recovery.

- **LDT-ALERT: Failed to resume rekey on GuardPoint [GuardPoint]**

- **LDT-ALERT: Failed to resume rekey on all GuardPoints**

  Solution: An I/O error is the most common cause of failure when updating persistent state of a GuardPoint. For I/O errors, you must fix the problem at the host OS or storage level.

  The appropriate corrective action depends on when this problem occurs and the source of the problem. If the resume request was at the host level as part of a QoS schedule, or the resume request was initiated by a user on the DSM or CipherTrust Manager, the failure occurred at the host level rather than the specified GuardPoint.

  Otherwise, the source is probably initiation of a resume operation on a specific GuardPoint after a backup is completed.

  If you cannot find and fix the host OS or storage issue, contact Thales Support for troubleshooting and recovery.

Failure to Take Over CTE-LDT Operations on an NFS Host

LDT-NFS-ALERT: Host waiting to take over NFS GuardPoint [GuardPoint] for LDT

Solution: Unguard the GuardPoint for the directory specified in the message from hosts other than this one. CTE-LDT operations for this GuardPoint are being handled by this host and cannot continue until this host has exclusive access to the GuardPoint. A list of hosts who have the GuardPoint guarded can be found by looking at the list of tag files in the private space. For details, see "CTE-LDT Host Tag Files" on page 78.

Insufficient Resources

The following alerts trigger when there are not enough resources for CTE-LDT operations.

- **LDT-ALERT: Aborting rekey of file [FileName] due to insufficient disk space**

  The key rotation process was stopped on a GuardPoint because there was not enough available disk space.

  Solution: Resize or free up space in the file system where the PathName resides. Key rotation automatically starts again when sufficient free space becomes available.

- **LDT-ALERT: Failed to launch transformation on [GuardPoint]**

  CTE-LDT failed to launch the data transformation process on the specified GuardPoint.

  Solution: The most common causes of this failure are:

  - Low amount of system resources, such as free disk space.
  - Check the system logs for insufficient free space in underlying file systems. You can free up space by removing older files or resizing your underlying file system. You may resize the file system on-line or off-line.

  For other issues, contact Customer Support.

- **LDT-ALERT: QoS failed to start**

  The Quality of Service subsystem failed to launch QoS services due to lack of system resources.

  Solution: A low amount of available memory is the most common cause of this failure.

  Contact Customer Support.

- **LDT-ALERT: Skipped key rotation on GuardPoint [GuardPoint] due to insufficient disk space**

- **LDT-ALERT: MDS file [PathName] exceeded diskspace quota on gp [GuardPoint]**

  There was not enough storage space in one or more file systems containing GuardPoints undergoing rekey.

  Solution: Check the file systems where your GuardPoints reside to see how much space it is using. Add more storage space as needed. After the condition is corrected and disk space is available, CTE-LDT operations resume automatically.

- **LDT-ALERT: Low space on guard point [GuardPoint], increase free space or LDT will be suspended.**

  Available storage space in the file system containing the GuardPoint is nearing the threshold below which CTE-LDT rekeying cannot continue. If available space does drop below that threshold, rekeying will be automatically suspended.

  Solution: Check the file system where your GuardPoint resides to see how much space is available. Make more free space available if necessary. After the condition is corrected and storage space is available, rekeying resumes automatically.

Failed to Update CTE-LDT Attribute

The following alerts trigger when CTE-LDT extended attributes are not properly updated.

- **LDT-ALERT: Failed to update LDT attribute on GuardPoint [GuardPoint]. Error: [ErrorNumber]**

  Solution: I/O error is the most common cause of failure when updating CTE-LDT extended attributes. For I/O errors, fix the problems at the host OS or storage level. In Linux, if you experience an I/O error, identify the file that had the error, and restore it from backup. See "Backing Up and Restoring CTE-LDT GuardPoints" on page 82.

  In Windows, you can recover a file that has corrupt metadata as follows:

  - If the file is fully encrypted, remove the metadata. Type:

    ```
    # voradmin ldt attr delete <file name path>
    ```

    Then apply the same policy that was used to encrypt the data.
  - If the file was only partially rekeyed, restore the file from a backup.

  If you cannot find and fix the underlying host OS or storage issue that is causing corrupt metadata, contact Customer Support for troubleshooting and recovery.

- **LDT-ALERT: Failed to update LDT attribute**

  CTE-LDT failed to create or update an CTE-LDT extended attribute of a file in a GuardPoint.

  Solution: An I/O error is the most common cause of failure when updating an CTE-LDT extended attribute. For I/O errors, fix the problem at the host OS or storage level.

  If you cannot find and fix the host OS or storage issue, contact Customer Support for troubleshooting and recovery.

Rekey Stopped

The following alert is specific to Windows and is triggered when rekey is stopped before it completes normally.

LDT-ALERT: Suspending rekey of binary file [PathName] during its execution

CTE-LDT operations encountered an executable file that is running.

CTE-LDT cannot rekey a running executable file until the execution of the binary file stops.

Solution (Windows)

1. Obtain the name of the executable file from the alert message.
2. Stop execution of the binary file.

Incomplete Key Rotation

- **LDT-ALERT: Failed to rotate key on GuardPoint [GuardPoint]. Another rekey operation is already in progress** (Windows only)
- **LDT-ALERT: Key rotation failed on GuardPoint [GuardPoint]**

CTE-LDT failed to complete the key rotation process on the specified GuardPoint. On Windows, this may occur if a key rotation process is already in progress on the GuardPoint. This pre-existing rekey operation could be active, or it could be in a suspended state, either because of the QoS schedule or a manual pause initiated by the administrator. See "Suspending and Resuming Rekey and/or Scan Phase" on page 54.

Solution

1. On Linux, you can run the following command to see whether key rotation is in progress on the GuardPoint.

   ```
   # voradmin ldt list all
   ```

2. On Linux or Windows, check the rekey status of GuardPoints in the DSM Management Console.
3. If key rotation is already in progress on a Windows GuardPoint, wait for the key rotation to complete.
4. On the **GuardPoint** tab on the DSM, select **Re-push Policies** to manually re-push the policies to the host to initiate key rotation on GuardPoints ready for key rotation.

Contact Customer Support if this error occurs on Linux or if the cause of key rotation failure on Windows is not a key rotation in progress.

Skipped Key Rotation

The following alerts trigger when CTE-LDT skips key rotation.

- **LDT-ALERT: Skipped key rotation on GuardPoint [GuardPoint]. It is on a read-only file system**

  Key rotation is skipped on a specified GuardPoint because the mount point where the GuardPoint resides does not permit write operations.

  Solution:

  1. Remount the file system where the GuardPoint resides, and change the mount option to read/write.
  2. On the GuardPoint page on the DSM or CipherTrust Manager, press **Re-push Policies** to manually re-push the policies to the host to initiate key rotation on GuardPoints ready for key rotation.

- **LDT-ALERT: Failed to rotate key on GuardPoint [GuardPoint] during pre-commit**
- **LDT-ALERT: Failed to rotate key on GuardPoint [GuardPoint] during commit**
- **LDT-ALERT: Skipped key rotation on GuardPoint [GuardPoint]. Error: [ErrorNumber]**
- **LDT-ALERT: Failed to update LDT attribute on GuardPoint [GuardPoint]. Error: [ErrorNumber]**

  CTE-LDT failed to start a key rotation process on a GuardPoint during a guard operation or when processing a key rotation notification from the DSM or CipherTrust Manager. For [*ErrorNumber*], a Linux error number is substituted, such as `errorcode 17`.

  Solution: The host returns error code 17 during CTE-LDT key rotation if it cannot perform the key rotation because there is already a rekey in progress. This pre-existing rekey operation could be active, or it could be in a suspended state, either because of the QoS schedule or a manual pause initiated by the administrator. See "Suspending and Resuming Rekey and/or Scan Phase" on page 54.

  An I/O error is the most common cause of failure when updating the persistent state of a GuardPoint. For I/O errors, fix the problem at the host OS or storage level.

  If you cannot find and fix the host OS or storage issue, contact Customer Support for troubleshooting and recovery.

Failed to Update CTE-LDT Metadata During Scan Phase

CTE-LDT triggers the following alerts when CTE-LDT cannot properly update metadata during the scan phase. (See "CTE-LDT Runtime Flow" on page 14 for information about the scan phase.)

- **LDT-ALERT: Scan error on [*GuardPoint*] removing MDS guard for relaunching dataxform**

  CTE-LDT failed to update metadata when restarting a scan on the specified GuardPoint.

  Solution: An I/O error is the most common cause of failure when updating CTE-LDT metadata. For I/O errors, fix the problem at the host OS or storage level and then restart operations.

  If you cannot find and fix the underlying host OS or storage issue that is causing the error, contact Thales Support for troubleshooting and recovery.

- **LDT-ALERT: Online Dataxform failed during post scan stage on [*GuardPoint*]**

  CTE-LDT failed to update metadata when transitioning from scan to rekey phase on the specified GuardPoint.

  Solution: An I/O error is the most common cause of failure when updating CTE-LDT metadata. For I/O errors, the problem must be fixed at the host OS or storage level and the operation restarted.

  If you cannot find and fix the underlying host OS or storage issue that is causing the error, contact Customer Support for troubleshooting and recovery.

File system inconsistencies after system crash

CTE-LDT sends two alerts to the DSM or CipherTrust Manager if CTE-LDT fails to resolve the issues encountered during recovery.

The alert below is an error report sent to the DSM or CipherTrust Manager whenever CTE-LDT encounters failure during recovery prior to enabling GuardPoint. This alert also reports the GuardPoint specified in the message is not enabled:

- **LDT-ALERT: Cannot enable guard point [GuardPoint] due to inconsistencies in underlying file system encountered during LDT recovery**

  The alert below is a warning report sent to the DSM or CipherTrust Manager only if `voradmin ldt recover` encounters errors and continues anyway. This alert reports the GuardPoint specified in the message has been enabled with some files in error status.

- **LDT-ALERT: LDT manual recovery on guard point [GuardPoint] completed with [#] errors.**

  Solution: See "Protecting CTE-LDT GuardPoints against Failure in Underlying File Systems (Linux)" on page 105.

Error Messages

This section describes runtime error messages. For information about other types of runtime messages, see "Alerts Playbook" on page 107 and "Warning and Info Messages" on page 114.

Failed to Transform File During Rekey

CTE-LDT could not complete transformation on a file.

Related Messages:

- **LDT: Rekey failed for file [*PathName*] on GuardPoint [*GuardPoint*]**
- **LDT: Extended attribute of inode [*InodeNumber*] is corrupted under GuardPoint [*GuardPoint*]**

Solution: An I/O error is the most common cause of failure when updating CTE-LDT metadata. For I/O errors, fix the problem at the host OS or storage level, and then restore the file from a backup.

If you cannot find and fix the underlying host OS or storage issue that is causing the error, contact Customer Support for troubleshooting and recovery.

Failure to Suspend CTE-LDT

A request to stop CTE-LDT processing did not succeed. The suspend request was at the host level as part of a QoS schedule, or the suspend request was initiated by a user on the DSM or CipherTrust Manager.

Related Messages:

- **LDT: Failed to suspend rekey on all GuardPoints**
- **LDT operations could not be suspended on the host**

Solution: An I/O error is the most common cause of failure when updating the persistent state of a GuardPoint. For I/O errors, fix the problem at the host OS or storage level.

If a backup operation is in progress when this message occurs, you must fix the cause of the suspend failure and then restart the backup. If the backup has already completed when the alert message occurs, the backup image on the GuardPoint may have inconsistent data and CTE-LDT metadata. Discard this backup image and do a fresh backup.

If you cannot find and fix the host OS or storage issue, contact Customer Support for troubleshooting and recovery.

Failure to Start or Stop Transformation

The following general messages are recorded when there are errors attempting to start or stop CTE-LDT:

- **LDT: Failed to abort key rotation on GuardPoint [*GuardPoint*]**
- **LDT: Failed to start**
- **LDT: Failed to stop**
- **LDT: Failed to exit**

Solution: Examine system logs for additional information as to the cause. If you cannot find and fix the underlying host OS or storage issue that is causing the error, contact Customer Support for troubleshooting and recovery.

Failure to Restart Transformation

The following message is recorded when an attempt to restart transformation after a system reboot fails because the file system is mounted as read-only. Transformation on the specified GuardPoint cannot continue until the file system is mounted with write permission.

- **LDT: Skipped LDT recovery on read-only file system [*GuardPoint*]**

  Solution: Re-mount the file system with write permissions.

Failure to Schedule Relaunch

The following message indicates that a rekey request was sent to a Linux GuardPoint that was already undergoing data transformation, and an error was encountered when CTE-LDT attempted to defer the rekey request until after the current data transformation completes:

- **LDT: Failed to flag GuardPoint [GuardPoint] for deferred key rotation, error [Error]**

  Solution: In your key manager, repush the policy to the host.

Temporary Failure to Start Transformation on a File

The following messages are recorded when there are communication issues or lack of resources on the host. Once the condition is corrected, transformation of the file continues automatically:

- **LDT: Insufficient memory condition encountered during LDT on GuardPoint [*GuardPoint*]**
- **LDT: Encryption key for file [*PathName*] unavailable for LDT, possibly due to loss of communication to DSM or CipherTrust Manager**
- **LDT: Aborting rekey of file [*PathName*] due to lack of free memory. Try closing other application to resolve the issue**

Solution: Resolve the communication issues or increase the available resources, then verify that CTE-LDT has resumed processing.

Transient Condition while enabling GuardPoint

The following messages are recorded when a request is made to guard an CTE-LDT GuardPoint when GuardPoint initialization is already in progress. During the GuardPoint initialization, the system initializes MDS file associated with the GuardPoint in preparation for CTE-LDT. Initialization of the MDS file can take a few minutes. During this time, if the system retries the guard operation, one of the following messages displays.

Related Messages:

- **Not re-guarding path [*path*] (Reason: GuardPoint initialization already in progress)**
- **Not re-guarding path [*path*] from container [*container*] (Reason: GuardPoint initialization already in progress)**

Solution: Wait for the current GuardPoint initialization to complete and then resubmit the new GuardPoint initialization request if desired.

Transient Failure to Read LDT Attributes from NFS

NFS may incorrectly returns 0's when reading LDT metadata from files in NFS shares. As CTE-LDT doesn't expect 0's returned for LDT attributes, it retries the read operation, and the second operation succeeds and returns valid LDT attribute.

The first read attempt returning 0's results in the first warning message logged in the system log file as LDT attribute validation fails. However, the second read attempt succeeds and reads valid LDT attribute data, resulting in the second message logged in response to the first message. You can ignore the first warning message if both messages appear together in system log.

Failed to transform passthrough files for AD database files (Windows Only)

CTE-LDT skipped the transformation of passthrough files for all AD database files. The problem occurs when the AD database remains in the default folder location.

Related Messages:

- **Skipping transformation of passthrough file [*filename*]. The file resides in the boot directory. CTE cannot encrypt boot directory files.**
- **Skipping transformation of passthrough file [*filename*]. CTE cannot encrypt these files. They are already encrypted using NTFS encryption or compressed.**
- **Skipping transformation of passthrough file [*filename*]. File is not in a GuardPoint directory.**

Solution: To fix this issue, move the AD database to any folder other than `c:\windows` or `c:\program files`.

Warning and Info Messages

This section describes runtime error messages. For information about other types of runtime messages, see "Alerts Playbook" on page 107 and "Error Messages" on page 111.

Stopping Transformation of a File on Volume Dismount (Windows only)

The following warning message is recorded when transformation of a file is stopped because the containing volume is dismounted.

- **LDT: Volume dismounted. Aborting transformation at file [*PathName*]**

Issues with Policy or System Configuration

The following warning messages are recorded when CTE cannot perform CTE-LDT on the GuardPoint, because there are errors in the policy associated with the GuardPoint, or the file system containing the GuardPoint is not supported by CTE-LDT. Files in the GuardPoint can still be accessed, but no CTE-LDT encryption occurs.

- **LDT: The GuardPoint [*GuardPoint*] does not have a valid transformation policy, there is no new key rule**
- **LDT: The GuardPoint [*GuardPoint*] does not have a valid transformation policy, there is no key_op rule**
- **LDT: The GuardPoint [*GuardPoint*] does not support online rekey, only file operations from the offline data transform process will be allowed**

Failure to Enable GuardPoint During Cleanup

CTE-LDT records the following informational message when a user attempts to enable a newly added GuardPoint. This message displays if the GuardPoint directory was previously guarded with an CTE-LDT policy and the CTE-LDT metadata cleanup is in progress when user is guarding under the new policy.

- **LDT: Cannot enable GuardPoint [*GuardPoint*] during LDT clean-up process**

Solution: Retry the operation once the cleanup completes.

General CTE-LDT Operations

The following informational messages are recorded during various CTE-LDT operations. No action is required.

- **LDT: Successfully suspended rekey on GuardPoint [*GuardPoint*]**
- **LDT: Successfully suspended rekey on all GuardPoints**
- **LDT: Successfully resumed rekey on GuardPoint [*GuardPoint*]**
- **LDT: Successfully resumed rekey on all GuardPoints**
- **LDT: Rekey operation completed on GuardPoint [*GuardPoint*]**

Missing CTE-LDT extended attribute

The following warning message reports that the file with the specified inode number in the specified GuardPoint directory does not have an CTE-LDT extended attribute, therefore, access to the file is denied.

- **LDT: Extended attribute of inode [*InodeNumber*] is missing under GuardPoint [*GuardPoint*]**

  Solution:CTE-LDT cannot determine the encryption key associated with the data in the file, therefore, you can only remove the file.

Locking Contention

The following messages are recorded during the rekey process on a file. When user access to the file is very high, it causes a high degree of locking contention between the rekey process and user access. The second message reports when the contention is no longer in effect and the rekey process has resumed accessing file to rekey.

- **LDT: Exclusive access for rekey delayed on inode [*InodeNumber*]**
- **LDT: Exclusive access for rekey granted after delay on inode [*InodeNumber*]**

Initiation and completion of CTE-LDT metadata cleanup

The following messages are recorded at the beginning and completion of CTE-LDT metadata cleanup through voradmin command.

- **LDT: Metadata will start getting removed from all files in GuardPoint [*GuardPoint*]**
- **LDT: Metadata has been removed from all files in GuardPoint [*GuardPoint*]**

Upgrading CTE Agent

On Windows, you cannot upgrade the CTE agent if there is an CTE-LDT GuardPoint with an active rekey in progress, that is at the mount point. On Linux, CTE-LDT is forced-suspended when upgrading CTE.

Reboot is required when upgrading earlier versions of CTE Agent to 6.0.1, if CTE-LDT is in-progress on a GuardPoint. Perform following steps to upgrade to version 7.1.1:

1. From the DSM Management Console, click **Host > Hosts > *hostName* > GuardPoint** tab.
2. Click **Suspend Rekey**.
3. Click **Refresh** and ensure that the GuardPoint(s) being rekeyed now show a status of Suspended.
4. Reboot the host.
5. Run the 7.1.1 CTE Agent install program and upgrade the agent.

   Wait for the upgrade to complete.
6. From the DSM Management Console, click **Host > Hosts > *hostName* > GuardPoint** tab.
7. Click **Resume Rekey**.
8. Click **Refresh** and ensure that the GuardPoints being rekeyed now show a status of Rekeying.

Recommendations and Considerations

The following information provides guidance for a better user experience.

All Platform Recommendations and Considerations

Binary Re-signing

Any executable that is part of either a Signature set or a Host setting, and that resides in a GuardPoint that uses an CTE-LDT policy, will use different signatures for an CTE-LDT key rotation. The result is that the Host Settings binaries will no longer be authenticated, or that the Signature Set policy rules will no longer trigger for those binaries.

To prevent these issues, the Security Administrator must manually re-sign each affected binary after each key rotation.

Alternatively, CTE can generate unencrypted signatures of binaries inside GuardPoints to avoid these problems. For details, see the *DSM Administration Guide*.

Check for available disk space for CTE-LDT metadata

Before launching CTE-LDT on a GuardPoint:

1. Check the available free disk space in the file system where your GuardPoint resides.
2. Type the following command to check the disk space requirement of CTE-LDT on a target GuardPoint:

   ```
   # voradmin ldt space /oxf-fs1/gp1
   /oxf-fs1/gp1: found 1501 files without LDT extended attributes
   LDT disk space requirements: total 169MB (LDT attributes=6MB, MDS=163MB)
   ```

   In this example, the output shows that CTE-LDT requires 169MB of available disk space to launch and execute CTE-LDT on `/oxf-fs1/gp1`.

   > **Note**
   > Make sure that the free space in your file system exceeds the disk space requirement for CTE-LDT.

CTE-LDT Requirements for Backup

CTE offers the option to backup encrypted data from files through a security rule, which skips the Apply Key function when reading encrypted files by backup applications or processes. Such files can also be restored from backup streams, to the same or a different GuardPoint, without Apply Key. If you choose to restore to another GuardPoint, the target GuardPoint must be protected with the same key and security rules as the source GuardPoint.

If you backup encrypted data from files inside GuardPoints with CTE-LDT policies, CTE-LDT imposes a hard requirement on the backup application, or process, to backup or restore CTE-LDT metadata. The CTE-LDT metadata is stored as an extended attribute on Linux and as alternate data streams on files on Windows platforms. If the backup process cannot backup the metadata, then CTE-LDT protected GuardPoints must backed up in clear key.

Customers are required to verify their backup application, or process, to ensure extended attributes (on Linux) or alternate data streams (on Windows) are backed up and restored through their backup/restore processes.

> **Note**
> Make sure that you suspend CTE-LDT operations before you start the backup process, and resume them after the backup process completes.

Learn Mode

Learn Mode provides a temporary method for disabling the blocking behavior of regular CTE or CTE-LDT policies. While useful for quality assurance, troubleshooting, and mitigating deployment risk, Learn Mode is not intended to be enabled permanently for a policy in production. This prevents the policy Deny rules from functioning as designed

in the policy rule set.

Ensure that the policy is properly configured for use in Learn Mode. Any Security Rule that contains a Deny effect must have Apply Key applied as well. This is to prevent data from being written in mixed states, resulting in the loss of access or data corruption.

> **Note**
> Apply Key will have no effect when combined with a Deny rule unless the policy is in Learn Mode.

Quality of Service (QoS)

By default, the QoS component of CTE-LDT does not monitor live transformation operations unless you enable QoS on your host by entering QoS parameters on the DSM. To avoid overhead of CTE-LDT on your production system, you must select QoS parameters suitable to your production environment. For information on tuning QoS, see "Quality of Service" on page 37. It is critical that you understand the impact of CTE-LDT on your system and how to manage this impact using QoS.

Upgrade to CTE 7.1.1

Since the the first release of the CTE-LDT feature, Thales has made several improvements in the areas of error handling, interoperability with applications, and Quality of Service (QoS). Thales strongly recommends that new deployments use version 7.1.1 (or later) of CTE. Thales further recommends that customers who have already deployed the CTE-LDT feature upgrade to version 7.1.1.

Windows Recommendations and Considerations

File Handling

The CTE-LDT process is subjected to all of the File System policies and attributes set on the files. In some cases, this prevents CTE-LDT from encrypting a file. If users or applications are accessing files while CTE-LDT is in progress, CTE-LDT cannot change the attributes of the files and encrypt the file. It is critical that you understand how CTE-LDT handles various types of files:

- **NTFS Encryption and Compression**

  If NTFS encryption or compression is enabled on a file or folder, the CTE-LDT process cannot encrypt these files. To maintain the data coherency, CTE-LDT skips the encryption of the these files. These files display as “passthrough” files in the CTE-LDT statistics.

- **Read-Only Files**

  As the CTE-LDT process performs a read-encrypt-write operation on a file, it cannot encrypt read-only files. The CTE-LDT process skips these files and changes to the INCOMPLETE state.

- **Executable Files**

  If a executable is running or files are exclusively locked by the application, the CTE-LDT process cannot encrypt those files as it is unable to acquire the required locks on the files. CTE-LDT skips these files and changes to the INCOMPLETE state.

File Modification

The CTE-LDT process performs a read-encrypt-write operation on the files that need to be encrypted, (also known as rekeying). Previously, file modification and access dates were changed when CTE-LDT was processing. In order

to maintain compatibility for applications, we addressed this issue by saving a copy of the original access and modification times, and restoring them after the encryption completed. Preserved timestamps are updated during the rekey process, if an application/user accesses the files during rekey.

> **Note**
> Thales strongly recommends that you upgrade to the newly released v6.1.0 so that the access time and modification time is restored correctly.

Logical Sector Size

CTE-LDT Windows transformation is supported if the **logical sector size** is more than 512 Bytes. (A logical sector size of 4K is supported.) To find the logical sector size of the file system, type:

```
> fsutil fsinfo ntfsInfo <volume pathname>
```

For example:

```
> fsutil fsinfo ntfsInfo C:
NTFS Volume Serial Number :        0x5092568a92567506
NTFS Version   :                   3.1
LFS Version    :                   2.0
Number Sectors :                   0x000000001c004eeb
Total Clusters :                   0x00000000038009dd
Free Clusters  :                   0x00000000008bb274
Total Reserved :                   0x0000000000001864
Bytes Per Sector  :                512
.
.
.
```

Upgrade Notes

> **CAUTION**
> **Do not upgrade the Windows agent without valid DSM or CipherTrust Manager connectivity.**

During the upgrade, InstallShield cleans up the agent configuration on the host. If the agent does not have DSM or CipherTrust Manager connectivity, then it cannot pull the configuration from the DSM or CipherTrust Manager after it reboots. As the configuration is cleaned up, the Windows agent removes all of the GuardPoints from the host.

VSS Volumes

If a Shadow Copy volume is present on a volume before CTE-LDT starts, and this volume is accessed by the application or user, then CTE reads the CTE-LDT metadata from the shadow copy volume snapshot and incorrectly uses this metadata for the original file. Because the metadata on the snapshot is old, this may lead to double encryption of the files inside the GuardPoint.

Thales highly recommends that, if you are running a version of VTE earlier than 6.0.3.68, you must delete all of the old VSS volumes where CTE-LDT GuardPoints exist in order to avoid this issue.

THALES

Contact us
For office locations and contact information,
visit cpl.thalesgroup.com/contact-us

> cpl.thalesgroup.com <