

*IBM Tape Device Drivers and Diagnostic Tool  
User's Guide*



---

# Contents

## **IBM Tape Device Drivers and Diagnostic Tool User's Guide..... 1**

### **IBM Tape Device Drivers and Diagnostic Tool User's Guide..... 1**

Introduction.....	2
Common extended features.....	4
Path failover and load balancing.....	4
Dynamic Runtime Attributes.....	9
Data encryption.....	9
Recommended access order (RAO) open function .....	12
AIX Tape and Medium Changer device driver.....	12
Product requirements.....	13
Installation and configuration instructions.....	13
Tape drive, media, and device driver parameters.....	16
Special files.....	22
Persistent Naming Support.....	24
Control Path failover support for tape libraries.....	25
Data Path failover and load balancing support for tape drives.....	27
System-managed encryption.....	29
Problem determination.....	30
Tape drive service aids.....	36
Performance considerations.....	38
Linux Tape and Medium Changer device driver.....	40
Product requirements.....	40
Installation and Configuration instructions.....	40
Tape drive, media, and device driver parameters.....	46
Special files.....	51
Control Path failover support for tape libraries.....	53
Data Path failover and load balancing support for tape drives.....	56
Open source device driver - lin_tape.....	59
System-managed encryption.....	61
Problem determination.....	62
Windows Tape and Medium Changer device driver.....	67
Product requirements.....	68
Installation and configuration instructions.....	68
Persistent Naming Support on Windows Server 2019, 2022 and 2025.....	71
Control Path failover support for tape libraries.....	72
Data Path failover support for tape drives.....	72
Problem determination.....	73
IBM Tape Diagnostic Tool (ITDT).....	76
Purpose.....	76
Accessing ITDT.....	77
Accessing documentation and software online.....	77
Supported systems.....	79
IBM Tape Diagnostic Tool - Standard Edition.....	80
IBM Tape Diagnostic Tool - Graphical Edition.....	151
Verifying correct attachment of your devices.....	183
Managing the microcode on the IBM tape drive.....	184
Notices.....	184
Terms and conditions for IBM Documentation.....	186

<b>Index.....</b>	<b>188</b>
-------------------	------------



# IBM Tape Device Drivers and Diagnostic Tool User's Guide

---

Welcome to the IBM Tape Device Drivers and Diagnostic Tool User's Guide documentation, where you can find information about how to install, maintain, and use IBM Tape Device Drivers and the IBM Tape Diagnostic Tool (ITDT) on multiple platforms.

Last updated: 2025-04-05.

## Installation and User's Guide

[“Introduction” on page 2](#)

The IBM tape and medium changer device drivers are designed specifically to take advantage of the features that are provided by the IBM tape drives and medium changer devices.

[“Common extended features” on page 4](#)

[“AIX Tape and Medium Changer device driver” on page 12](#)

[“Linux Tape and Medium Changer device driver” on page 40](#)

[“Windows Tape and Medium Changer device driver” on page 67](#)

[“IBM Tape Diagnostic Tool \(ITDT\)” on page 76](#)

## Troubleshooting and support

[“Verifying correct attachment of your devices” on page 183](#)

[“Managing the microcode on the IBM tape drive” on page 184](#)

[IBM tape storage support](#)

[IBM Support home page](#)

## More information

[IBM Tape Device Drivers Programming Reference](#)

[IBM Tape Device Drivers Installation and User's Guide \(Legacy\)](#)

[IBM TS4500 tape library documentation](#)

[IBM TS4300 tape library documentation](#)

[IBM TS2900 tape autoloader documentation](#)

[IBM TS2270 tape drive documentation](#)

[IBM TS2280 tape drive documentation](#)

[IBM Community \(Community platform\)](#)

[IBM Support content \(Product support\)](#)

[Redbooks home page](#)

# IBM Tape Device Drivers and Diagnostic Tool User's Guide

---

Welcome to the IBM Tape Device Drivers and Diagnostic Tool User's Guide documentation, where you can find information about how to install, maintain, and use IBM Tape Device Drivers and the IBM Tape Diagnostic Tool (ITDT) on multiple platforms.

Last updated: 2025-04-05.

## Installation and User's Guide

[“Introduction” on page 2](#)

The IBM tape and medium changer device drivers are designed specifically to take advantage of the features that are provided by the IBM tape drives and medium changer devices.

[“Common extended features” on page 4](#)

[“AIX Tape and Medium Changer device driver” on page 12](#)

[“Linux Tape and Medium Changer device driver” on page 40](#)

[“Windows Tape and Medium Changer device driver” on page 67](#)

[“IBM Tape Diagnostic Tool \(ITDT\)” on page 76](#)

### **Troubleshooting and support**

[“Verifying correct attachment of your devices” on page 183](#)

[“Managing the microcode on the IBM tape drive” on page 184](#)

[IBM tape storage support](#)

[IBM Support home page](#)

### **More information**

[IBM Tape Device Drivers Programming Reference](#)

[IBM Tape Device Drivers Installation and User's Guide \(Legacy\)](#)

[IBM TS4500 tape library documentation](#)

[IBM TS4300 tape library documentation](#)

[IBM TS2900 tape autoloader documentation](#)

[IBM TS2270 tape drive documentation](#)

[IBM TS2280 tape drive documentation](#)

[IBM Community \(Community platform\)](#)

[IBM Support content \(Product support\)](#)

[Redbooks home page](#)

## **Introduction**

---

The IBM tape and medium changer device drivers are designed specifically to take advantage of the features that are provided by the IBM tape drives and medium changer devices.

The goal is to give applications access to the functions required for basic tape functions (such as backup and restore) and medium changer operations (such as cartridge mount and unmount), and to the advanced functions needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application. IBM maintains the levels of device drivers and driver documentation for the drive on the Internet. You can access this material at <http://www.ibm.com/support/fixcentral>.

### **Hardware requirements**

The tape drivers and the IBM Tape Diagnostic Tool (ITDT) are developed to support various versions of different platforms. For the latest support, refer to the Interoperation Center website - [IBM System Storage Interoperation Center \(SSIC\)](#).

**Note:** A single Fibre Channel host bus adapter (HBA) for concurrent tape and disk operations is not recommended. Tape and disk devices require incompatible HBA settings for reliable operation and optimal performance characteristics. Under stress conditions (high I/O rates for either tape, disk, or both) where disk and tape subsystems share a common HBA, stability problems are observed. These issues are resolved by separating disk and tape I/O streams onto separate HBAs and by using SAN zoning to minimize contention. IBM is focused on assuring server/storage configuration interoperability. It is strongly recommended that your overall implementation plan includes provisions for separating disk and tape workloads.

For information about this issue, see the following Redbook, section 4.1.3 in <http://www.redbooks.ibm.com/abstracts/sg246502.html?Open>.

### **Software requirements**

**Important:** If you use a third-party application, consult with your application provider about the compatibility with IBM tape device drivers.

Industry-leading compatible software offerings provide storage and tape management software for the LTO tape drives. Supporting software and applications must be obtained separately from IBM®, IBM Business Partners, or independent software vendors (ISV). For a list of compatible software and additional information, refer to the ISV Matrix that is available at [Independent Software Vendor \(ISV\) matrix for 3592 and LTO](#).

## IBM tape products

The IBM tape product family provides an excellent solution for customers with small to large storage and performance requirements.

**Remember:** xx in the following list represents generation of the drive.

- IBM TS22xx Tape Drive
- IBM TS2350/TS2360 Tape Drive
- IBM TS11xx Tape Drive (Enterprise)
- IBM Diamondback Tape Library
- IBM TS4500 Tape Library (also known as IBM tape library 3584)
- IBM TS4300 Tape Library
- IBM TS2900 Tape Autoloader

The image illustrates the attachment of various current products to an open systems server.

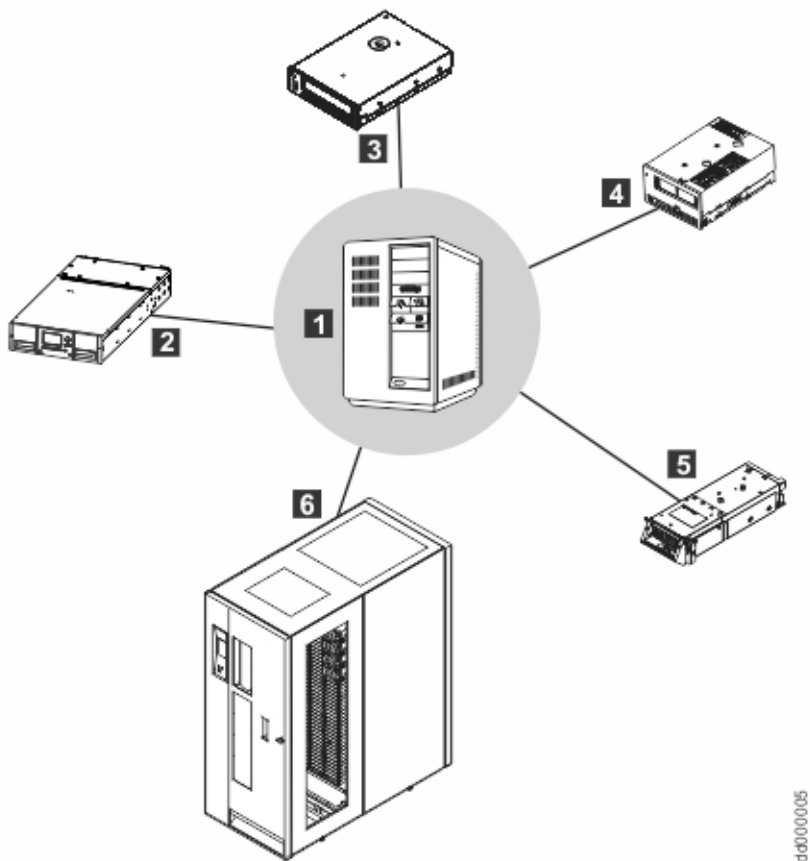


Figure 1. Current attachment array

**1** Open Systems Server

**2** IBM TS4300 Tape Library

**4** IBM TS2350/TS2360 (or 3580) Tape Drive

**5** IBM TS11xx Tape Drive [also known as Enterprise]

## Common extended features

### Purpose

This chapter provides general information about the IBM device drivers, requirements, and advanced functionality.

### Path failover and load balancing

Device driver path failover support configures multiple physical paths to the same device within the device driver and provides two basic functions:

1. Automatic failover to an alternate physical path when a permanent error occurs on one path.
2. Dynamic load balancing for tape devices by using multiple host bus adapters (HBA).

Path failover is supported on certain tape products with the latest IBM device drivers available on the following website - <http://www.ibm.com/support/fixcentral>. Instructions for downloading drivers can be found in “Accessing documentation and software online” on page 77. Some devices require a path failover feature code to be installed before path failover support is enabled in the device driver. Refer to “Supported devices and feature codes” on page 4 for a list of supported devices and what path failover feature code is required for your machine type.

At startup or configuration, the system detects multiple logical devices of the tape drive. Each logical device is a physical path to the same tape drive. A backup and restore application can open and use only one logical device at a time because they represent the same physical device.

Without path failover support, if a permanent path error occurs (because of an HBA or cable failure, for example), the application fails. It is possible to initiate manual failover by restarting the application on the alternate logical device, but the application must be restarted from the beginning. A long backup or restore operation might be in progress when the path error occurred. Sometimes manual failover requires operator intervention to reset the drive because a SCSI Reservation might still exist on the failing HBA path.

When path failover support is enabled on all logical devices, the device driver configures them internally as a single device with multiple paths. The application can still open and use only one logical device at a time. If an application opens the primary device and a permanent path error occurs, the device driver initiates failover error recovery automatically on an alternate path. If successful, the current operation continues on an alternate path without interrupting the application. The data path failover error recovery first restores the previous device state, SCSI Reservation, and tape position. Then, it tries the failing operation again.

### Supported devices and feature codes

Path failover is supported only for the devices that are listed in [Table 1 on page 4](#). Path failover includes Control Path failover (CPF) for tape libraries and Data Path failover (DPF) for tape drives. To use path failover support, some devices require feature codes as listed in [Table 1 on page 4](#).

<i>Table 1. Supported devices and feature codes</i>	
Supported tape library/driver	Feature code (FC), if required
<b>Diamondback Tape Library</b>	Standard, no FC required (CPF and DPF)
<b>TS4500</b>	Standard, no FC required (DPF) FC 1682 (CPF)



Table 1. Supported devices and feature codes (continued)	
Supported tape library/drive	Feature code (FC), if required
TS4300/LTO	FC 1682 (CPF and DPF)

**Notes:**

1. Path failover is only supported on SAS devices that are attached to Windows and Linux® for Intel/AMD processor-based servers. SAS is not supported on System p servers (AIX® and Linux).
2. If your device does not support path failover, you must disable this option in the device driver. See the specific platform section for driver default behavior and enable/disable failover instructions.

## Automatic failover

The automatic failover support provides error recovery on an alternate path when a permanent error occurs on the primary path. This support is transparent to the running application. The two types of path failover are Data Path failover (DPF) and Control Path failover (CPF). They are closely related. However, the difference is that DPF is an automatic failover support for the transfer of data, which provides error recovery for systems that are connected to tape drives. CPF is an automatic failover support for the transfer of commands to move tape cartridges. Examples of different configurations that can be constructed follow.

### Data path failover

The following flowcharts outline the different types of configurations for data path failover (DPF). These configurations are presented in order of best practices as recommended by IBM.

### Dual Host Bus Adapters (HBAs) to a multi-port drive

Consider a multipath connection that consists of two Host Bus Adapters (HBAs) connected through a fabric to a multi-port drive.

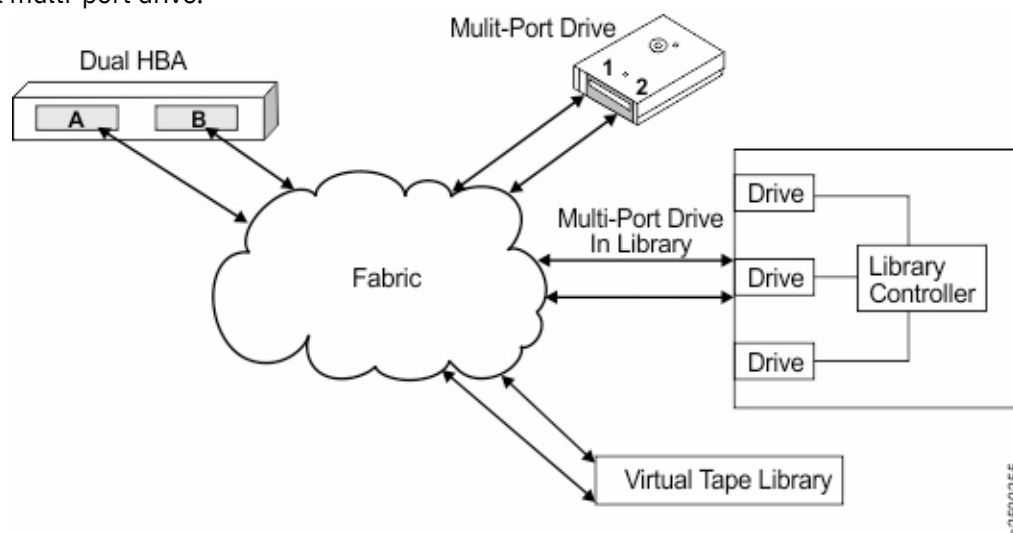


Figure 2. Dual HBA and multi-port drives

As seen in [Figure 2 on page 5](#), four available paths are available between the drive and the host system. These paths are

- HBA A to drive port 1 [A, p1]
- HBA A to drive port 2 [A, p2]
- HBA B to drive port 1 [B, p1]
- HBA B to drive port 2 [B, p2]

One path is the primary path and the other three are alternate paths. If [A, p1] is the primary path and if HBA A fails, two valid paths ([B, p1] and [B, p2]) remain. The DPF tries to switch to one of the available configured paths. Conversely, if the cable to port 1 of the drive fails with [A, p1] as the primary path, two valid paths to the drive [A, p2] and [B, p2] are still available. Without DPF support, if a permanent path error occurs (because of HBA or cable failover, for example), the application fails. With DPF, if the permanent failure occurs with this configuration, two valid physical paths for the data are still available for transmitting data. The running application is not affected.

If the path that failed is restored, the device driver claims the path as available and uses it as a valid alternate path in most conditions. This action is dependent on Operating System and HBA behavior, not the IBM tape device driver behavior.

### Dual Host Bus Adapters (HBAs) to a single-port drive

Consider a multipath connection that consists of two HBAs connected through a fabric to a single-port device.

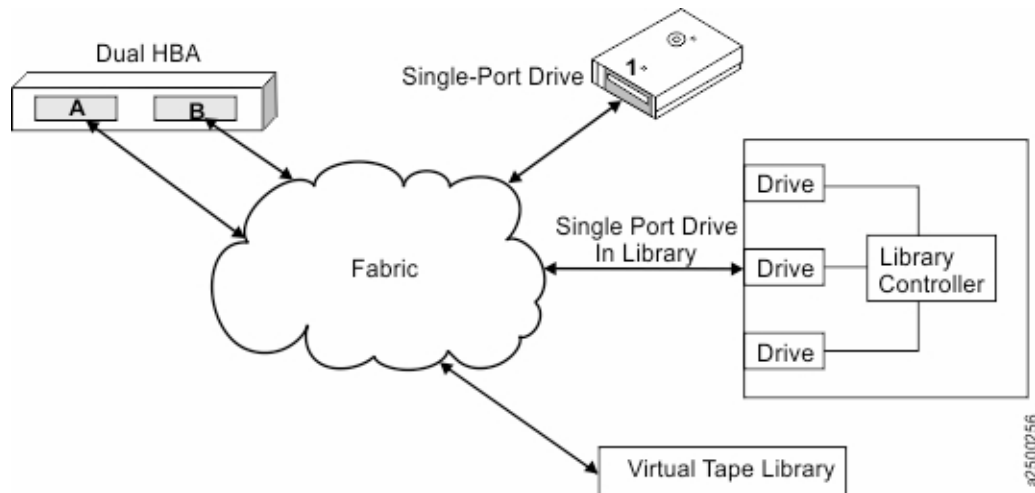


Figure 3. Dual HBA and single-port drive

This configuration supplies two physical paths to the same device. However, if the port or cable from the device fails, the automatic failover does not work. That connection is severed and a permanent path error occurs. If, however, the failure was with one of the HBAs or their cables, the automatic data path failover selects the other HBA. Then, the information continues through the alternate path. An example here is with the connections [A, p1] and [B, p1]. If [A, p1] is the primary path and a failure occurs with the HBA or HBA cable, then DPF automatically moves the connection to [B, p1] without affecting the application.

### Single Host Bus Adapters (HBA) to a multi-port drive

Consider a single path from the HBA through the fabric to a multi-port device.

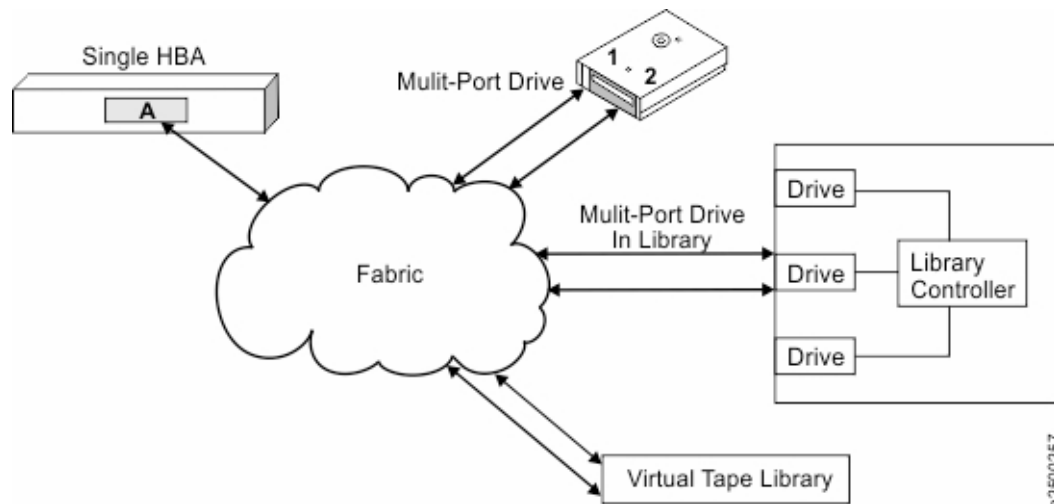


Figure 4. Single HBA and multi-port drive

This configuration also provides a failover path unless the failure is with the HBA or the HBA's cable. At which point, the connection is severed and a permanent path error occurs. Whereas, if the failure occurs on the device side, an alternative path is still available for the information to go through which DPF automatically failovers to.

### Control path failover

The following flowcharts outline the different types of configurations for control path failover (CPF). These configurations are presented in order of best practices as recommended by IBM.

### Dual Host Bus Adapters (HBAs) to multi-port drives

Consider a multipath connection that consists of two Host Bus Adapters (HBAs) connected through a fabric to the library by at least two drives.

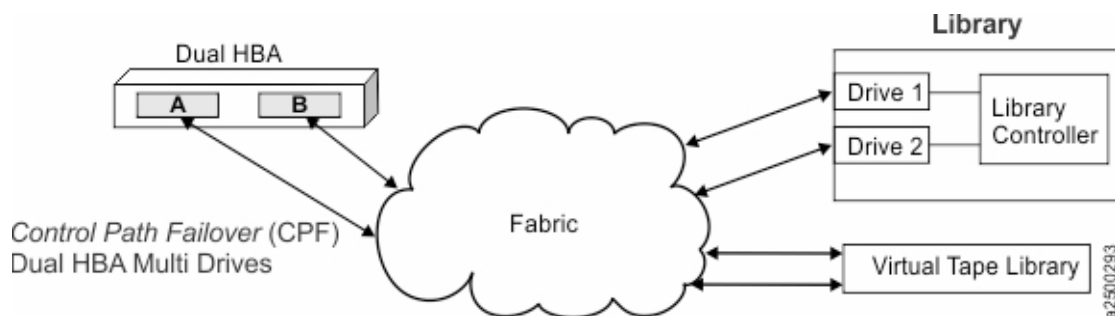


Figure 5. Dual HBA and multi-port drives

As seen in Figure 5 on page 7, four available paths are available between the drive and the host system. These paths are

- HBA A to drive 1 [A, d1]
- HBA A to drive 2 [A, d2]
- HBA B to drive 1 [B, d1]
- HBA B to drive 2 [B, d2]

As with DPF, one path is the primary path and the other three are alternate paths. If [A, d1] is the primary path and if HBA A fails, two remaining valid paths ([B, d1] and [B, d2]) are still available. The CPF attempts to switch to one of the available configured paths. Conversely, if the cable to drive 1 or drive 1 fails with [A, d1] as the primary path, two valid paths to the drive ([A, d2] and [B, d2]) are available. Without CPF support, if a permanent path error occurs (because of HBA or cable failover, for example), the application

fails. With CPF, if a permanent failure with this configuration occurs, two valid physical paths are available for the data to be transmitted. Also, the running application is not affected.

If the failed path is restored, the device driver claims the path as available and uses it as a valid alternate path in most conditions. This action is dependent on Operating System and HBA behavior, not the IBM tape device driver behavior.

**Note:** In the operating systems logs, reservation conflict information might appear, which is because of scsi2 reservations that are not cleared. However, the device driver continues to try any paths that are available to make the reservation conflict transparent to the operating system.

## Single Host Bus Adapter (HBA) to multi-port drives

Consider a single path from the HBA through the fabric to two drives in a library.

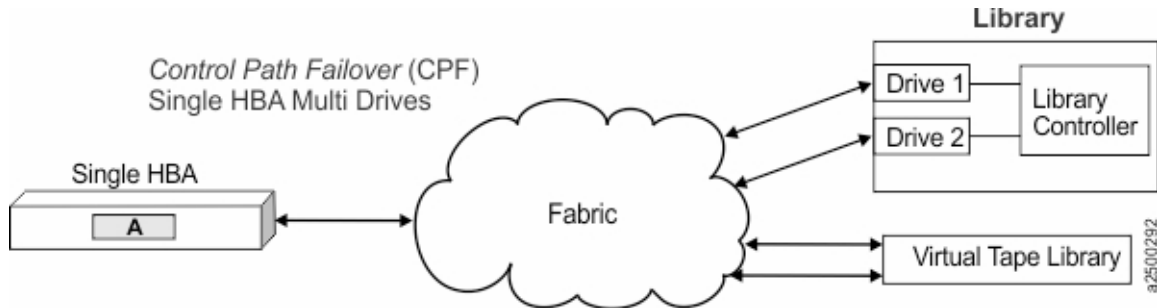


Figure 6. Single HBA and multi-port drives

This configuration also provides a failover path unless the failure is with the HBA or the HBA's cable. At which point, the connection is severed and a permanent path error occurs. Whereas, if the failure occurs with a drive or a drive's cable, an alternative path is still available for the information to go through which CPF automatically failovers to.

## Dynamic load balancing

The dynamic load balancing support optimizes resources for tape devices that have physical connections to multiple host bus adapters (HBA) in the same machine. When an application opens a device that has multiple configured HBA paths, the device driver determines which path has the HBA with the lowest usage. Then, it assigns that path to the application. When another application opens a different device with multiple HBA paths, the device driver again determines the path with the lowest HBA usage. Then, that path is assigned to the second application. The device driver updates the usage on the HBA assigned to the application when the device is closed. Dynamic load balancing uses all host bus adapters whenever possible and balance the load between them to optimize the resources in the machine.

For example, consider a machine with two host bus adapters, HBA1 and HBA2, with multiple tape drives attached. Each tape drive is connected to both HBA1 and HBA2. Initially, there are no tape drives currently in use. When the first application opens a tape drive for use, the device driver assigns the application to use HBA1. When a second application opens a tape drive for use, the device driver assigns the second application to use HBA2. A third application is assigned to HBA1 and a fourth application is assigned to HBA2. Two applications are assigned to HBA1 and two applications are assigned to HBA2.

If the first application finishes and closes the device, there is now one application with HBA1 and two applications with HBA2. When the next application opens a tape drive, it is assigned to HBA1, so again there are two applications with HBA1 and two applications with HBA2. Likewise, if the second application finishes and closes the device, HBA2 has one application that is assigned to it. The next application that opens a tape drive is assigned to HBA2.

The dynamic load balancing support is independent from the automatic failover support. Regardless of the path that is assigned initially for load balancing, if that path fails, the automatic failover support attempts recovery on the next available path.

## Dynamic Runtime Attributes

There are frequently field issues where customers must know which Initiator is holding a reservation in a drive or preventing the media from being unloaded. Also, they must correlate which drive special file name is used for the drive (such as `ımt2`). Sometimes this issue occurs over transport bridges and translators, losing any transport identifiers to help in this effort. LTO5, 3592 E07 (Jag 4) and later physical tape drives support attributes that are set to the drive dynamically by a host. This function is called Dynamic Runtime Attributes (DRA).

This feature is enabled by default. The attributes are set in the drive by the host during open, close, device reset, and data path change only. If there is a problem with sending the attributes to the drive, the error is ignored and not returned to the application.

There is no `ioctl` in the IBM tape drivers to retrieve the dynamic runtime attributes but there is a command on ITDT (See <https://www.ibm.com/docs/en/ts4300-tape-library?topic=commands-12-query-runtime-info>). The attributes can also be retrieved through a pass through `ioctl` to issue **Read Dynamic Runtime Attributes** SCSI command (see applicable *IBM Tape Drive SCSI Reference*). See the host platform section for any special information that pertains to the driver that concerns DRA. If there is a question whether your driver level supports DRA, see the fixlist that comes with your driver to see whether it was added. Updates are also required with the drive firmware.

## Data encryption

### Tape and library requirements

For specific encryption support, refer to SSIC at [IBM System Storage Interoperation Center \(SSIC\)](#)

The following three major elements comprise the tape drive encryption solution.

- **The encryption-enabled tape drive**

The 3592 Model E07 and newer model tape drives, and the LTO Ultrium 4 and newer Ultrium drives are encryption capable. To run hardware encryption, the tape drives must be encryption-enabled. Encryption can be enabled on the encryption-capable tape drives through the **Tape Library Specialist Web** interface. Refer to the appropriate section in the documentation for your library for information about how to enable encryption.

**Note:** FC 1604, Transparent LTO Encryption, is required to use library-managed encryption on LTO Ultrium 4 and newer tape drives. It is not required for application-managed encryption. Refer to the sections on each method of encryption for information.

- **Encryption key management**

Encryption involves the use of several kinds of keys, in successive layers. How these keys are generated, maintained, controlled, and transmitted depends upon the operating environment where the encrypting tape drive is installed. Some data management applications, such as Tivoli® Storage Protect, can run key management. For environments without such applications or where application-agnostic encryption is wanted, IBM provides a key manager (such as the IBM Guardium Key Lifecycle Manager) to complete all necessary key management tasks.

- **Encryption policy**

The method that is used to implement encryption. It includes the rules that govern which volumes are encrypted and the mechanism for key selection. How and where these rules are set up depends on the operating environment.

The LTO Ultrium 6 and later encryption environment is complex and requires knowledge beyond that of product trained Service Support Representatives (SSRs). The Encryption function on tape drives (desktop, stand-alone, and within libraries) is configured and managed by the customer. In some instances, SSRs are required to enable encryption at a hardware level when service access or service password controlled access is required. Customer setup support is by Field Technical Sales Support (FTSS), customer

documentation, and software support for encryption software problems. Customer 'how to' support is also provided with a support line contract.

In the open system environment, there are two methods of encryption management to choose from. These methods differ in where you choose to locate your encryption key manager application. Your operating environment determines which is the best for you, with the result that key management and the encryption policy engine might be in any one of the three environmental layers: application layer, system layer, and library layer.

## Application-managed tape encryption

This method is best where operating environments run an application already capable of generating and managing encryption policies and keys, such as IBM Storage Protect. Policies specifying when encryption is to be used are defined through the application interface. The policies and keys pass through the data path between the application layer and the Encryption is the result of interaction between the application and the encryption-enabled tape drive, and is transparent to the system and library layers.

Refer to [“Planning for application-managed tape encryption” on page 10](#) for details on the hardware and software requirements for application-managed encryption. For details on setting up application-managed tape encryption refer to the IBM Storage Protect documentation or for information, visit [IBM Storage Protect documentation](#).

It is required to use the latest device drivers available. Refer to [“Accessing documentation and software online” on page 77](#) for downloading drivers. In different operating system environments, refer to the applicable chapter for each operating system.

## Library-managed tape encryption

This method is best for encryption-capable tape drives in open attached IBM tape libraries. Scratch encryption policies that specify when to use encryption are set up through the IBM Tape Library Specialist Web interface. Policies are based on cartridge volume serial numbers. Key generation and management are run by an encryption key manager. Policy control and keys pass through the library-to-drive interface, therefore encryption is transparent to the applications.

Refer to [“Planning for library-managed tape encryption” on page 11](#) for details on the hardware and software requirements for library-managed encryption. For details on setting up library-managed encryption on encryption-capable tape drives, refer to the *IBM Tape Library Operator's Guide* for your library.

## Planning for application-managed tape encryption

**Note:** Contact your IBM representative for information about encryption on the IBM encryption-capable tape drive.

To run encryption on the encryption-capable tape drive, the following is required.

- Encryption-capable tape drive
- Encryption configuration features:
  - Library code updates and Transparent LTO Encryption feature code for encryption-capable libraries
  - Tape drive code updates

## Application-managed tape encryption setup tasks

Any task that is not identified as an IBM service task is the responsibility of the customer.

1. Install, cable, and configure the encryption-capable tape drive (refer your *IBM Tape Drive or Library Operator's Guide*)
2. Install appropriate IBM tape device driver level (Atape, for example).
3. Set up encryption policies. Refer to the appropriate IBM Storage Protect documentation.

4. Perform write/read operation to test encryption.
5. Verify encryption of the test volume by Autonomic Management Engine (AME): issue `QUERY VOLUME FORMAT=DETAILED`

Verify that **Drive Encryption Key Manager** is set to IBM Storage Protect.

## Planning for library-managed tape encryption

**Note:** Contact your IBM representative for information about encryption on the IBM encryption-capable tape drive.

To complete encryption on the encryption-capable tape drive, the following items are required.

- Encryption-capable tape drive
- Keystore (Refer to documentation on Guardium Key Lifecycle Manager (GKLM))
- Encryption configuration features
  - Guardium Key Lifecycle Manager (GKLM)
  - Tape system library code updates and Transparent LTO Encryption feature code for encryption-capable libraries
  - Tape drive code updates

## Library-managed tape encryption tasks

Any task that is not identified as an IBM service task is the responsibility of the customer.

1. Install, verify, and configure
  - a. Keystore
  - b. EKM (Refer to documentation on Security Key Lifecycle Manager (SKLM)) for information on both.
2. Install and cable the encryption-capable tape drive (IBM service task for TS1120 Tape Drive).
3. Use IBM tape library specialist to enable the tape drive for library-managed tape encryption (refer to your *IBM Tape Drive or Library Operator's Guide*).
4. Use library diagnostic functions to verify.

## Bulk rekey

For customers with Library-Managed Encryption with 3592 Enterprise tape drives and IBM tape and changer drivers that are running on open systems operating system (AIX, Linux, Windows), sample code for completing bulk rekey operations is available. The sample code packages are provided "as-is" with limited testing, and are provided to give customers guidance on bulk rekey operations.

For UNIX operating systems, a sample script (**rekey\_unix.sh**) is provided and must be used with the *tapeutil* version that is bundled in the same package. For Windows operating systems, a sample c program (**rekey\_win.c**) is provided. Both of these sample programs must be used with both the IBM tape and changer drivers. In addition, data cartridges must be in storage cells, not in I/O station cells or tape drives.

For information and to download the sample code packages, see <http://www.ibm.com/support/fixcentral/>.

## Encryption feature codes

To use system-managed and library-managed encryption, the Transparent LTO Encryption feature codes that are listed in [Table 2 on page 12](#) are required for the associated IBM tape libraries with encryption-capable tape drives. If the drives in use are TS1120 tape drives, this feature code is not required for system-managed or library-managed encryption. If you are using application-managed encryption, no feature code is required on any encryption-capable tape drives.

Table 2. Feature codes (encryption)	
Tape library	Feature code
Diamondback	Standard, no FC required
TS4500	FC 1604
TS4300	FC 5900

## Recommended access order (RAO) open function

RAO enables tape control applications to accelerate the retrieval of a certain number of files from a single tape thereby reducing the seek time between those files.

A feature of the LTO-9 full-height drives is the ability to accept a list of User Data Segments and reorder those User Data Segments into a recommended access order that minimizes the locate portion of the time to read those User Data Segments. This sorted list is called a Recommended Access Order (RAO) list. A User Data Segment (UDS) is defined as a grouping of contiguous logical objects (i.e., logical blocks and filemarks) and is described by partition number, beginning logical object identifier, and ending logical object identifier.

The RAO implementation in LTO produces the best results for performance enhancement when there is little variability in block size or data compression ratio. When the variability in compression ratio or block sizes increase, the accuracy of the locate estimates may be reduced and any potential performance enhancements may be diminished.

## AIX Tape and Medium Changer device driver

This chapter describes the IBM AIX Enhanced Tape and Medium Changer Device Driver (Atape) for IBM tape devices.

### Purpose

The IBM AIX Enhanced Tape and Medium Changer device driver is designed to take advantage of the features that are provided by the IBM tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape operations (such as backup and restore) and medium changer operations (such as mount and unmount the cartridges), and to the advanced functions needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application.

### Data flow

The software that is described in this chapter covers the AIX Enhanced Device Driver (Atape device driver) and the interface between the application and the tape device. For data flow, refer to [Figure 7 on page 12](#).

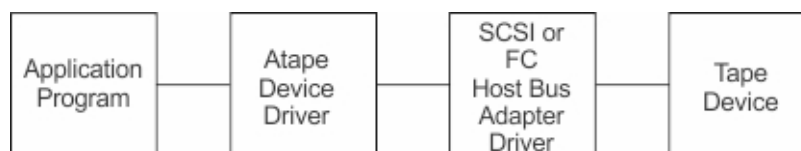


Figure 7. Data flow for AIX Device Driver (Atape)



# Product requirements

## Hardware requirements

Refer to [“Hardware requirements” on page 2](#) the latest hardware that is supported by the Atape device driver.

## Software requirements

The AIX Enhanced device driver (Atape device driver) supports AIX 5L Version 5.3 and later releases on IBM POWER-based AIX servers.

For current software requirements, refer to the [“Software requirements” on page 2](#).

# Installation and configuration instructions

The recommended procedure for installing a new version of the device driver is to uninstall the previous version.

Instructions for uninstalling the device driver are outlined in [“Uninstalling” on page 16](#).

1. At the end of the installation procedure, the **installp** facility automatically runs the AIX **bosboot** command to update the boot record with the newly installed Atape files. When the **bosboot** command completes, the following messages are displayed:

```
0503-292 This update does not fully take effect until after a system reboot.  
installp: bosboot process completed.
```

This message refers to the updates to the boot record only. If the installation summary shows that the Atape driver was installed successfully, it is not necessary to reboot the machine currently.

If the installation summary shows that the installation failed, you must reboot the machine and attempt to install the Atape driver a second time.

2. During the Atape install, the following entries are entered into the two system files.

- **/usr/lpp/bosinst/cdfs.optional.list** to help the system image backup to DVD/CD media

The entry list in **/usr/lpp/bosinst/cdfs.optional.list**:

```
/usr/lib/drivers/Atape      /usr/lib/drivers/Atape      Atape.driver  
/usr/lib/methods/cfgAtape /usr/lib/methods/cfgAtape  Atape.driver  
/usr/lib/methods/ucfgAtape /usr/lib/methods/ucfgAtape Atape.driver  
/usr/lib/methods/defAtape /usr/lib/methods/defAtape  Atape.driver  
/usr/lib/methods/udefAtape /usr/lib/methods/udefAtape Atape.driver  
/usr/lib/methods/chgAtape /usr/lib/methods/chgAtape  Atape.driver
```

- **/usr/lpp/bosinst/tape/tapefiles1** to create a bootable tape

The entry list in **/usr/lpp/bosinst/tape/tapefiles1**:

```
/usr/lib/drivers/Atape  
/usr/lib/methods/ucfgAtape  
/usr/lib/methods/cfgAtape  
/usr/lib/methods/udefAtape  
/usr/lib/methods/defAtape  
/usr/lib/methods/chgAtape
```

The entries are removed from the files when Atape is uninstalled.



**Attention:** The entries might be lost when a user upgrades the AIX file set of **bos.sysmgt.sysbr** for System Backup and BOS Install Utilities after Atape installation. It is recommended that the user check whether the entries still exist and add the entries into the files if needed.

## Installation procedure

For information on obtaining the latest version of device drivers and the latest documentation, refer to [“Accessing documentation and software online” on page 77](#).

### Preinstallation considerations

Before the installation starts, verify the following items:

1. The tape device is properly functioning, properly attached to the server, and is powered up.
2. You logged on to the server on an account that has *root* authority.
3. You have a command shell window open on the server to run the installation procedure.
4. Make sure that the current path is defined in the command shell PATH variable. This definition can be accomplished in the Korn shell by using the following command:

```
EXPORT PATH=.:$PATH
```

5. If the tape device was configured previously by another device driver (not Atape), remove any existing device definitions for it. The following command is an example: **rmdev -l ost1 -d**

### Installation procedure

Enter the following command to list the currently installed *Atape.driver* version:

```
lslpp -l Atape.driver
```

Enter the following command to install the Atape driver in the current directory. For example

```
installp -acXd Atape.x.x.x.x Atape.driver
```

This command installs and commits the Atape driver on the system.

### Configuring Tape and Medium Changer devices

After the driver software is installed and a tape device is connected to the adapter, the device can be configured and made available for use. Access to the device is not provided until the device is configured.

**Note:** If the tape device was configured previously by another SCSI device driver, such as OST (Other SCSI Tape), issue the following command to remove the device definition before the following steps are completed.

```
rmdev -l [device]
```

Configure a tape device by using one of the following procedures.

- Enter the following command with no parameters.

```
cfgmgr
```

The command configures all devices automatically (including any new tape or medium changer devices).

- Power Off your subsystem and reboot the system to configure it automatically and make available any new tape or medium changer devices on the system.

### Configuring limitations

The subsequent limitations are applied for the Atape driver that runs on an AIX host.

Maximum supported number of tape devices	1024
--	------

Maximum supported number of HBA ports	32
Maximum supported number of paths for a device (DPF/CPF)	16/16
Maximum LUN size per target for FC HBA*	4095

**Note:** \*On AIX systems, the maximum LUN number is 4095. Since Atape supports up to 1024 devices, Atape configures a total of 1024 devices by using the range from LUN 0 - 4095. For instance, a device with LUN 4095 at a SCSI target address can be configured by Atape if the total number of devices on the system is less than 1024.

Every opened tape device uses a certain amount of resources. The user must also consider other resources such as physical memory and virtual space on the system before you attempt to reach the limits.

## Deconfiguring tape devices

**Note:** In the following examples, replace the letter *n* with the appropriate number for the chosen device.

Deconfigure the tape device by using one of the following procedures:

1. The first method leaves the tape device that is defined in the configuration database. It is similar to bringing the device offline (not in use).

Enter the following command to bring the **/dev/rmt $n$**  tape device offline, but leave it defined in the device database.

```
rmdev -l rmt $n$ 
```

2. The second method brings the tape device offline and removes its definition from the device database.

Enter the following command.

```
rmdev -l rmt $n$  -d
```

The device driver is not unloaded from the kernel until the last device is deconfigured.

## Deconfiguring Medium Changer devices

**Note:** In the following examples, replace the letter *n* with the appropriate number for the chosen device.

Deconfigure the medium changer device by using one of the following procedures:

1. The first method leaves the device that is defined in the configuration database. It is similar to bringing the device offline.

Enter the following command to bring the **/dev/smc $n$**  medium changer device offline, but leave it defined in the device database.

```
rmdev -l smc $n$ 
```

2. The second method brings the medium changer device offline and removes its definition from the device database.

Enter the following command.

```
rmdev -l smc $n$  -d
```

The device driver is not unloaded from the kernel until the last device is deconfigured.

## Uninstalling



**Attention:** All devices that use the Atape driver must be closed and cannot be in use when Atape is uninstalled or the uninstall fails.

You can uninstall the Atape device driver by using the **smit** command menu to uninstall software and selecting **Atape.driver** or by using the following **installp** command

```
installp -u Atape.driver
```

## Tape drive, media, and device driver parameters

This chapter describes the parameters that control the operating modes of the AIX Enhanced Tape and Medium Changer Device Driver.

### Configuration parameters

The operating parameters for the tape drive and device driver are set and changed by configuration parameters. The installation defaults are provided for all parameters initially. The AIX **smit** command is used to set these parameters when a device is configured or to change these parameters. The AIX **chdev** command is used to change the configuration parameters.

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. These parameters can be queried by an application. Some parameters can be temporarily changed during the *open* subroutine by an application. But, they are always restored to the configuration values when a device is closed. The configuration parameters are

- Alternate Pathing
- Autoloading
- Emulate autoloader (359x devices only)
- Block size
- Buffered mode (359x devices only)
- Compression
- Fail degraded media (359x devices only)
- Logical write protect (359x devices only)
- Logging
- Maximum size of the log file
- New logical name
- Read error recovery time (359x devices only)
- Record space mode
- Reservation key
- Reservation support
- Reservation type
- Retain reservation
- Rewind immediate
- System encryption
- System encryption for Write Commands
- Trailer labels
- SCSI status busy retry
- iostat support for tape

## Alternate pathing

This parameter enables or disables the path failover support when a device is configured. See [“Data Path failover and load balancing support for tape drives” on page 27](#) for a description of the path failover and failover support.

The installation default is **no** (path failover is not enabled).

## Autoloading

This parameter enables the autoloading feature of the device driver. It is used with the autoloading capability of the autoloader, ACF, ACL, or CSL installed on the tape device.

**Note:** The autoloading feature is not supported on the IBM 3584 UltraScalable tape library and the IBM 3583 Ultrium Scalable tape library with more than one IBM 3580 Ultrium tape drive installed.

**Note:** The autoloading feature is supported only on the following device types and configurations:

- IBM 3490E Models C11, C22, E01, E11, F01, and F11
- IBM Enterprise Tape System 3590, Models B11, E11, and H11
- IBM Magstar MP 3570 Models B01, C01, B11, and C11
- IBM Magstar MP 3570 Models B02, B12, C02, and C12 (configured in split mode only)
- IBM 7332 (all models)

Do not enable autoloading if one of the following conditions is true.

- The device is used by an application that provides library medium changer support for the IBM 3581 or IBM 3583.
- The device is installed in a 3494 Enterprise Tape Library.
- The device is used by an application with stack loader support.
- The application is MKSYSB.
- The tapes that are read were not written with the autoloading feature.

Tapes that are created with AUTOLOAD=YES are not readable in configurations without Atape autoloading enabled, or on other UNIX operating systems, or on device types/models that are different from the backup device type/model.

If the parameter is set to On, then the tape stacker acts as one large virtual tape. During a read, write, or forward space file operation, no end of tape is detected by the application. When the end of tape is reached, the device driver automatically rewinds and unloads the tape, then loads the next tape. Then, it continues reading or writing the next tape. The following conditions are required to use this feature:

- The autoloading parameter must be set to On.
- The cartridge stacker must be loaded with one or more tapes.
- The ACF, ACL, or CSL must be set to Automatic, System, or Random mode.

This feature allows multivolume backups (with commands such as **tar**) without prompting for a volume change.

The installation default is Off (no autoloading).

## Emulate autoloader

This parameter controls how the device driver operates when the ACF on the IBM Enterprise Tape System 3590, the IBM Magstar MP tape device, or the IBM 3490E Model Fxx is set to Random mode. If this parameter is set to On and the ACF is in Random mode, the device driver emulates an autoloading tape drive. When an unload command is sent to the device driver to unload a tape, the tape is unloaded, returned to the magazine, and the next tape in the magazine is loaded automatically into the tape drive. If this parameter is set to Off, the normal unload operation occurs, and the tape remains in the drive.

The emulate autoloader parameter can be used for legacy applications that are written for the IBM 3490E Automated Cartridge Loader (ACL) when the IBM Enterprise Tape System 3590, the IBM Magstar MP 3570, or the IBM 3490 Model F autoloader is set to Random mode. This parameter eliminates the need to reconfigure the autoloader of the device Random or Automatic operation.

The installation default is Off (do not emulate autoloader).

**Note:** On IBM Magstar MP 3570 Models B02, C02, and C12, this feature is supported only when the two drives are configured in Split mode, or in Base mode with one drive that is configured and available to AIX. This feature does not work in Base mode if both drives are in the available state to AIX.

## Block size

This parameter specifies the block size that is used for read and write operations. A value of zero is the variable block size. Any other value is a fixed block size.

The installation default is zero (use variable length) except for the IBM 7332 4-mm Tape Cartridge Autoloader, for which the default is a fixed block size of 1024 bytes.

## Buffered mode

When a **write** command is processed, the data is either stored directly on the physical tape or buffered in the tape device. Buffering can increase the device performance.

The installation default is On (use Buffered mode).

## Compression

Hardware compression is implemented in the device hardware. This parameter turns the compression feature On and Off. If compression is enabled, then the effective performance can increase based on the compressibility of the data.

The installation default is On (use compression).

## Fail degraded media

This parameter controls whether the device driver fails a tape operation when degraded media is detected by the IBM Enterprise Tape System 3590. If a tape is loaded and the IBM 3590 cannot read the positioning information from the tape, the device driver is notified when the first command is sent to the tape drive. If this parameter is set to On, the device fails the command and returns a media error to the application. If this parameter is set to Off, the device driver does not fail the command.

Degraded media is a correctable condition that prevents the IBM Enterprise Tape System 3590 from running high speed **Locate** operations. A **Locate** command can take over 20 minutes, depending on the wanted position and the amount of data on the tape. This parameter is intended for use by real-time applications that cannot tolerate long **Locate** commands.

The installation default is Off (do not fail the tape operation if degraded media is detected).

## Logging

This parameter turns the volume information logging on and off. If logging is set to On, the statistical information about the device and media is saved in a log file when a tape is unloaded. If logging is set to Off, the information is not saved. This parameter has no effect on error logging because error logging is always enabled. For information, refer to [“Device and volume information logging” on page 31](#).

The installation default is Off (no logging).

## Maximum size of the log file

This parameter specifies the number of entries that are made before the log file starts to wrap. Each entry is approximately 2 KB (2048 bytes). After the log file starts to wrap, the number of entries stays constant.

Each time a new entry is made, the oldest entry is overlaid. For information, refer to [“Device and volume information logging” on page 31](#).

The installation default is 500.

## New logical name

Setting this parameter changes the logical name of the device to a new name as specified. After the logical name is changed, the new logical name parameter is cleared. For information, refer to [“Persistent Naming Support” on page 24](#).

There is no installation default value for this parameter.

## Read error recovery time

This parameter controls the read error recovery time for the IBM Enterprise Tape System 3590. If this parameter is set to On, the recovery time for read errors is limited to a maximum of 5 seconds. If this parameter is set to Off, full recovery is used by the device and can take up to 10 minutes. This parameter is intended for use by real-time applications that cannot tolerate long delays when data is read from the tape.

The installation default is Off (do not limit the read error recovery time).

## Record space mode

This parameter specifies how the device driver operates when a forward or backward space record operation encounters a filemark. The two modes of operation are SCSI and AIX.

The SCSI mode is the default mode of operation. When a forward or backward space record operation is issued to the driver and a filemark is encountered, the device driver returns **-1** and the *errno* variable is set to input/output error (EIO). The tape is left positioned after the filemark (the end-of-tape side of the filemark on the forward space and the beginning-of-tape side of the filemark on the backward space).

The AIX mode returns the same EIO *errno* value as the SCSI mode when a filemark is encountered except that the tape is left positioned before the filemark (the beginning-of-tape side of the filemark on the forward space and the end-of-tape side of the filemark on the backward space).

The installation default is SCSI mode.

## Reservation key

This parameter specifies the SCSI Persistent Reservation key that is used by the device driver when either the **Reservation Type** parameter is SCSI Persistent Reserve and the **Alternate Pathing** parameter is set to no or when the **Alternate Pathing** parameter is set to **Yes**.

The default for this attribute is blank (NULL).

If the **Reservation Key** parameter is specified as blank (NULL), then the device driver uses an internal unique key for all devices on the host they are configured on. Another AIX host that shares devices also have an internal unique key for all devices if the **Reservation Key** parameter was blank (NULL).

If the default is not used, then the Reservation Key value can be specified as either a 1-8 character ASCII alphanumeric key or a 1-16 hexadecimal key that has the format 0xkey. If fewer than 8 characters are used for an ASCII key (such as host1), the remaining characters are set to 0x00 (NULL). If less than a 16 hexadecimal key is used, the remaining bytes are set to 0x00.

**Note:** When a Reservation Key is specified on each host that shares a device, the key must be unique to each host.

## Reservation support

The parameter of **reserve\_support** indicates that the Atape driver manages the reservation for the tape device when it is enabled. Atape reserves the tape device in open and releases it in close, and maintains the reservation in error recovery procedure (ERP).

**Note:** For the medium changer, this parameter is not applied when the **Alternate Pathing** (path failover) parameter is set to **Yes**. The device driver forces the setup to be disabled and the medium changer is not reserved in open when the **Alternate Pathing** parameter is set to **Yes**.

The installation default is **Yes**.

## Reservation type

This parameter specifies the SCSI Reservation type that is used by the device driver, either a **SCSI Reserve 6** command or a **SCSI Persistent Reserve** command.

**Note:** This parameter is not used if the **Alternate Pathing** (path failover) parameter is set to **Yes**. The device driver uses SCSI Persistent Reserve when the **Alternate Pathing** parameter is set to **Yes**.

The installation default is SCSI Reserve 6.

## Retain reservation

When this parameter is set to 1, the device driver does not release the device reservation when the device is closed for the current open. Any subsequent opens and closes until the **STIOCSETP** IOCTL is issued with **retain\_reservation** parameter set to 0. The device driver still reserves the device on open to make sure that the previous reservation is still valid.

The installation default is Off (the reservation is released in close).

## Rewind immediate

This parameter turns the immediate bit On and Off in rewind commands. If it is set to On, the rewind tape operation runs faster. However, the next command takes a long time to finish unless the rewind operation is physically complete. Setting this parameter reduces the amount of time that it takes to close a device for a **Rewind on Close** special file.

The installation default is Off (no rewind immediate) except for the IBM 7332 4-mm Tape Cartridge Autoloader, for which the default is On (rewind immediate).

## System encryption

This parameter specifies whether System-Managed Encryption must be used. For information, refer to [“System-managed encryption” on page 29](#).

The installation default is No.

## System encryption for Write commands

This parameter controls if System-Managed Encryption is used for Write commands. For information, refer to [“System-managed encryption” on page 29](#).

The installation default is Custom.

## Trailer labels

If this parameter is set to On, then writing a record past the early warning mark on the tape is allowed. The first write operation to detect EOM fails, and the *errno* variable is set to ENOSPC. No data is written during the operation. All subsequent write operations are allowed to continue until the physical end of the volume is reached and EIO is returned.



This parameter can also be selected by using one of three device special files that allow trailer-label processing. The special files are `rmtx.40`, `rmtx.41`, and `rmtx.60`, where `x` is the name of the device (for example, `rmt0.40`).

The installation default is Off (no trailer labels).

## SCSI status busy retry

Atape retries the SCSI command fail due to the SCSI status **Busy** when the parameter of **busy\_retry** is set to On. Otherwise, Atape fails the SCSI command if it is set to Off.

The installation default is Off.

## iostat support for tape

The **iostat** command is used to monitor system input/output (I/O) devices. To work this system command for tape, Atape reports input and output statistics on each tape drive.

The installation default is On.

## Media parameters

The ability to set or change media parameters is a tape diagnostic and utility function, refer to [“IBM Tape Diagnostic Tool \(ITDT\)”](#) on page 76.

The media parameters can be queried and set by ITDT or the tape diagnostic and utility function by using the Query/Set Parameters option in the window.

These parameters cannot be set or changed by the configuration procedures. The media parameters are

- Capacity scaling
- Logical write protect
- Volume ID for logging
- Archive mode unthread (AMU)

## Capacity scaling

This parameter sets the capacity or logical length of the current tape on IBM Enterprise Tape System 3590, IBM Enterprise Tape System 3592, or Magstar MP tape subsystems. By reducing the capacity of the tape, the tape drive can access data faster at the expense of data capacity.

Capacity scaling can be set at 100% for the entire tape (which is the default) or set at 75%, 50%, or 25% of the tape or any device-specific hexadecimal value. For example, on IBM 3592, to change capacity scaling from a 300 GB format tape (100%) to a 60 GB format tape, select the capacity scaling option. Then, select the option to enter a hexadecimal value and enter 35. Capacity scaling remains with the tape across mounts until it is changed.

### Note:

1. The tape position must be at the start of the tape to change this parameter from its current value.
2. Changing this parameter deletes any existing data on the tape.
3. Attempting to set capacity scaling that is not supported by a device or the current loaded media always returns 100% and cannot be changed. For example, 60 GB media for the IBM 3592 cannot be capacity scaled and is always 100%.

## Logical write protect

This parameter sets or resets the logical write protect of the current tape on IBM Enterprise Tape System 3590, IBM Enterprise Tape System 3592, or Magstar MP tape subsystems. The three types of logical write-protect are associated-protect, persistent-protect, and write-once read-many (WORM)-protect.

Associated protect remains only while the current tape is mounted or associated with the tape drive. It is reset when the tape is unloaded or the tape drive is reset.

Persistent protect remains or persists with the tape across mounts until it is reset.

WORM protect also remains with the tape across mounts, but (unlike persistent protect) it cannot be reset on the tape. After a tape is WORM protected, it can never be written on again.

**Note:**

1. The tape position must be at the start of the tape to change this parameter from its current value.
2. Attempting to set logical write protect that is not supported by a device or the current loaded media always returns "No" and cannot be changed.

## Volume ID for logging

This parameter is the volume ID of the current loaded tape. It is used in the log file entry (if volume logging is active) to identify the entry with a particular volume. The device driver sets the volume ID to UNKNOWN initially and when the tape is unloaded.

## Archive mode unthread (AMU)

This parameter turns Archive mode unthread (AMU) On and Off. When it is set to On, Atape manages and works the AMU feather that rewinds the tape cartridge to the end of tape at a low tension for long-term storage. The AMU feature on the tape drive is enabled for 3592 tape drives and disabled for LTO tape drives by default. To enable or disable this feature on the tape drive, the parameter must be turned on in the Atape driver.

## Special files

When the driver is installed and a tape device is configured and available for use, access is provided through the special files. These special files, which consist of the standard AIX special files for tape devices (with other files unique to the Atape driver), are in the **/dev** directory.

## Special files for tape devices

Each tape device has a set of special files that provides access to the same physical drive but to different types of functions. As shown in [Table 3 on page 23](#), in addition to the tape special files, a special file is provided for tape devices that allow access to the medium changer as a separate device.

**Note:** The asterisk (\*) represents a number that is assigned to a particular device (such as **rmt0**).

For tape drives with attached SCSI medium changer devices, the **rmt\*.smc** special file provides a separate path for commands that are issued to the medium changer. When this special file is opened, the application can view the medium changer as a separate SCSI device.

Both this special file and the **rmt\*** special file can be opened at the same time. The file descriptor that results from opening the **rmt\*.smc** special file does not support the following operations.

- Read
- Write
- Open in Diagnostic mode
- Commands that are designed for a tape drive

If a tape drive has a SCSI medium changer device that is attached, then all operations (including the medium changer operations) are supported through the interface to the **rmt\*** special file. For detailed information, refer to [Table 3 on page 23](#).

Table 3. Special files for tape devices

Special file name	Rewind on Close (Note 1)	Retension on Open (Note 2)	Bytes per Inch (Note 3)	Trailer Label	Unload on Close
/dev/rmt*	Yes	No	N/A	No	No
/dev/rmt*.1	No	No	N/A	No	No
/dev/rmt*.2	Yes	Yes	N/A	No	No
/dev/rmt*.3	No	Yes	N/A	No	No
/dev/rmt*.4	Yes	No	N/A	No	No
/dev/rmt*.5	No	No	N/A	No	No
/dev/rmt*.6	Yes	Yes	N/A	No	No
/dev/rmt*.7	No	Yes	N/A	No	No
/dev/rmt*.10 (Note 4)	No	No	N/A	No	No
/dev/rmt*.20	Yes	No	N/A	No	Yes
/dev/rmt*.40	Yes	No	N/A	Yes	No
/dev/rmt*.41	No	No	N/A	Yes	No
/dev/rmt*.60	Yes	No	N/A	Yes	Yes
/dev/rmt*.null (Note 5)	Yes	No	N/A	No	No
/dev/rmt*.smc (Note 6)	N/A	N/A	N/A	N/A	N/A

**Note:**

1. The Rewind on Close special files write filemarks under certain conditions before rewinding.
2. The Retensions on Open special files rewind the tape on open only. Retensioning is not done because these tape products complete the retension operation automatically when needed.
3. The Bytes per Inch options are ignored for the tape devices that are supported by this driver. The density selection is automatic.
4. The **rmt\*.10** file bypasses normal close processing, and the tape is left at the current position.
5. The **rmt\*.null** file is a pseudo device similar to the **/dev/null** AIX special file. The input/output control (*ioctl*) calls can be issued to this file without a real device that is attached to it, and the device driver returns a successful completion. Read and write system calls return the requested number of bytes. This file can be used for application development or debugging problems.
6. The **rmt\*.smc** file can be opened independently of the other tape special files.

## Special files for Medium Changer devices

After the driver is installed and a medium changer device is configured and made available for use, access to the robotic device is provided through the **smc\*** special file in the **/dev** directory.

Table 4 on page 24 shows the attributes of the special file. The asterisk (\*) represents a number that is assigned to a particular device (such as **smc0**). The term **smc** is used for a SCSI medium changer device. The **smc\*** special file provides a path for issuing commands to control the medium changer robotic device. For information, refer to Table 4 on page 24.

Table 4. Special files for Medium Changer devices

Special file name	Description
/dev/smc*	Access to the medium changer robotic device
/dev/smc*.null	Pseudo medium changer device
<b>Note:</b> The <b>smc*.null</b> file is a pseudo device similar to the <b>/dev/null</b> AIX special file. The commands can be issued to this file without a real device that is attached to it, and the device driver returns a successful completion. This file can be used for application development or debugging problems.	

The file descriptor that results from opening the **smc** special file does not support the following operations:

- Read
- Write
- Commands that are designed for a tape device

## Persistent Naming Support

Persistent naming support is used to ensure that attached devices are always configured with the same logical name based on the SCSI ID, LUN ID, and host bus adapter (HBA), even when the system is rebooted.

When the AIX operating system is booted, the HBA runs a device discovery and assigns a default logical name to each device found in a sequential order. If there are three tape drives attached to a parallel SCSI adapter, each with a LUN ID of 0 and a target address of 0, 1, and 2, the HBA initially configures them as Available with the following logical names.

```
rmt0    target 0, lun 0    Available
rmt1    target 1, lun 0    Available
rmt2    target 2, lun 0    Available
```

Run the following commands before the machine is rebooted.

```
- rmdev -dl rmt1
- rmdev -dl rmt2
```

On the next reboot, if the existing rmt1 target 1 device is powered off or not connected, the HBA initially configures two devices as Available with the following logical names:

```
rmt0    target 0, lun 0    Available
rmt1    target 2, lun 0    Available
```

If the previous rmt1 target 1 device is powered on after reboot and the **cfgmgr** command is run, the HBA configures the device as rmt2 instead of rmt1.

```
rmt2    target 1, lun 0    Available
```

This is one example, but there are other cases where the logical names of devices could change when the system is rebooted. For applications that need a consistent naming convention for all attached devices, it is accomplished with persistent naming support by defining a unique logical name (other than the AIX default names) that are associated with the specific SCSI ID, LUN ID, and HBA that the device is connected to.

## Changing the logical name after initial boot

The logical name of a device can be changed after an initial boot and configured. This procedure can be done by using the **SMIT** menu or the **chdev** command from a script or command line.

For example, a default rmt0 logical name for a tape drive can be changed to rmt-0, tape0, or any descriptive name wanted. In this example, if the three tape drives are changed to rmt-0, rmt-1, and rmt-2, and the system is then rebooted with rmt-1 powered off, the HBA detects that unique names are predefined for the attached devices, and the HBA uses those names. In this case, the devices configure as follows:

```
rmt-0    target 0, lun 0    Available
rmt-1    target 1, lun 0    Defined
rmt-2    target 2, lun 0    Available
```

Since rmt-1 is not detected by the HBA but is predefined at the SCSI ID and LUN ID, it remains in the defined state and is not configured for use. But, the next rmt-2 tape drive configures as the same name at the same location after reboot.

## Changing the logical name with SMIT

To change the logical name by using SMIT, complete the following steps:

1. Run **SMIT** from a command line and select **Devices**.
2. Select **Tape Drive**.
3. Select **Change/Show Characteristics of a Tape Drive**.
4. Select the logical device to be changed from the list displayed.
5. In the **New Logical Name** field, enter a non-AIX default logical name.
6. Press **Enter** to process the change.

## Changing the logical name with the chdev command

The logical name of a device can be changed by using the **chdev** command. For example, to change the logical name of the device from rmt0 to rmt-0, run

```
chdev -l rmt0 -a new_name=rmt-0
```

The output of this command displays

```
rmt0 changed
```

### Note:

- When path failover is enabled, if you change the logical name for either a primary or alternate device, only the individual device name changes.
- Follow the naming convention whenever you run **mksysb**, **bosboot**:
  - The prefix name of "rmt" cannot be changed.
  - A sequence number must be a positive integer. The smallest sequence number is 0.
  - The prefix name cannot contain non-numerical characters. For example, **rmt1\_rescu** is not an acceptable prefix name.
  - When a device instance logical name is generated, the SMIT tool automatically assigns the next available sequence number (relative to a specific prefix name). The next available sequence number is defined as the smallest sequence number for a particular prefix name that is not yet allocated.

## Control Path failover support for tape libraries

**Note:** The library control path failover feature code must be installed before enabling the path failover support in the Atape device driver. Refer to [“Automatic failover” on page 5](#) for what feature codes might be required for your machine type.

The Atape device driver path failover support configures multiple physical control paths to the same logical library within the device driver. It also provides automatic failover to an alternate control path when a permanent error occurs on one path. This support is transparent to the running application.

## Configuring and unconfiguring path failover support

Path failover support is not enabled automatically when the device driver is installed. It must be configured initially on each logical device after installation. When path failover support is enabled for a logical device, it remains set until the device is deleted or the support is unconfigured. The alternate path failover setting is retained even if the system is rebooted.

To enable or disable the support on a single logical device, use the **SMIT** menu to **Change/Show Characteristics of a Tape Drive**, select the logical device to change such as smc0, smc1, then select **Yes** or **No** for **Enable Path Failover Support**. The support can also be enabled or disabled by using the **chdev** command, for example,

```
chdev -l smc0 -aalt_pathing=yes
chdev -l smc1 -aalt_pathing=yes
chdev -l smc0 -aalt_pathing=no
chdev -l smc1 -aalt_pathing=no
```

## Primary and alternative paths

When the device driver configures a logical device with path failover support enabled, the first device that is configured always becomes the primary path. On SCSI attached devices, **-P** is appended to the location field. On Fibre attached devices, **-PRI** is appended to the location field of the device.

When a second logical device is configured with path failover support enabled for the same physical device, it configures as an alternative path. On SCSI attached devices, **-A** is appended to the location field. On Fibre attached devices, **-ALT** is appended to the location field of the device.

A third logical device is also configured as an alternative path with either **-A** or **-ALT** appended, and so on. The device driver supports up to 16 physical paths for a single device.

The labeling of a logical device as either a primary or alternative path is for information only, to

1. Be able to identify the actual number of physical devices that are configured on the system and a specific logical device that is associated with them. Only one logical device is labeled as the primary path for each physical device. However, many (multiple) logical devices can be labeled as an alternative path for the same devices.
2. Provide information about which logical devices configured on the system have path failover support enabled.

## Querying primary and alternative path configurations

You can see the primary and alternative path configuration for all devices with the **lsdev** command. Two or more logical devices can be configured for a single physical device, but the first device that is configured is labeled the primary device. All other logical devices that are configured after the first device are labeled as alternative devices. To see this information, run the **lsdev -Cc tape** command and look at the location field in the data. Run the following command,

```
lsdev -Cc tape | grep P
```

For example, you can easily determine how many physical devices are configured with path failover support.

**Note:** Show the primary and alternative path configuration for any device by using tape diagnostic and utility functions. Refer to [“IBM Tape Diagnostic Tool \(ITDT\)” on page 76](#).

## Configuring and unconfiguring primary and alternative devices

Logical devices that are configured as alternative paths can be unconfigured and reconfigured at any time after the initial configuration is run. Unconfiguring an alternative path device removes that device from the primary device path list, removes the **-A** or **-ALT** appended to the location field, and changes the device to the **Defined** state. The primary and any other alternative devices are still available.

Likewise, configuring a new alternative path device or reconfiguring an existing one in the **Defined** state adds that device to the primary device path list, appends **-A** or **-ALT** to the location field, and makes the device available.

Logical devices that are configured as primary paths can also be unconfigured and reconfigured at any time after initial configuration is run. However, the operation is different for a primary device. When a primary device is unconfigured, the following events occur.

1. All alternative devices are unconfigured as described previously.
2. The primary device is unconfigured.
3. The **-P** or **-PRI** appended to the location field is removed.
4. The device is changed to the **Defined** state.
5. All alternative devices that were unconfigured are reconfigured. The first device that is reconfigured becomes the new primary device. All remaining alternative devices are reconfigured as alternative paths.

These methods can unconfigure and reconfigure physical devices on the system when device connections or addressing changes are made.

## Data Path failover and load balancing support for tape drives

### Note:

1. Some devices require a path failover feature code that is installed before the path failover support is enabled in the Atape device driver. Refer to [“Automatic failover” on page 5](#) for what feature code might be required for your machine type.
2. DPF keys do not need to be added if you are running the latest drive code on Ultrium 3 and Ultrium 4 drives.
3. This function is not supported for devices that are attached through an IBM San Data Gateway or on the IBM Virtualization Engine TS7510.
4. The AIX operating system supports only a static configuration of devices, which also applies to the Path Failover and Failover Support. When devices are initially configured at a specific SCSI ID and physical connection (drive port, host bus adapter, and switch number/port, if applicable) and in the Available state, changing the physical device address/connection without either rebooting or unconfiguring and reconfiguring the devices has unpredictable results and is not supported.

## Installing Data Path failover license key

Use the following command line script to query, add, or delete license keys for this feature before the path failover feature is enabled as described below. The key is a 16-digit hexadecimal value, for example, 1234567890abcdef.

All key values “A-F” must be entered in lowercase letters as “a-f.”

- Query installed keys: `dpf_keys`
- Install a license key: `dpf_keys -a key`
- Delete a license key: `dpf_keys -d key`

## Configuring and unconfiguring path failover support

Path failover support is not enabled automatically when the device driver is installed. It must be configured initially on each logical device after installation. When path failover support is enabled for a logical device, it remains set until the device is deleted or the support is unconfigured. The path failover setting is retained even if the system is rebooted.

Path failover support can be enabled on all configured devices at one time, or it can be enabled or disabled selectively by logical device. It might be desirable at times to configure some, but not all, logical paths to a device with the support enabled.

To enable the support globally on all currently configured devices, run the command

```
/usr/lpp/Atape/instAtape -a
```

This action unconfigures all devices that have path failover set to **No**, and reconfigures all devices, setting path failover to **Yes**.

To enable or disable the support on a single logical device, use the **SMIT** menu to **Change/Show Characteristics of a Tape Drive**, then select **Yes** or **No** for **Enable Path Failover Support**. The support can also be enabled or disabled by using the **chdev** command, for example:

```
chdev -l rmt0 -aalt_pathing=yes
chdev -l rmt0 -aalt_pathing=no
```

## Primary and alternative paths

When the device driver configures a logical device with path failover support enabled, the first device that is configured always becomes the primary path and PRI is appended to the location field of the device. When a second logical device is configured with path failover support enabled for the same physical device, it configures as an alternative path and ALT is appended to the location field. A third logical device is configured as the next alternative path with ALT appended, and so on. The device driver supports up to 16 physical paths for a single device.

For example, if rmt0 is configured first, then rmt1, the **lsdev -Cc** tape command output is similar to the following command.

```
rmt0 Available 20-60-01-PRI IBM 3590 Tape Drive and Medium Changer (FCP)
rmt1 Available 30-68-01-ALT IBM 3590 Tape Drive and Medium Changer (FCP)
```

If rmt1 is configured first, then rmt0, the command output is similar to the following.

```
rmt0 Available 20-60-01-ALT IBM 3590 Tape Drive and Medium Changer (FCP)
rmt1 Available 30-68-01-PRI IBM 3590 Tape Drive and Medium Changer (FCP)
```

The labeling of a logical device as either a primary or alternative path is for information only, to

1. Identify the actual number of physical devices that are configured on the system and a specific logical device that is associated with them. Only one logical device is labeled the primary path for each physical device. However, many (multiple) logical devices can be labeled as an alternative path for the same devices.
2. Provide information about which logical devices configured on the system have path failover support enabled.

## Querying primary and alternative path configuration

You can see the primary and alternative path configuration for all devices with the **lsdev** command. Two or more logical devices might be configured for a single physical device, but the first device that is configured is labeled the primary device. All other logical devices that are configured after the first device are labeled as alternative devices.



To see this information, run the **lsdev -Cc tape** command and look at the location field in the data. By running `lsdev -Cc tape | grep PRI`, for example, you can easily determine how many physical devices on the RS/6000 or System p (also known as pSeries) server are configured with path failover support.

## Configuring and unconfiguring primary and alternative devices

Logical devices that are configured as alternative paths can be unconfigured and reconfigured at any time after the initial configuration is run. Unconfiguring an alternative path device removes that device from the primary device path list, removes the **ALT** appended to the location field, and changes the device to the **Defined** state. The primary and any other alternative devices are still available. Likewise, configuring a new alternative path device or reconfiguring an existing one in the **Defined** state adds that device to the primary device path list, appends **ALT** to the location field, and makes the device available.

Logical devices that are configured as primary paths can also be unconfigured and reconfigured at any time after initial configuration is run. However, the operation is different for a primary device. When a primary device is unconfigured, the following events occur.

1. All alternative devices are unconfigured as described previously.
2. The primary device is unconfigured.
3. The **PRI** appended to the location field is removed.
4. The device is changed to the **Defined** state.
5. All alternative devices that were unconfigured are reconfigured. The first device that is reconfigured becomes the new primary device. All remaining alternative devices are reconfigured as alternative paths.

These methods unconfigure and reconfigure physical devices on the system when device connections or addressing changes are made.

## System-managed encryption

### Device driver configuration

System-managed encryption can be set on a specific tape drive by using the standard **SMIT** panels to **Change/Show Characteristics** of a tape device or the command line **chdev** command. There are two new attributes added for encryption.

sys_encryption "yes/no" Manager	Use System Encryption FCP Proxy
wrt_encryption "off/on/custom"	System Encryption for Write Commands at BOP

The `sys_encryption` attribute enables device driver system-managed encryption for a tape drive by setting the value to **yes**.

The `wrt_encryption` attribute controls whether the device driver can set the tape drive to encryption enabled for write commands. When set to **off**, the tape drive uses encryption for read operations; write operations do not use encryption. When set to **on**, the tape drive uses encryption for both read/write operations. When set to **custom**, the device driver does not modify current tape drive setting. The custom setting is intended for applications that use system-managed encryption to control write encryption without device driver intervention.

**Note:** If `wrt_encryption` is set to **on**, an application cannot open a tape drive by using the append mode.

### Querying tape drive configuration

Querying the tape drive configuration is a tape diagnostic and utility function. Refer to [“IBM Tape Diagnostic Tool \(ITDT\)”](#) on page 76.

## Testing data encryption configuration and connectivity

A data encryption test is available to validate the **ibmekm.conf** file server entries and test tape drive to server connectivity operations.

This test is a tape diagnostic and utility function. Refer to [“IBM Tape Diagnostic Tool \(ITDT\)” on page 76](#).

### Error logging

Encryption errors are logged along with other tape operation errors by using the standard TAPE\_ERR1 Template “Tape Operation Error” with associated sense data in the detail data.

An encryption failure is indicated when the **asc/ascq** in the sense data is EFxx or EExx. Refer to the tape drive hardware reference for information on the **asc/ascq** being reported. The **asc/ascq** can be found in the first column of the second row in detail sense data. For example,

```
Detail Data
SENSE DATA
0A00 0000 5A08 25FF 0000 00FF FF00 0000 0000 0000 F000 0600 0000 1458 0000 0000
EF11 FF02 D105 0000 0009 0191 0002 0000 0000 0A00 0000 0000 0000 0000 0000
0000 0000 0000 FFFF FF00 0000 FFF0 B7E3 0001 2127 0000 0000 0000 0000 3930 3220
2020 2000 0041 4A00 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
```

### Field support information

When encryption failures require field support or development analysis, the following data must be provided for a problem on a specific tape drive from the machine (rmt1 for example) for the device driver. Tape drive dumps and EKM server logs might be needed in addition to this information.

```
errpt -a > errpt.rmt1
lsattr -El rmt1 > lsattr.rmt1
All Atape files in /var/adm/ras/Atape*
```

## Problem determination

A set of tools is provided with the device driver to determine whether the device driver and the tape device are functioning correctly. The standard AIX interface is provided for problem determination.

### Dump support

Dump support is provided through the dump entry point in the driver. Refer to appropriate AIX manuals for a description of how to use the dump devices and how to read the dump data.

### Dump device commands

To list the current dump devices, enter the following command,

```
sysdumpdev -l
```

To establish the **rmt1** tape device as a secondary dump device, enter the following command,

```
sysdumpdev -s /dev/rmt1
```

To run a dump operation, use the **sysdumpstart** command. To send the dump data to the secondary dump device, enter the following command:

```
sysdumpstart -s
```

**Note:** This command stops the system. Use the **sync** command to ensure that the cache is flushed before the **sysdumpstart -s** command is issued.

To list the last dump data, enter the following command,

```
sysdumpdev -z
```

After the dump data is placed on the tape, copy it to a file on the disk before the **crash** command is used to process it. For example,

```
dd if=/dev/rmt1 of=tapedump1 ibs=4096 obs=512  
crash tapedump1
```

**Note:** The *ibs* value is the input block size.

If the block size of the tape device is larger than the block size sent during the dump process, the dump operation fails. Set the block size to zero on the tape device and experiment with the *ibs* value for the **dd** command.

## Device and volume information logging

An optional logging utility is provided to log the information about the device and the media. The information is extensive for some devices and limited for other devices. If set to **On**, the logging facility gathers all available information through the **SCSI Log Sense** command.

This process is a separate facility from error logging. Error logging is routed to the system error log. Device information logging is sent to a separate file.

The following parameters control this utility,

- Logging
- Maximum size of the log file
- Volume ID for logging

Refer to [“Tape drive, media, and device driver parameters” on page 16](#) for a description of these parameters.

Each time the rewind and unload sequence occurs or the **STIOC\_LOG\_SENSE ioctl** command is issued, an entry is added to the log. Each time a new cartridge is loaded, the values in the device log buffers are reset with the **Log Sense** command. The log data is gathered on a per-volume basis.

## Log file

The data is logged in the `/usr/adm/ras` directory. The file name is dependent on each device; therefore, each device has a separate log. An example of the **rmt1** device file is

```
/usr/adm/ras/Atape.rmt1.log
```

The files are in binary format. Each entry has a header followed by the raw **Log Sense** pages as defined for a particular device.

The first log page is always page 0x00. This page, as defined in the SCSI-2 ANSI specification, contains all pages that the device supports. Page 0x00 is followed by all pages that are specified in page 0x00. The format of each following page is defined in the SCSI specification and the device manual.

## Tape log utility

A tape log utility is installed with the *tapelog* device driver that displays the contents of the log file in ASCII text. The log pages are shown as hexadecimal values in dump format.

The syntax for the tape log utility is

```
tapelog -l Name [-d] or tapelog -f File [-d]
```

**Note:**

1. **Name** is the logical name of the device (such as **rmt0**).
2. **File** is the name of a log file (such as `Atape.rmt0.log`).
3. The **-d** parameter, if used, deletes the log file for the specified device.

The contents of the log file are displayed as standard output. To save the log in a file, use the AIX redirection function. For example,

```
tapelog -l rmt0 > rmt0.log
```

## Reservation conflict logging

When the device driver receives a reservation conflict during open or after the device is opened, it logs a reservation conflict in the AIX error log. Before it logs the error, the device driver issues a **Persistent Reserve In** command to determine whether a SCSI Persistent Reservation is active on the reserving host to get the reserving host initiator WWPN (worldwide port name) and reserve key. If successful, the device driver logs this information as follows,

```
Reserving host key xxxxxxxx WWPN xxxxxxxx
```

Where the first xxxxxxxx is the actual reserve key, and the second xxxxxxxx is the reserving host initiator WWPN.

After the reserving host WWPN is initially logged, subsequent reservation conflicts from the same reserving host WWPN are not logged. This action prevents multiple entries in the error log until either the reserving host WWPN is different from the one initially logged or the device driver reserved the device and another reservation conflict occurs.

If the **Persistent Reserve In** command fails, the detail data contains the following entry with the `errno` from the failing **Persistent Reserve In** command.

```
Unable to obtain reserving host information, errno x
```

The possible *errno* values are

- ENOMEM - Device driver cannot obtain memory to run the command
- EINVAL - Device has a Persistent Reservation but does not support the Persistent Reserve In service action
- EBUSY - Device failed the command with reservation conflict and has an SCSI-2 Reserve active
- EIO - Unknown I/O failure occurred

## Error logging

The device driver provides logging to the AIX system error log for various errors. The error log can be viewed for specific devices by using the **Error Log Analysis** utility that is provided with the tape drive service aids. Refer to [“Error Log Analysis” on page 38](#). The error log can also be viewed by using the **smitt** or the **errpt** command.

## Error log templates

The error log templates the device driver uses follow the same format as the default AIX tape error log entries. Each error log entry is identified by an error label and contains detail data that is associated with the type of error. The following items describe the error labels and detail data for the templates that are used for logging tape device, media, and SCSI adapter-related errors in the AIX system error log.

## Error labels

Errors are logged with an associated error label and error ID. The error label indicates the basic type of error.

- TAPE\_ERR1  
Tape media error
- TAPE\_ERR2  
Tape hardware error
- TAPE\_ERR4  
SCSI Adapter detected error
- TAPE\_ERR5  
Unknown error
- RECOVERED\_ERROR  
Temporary tape hardware or media error
- SIM\_MIM\_RECORD\_3590  
3590 Service/Media Information Message (Log Page X '31')
- TAPE\_SIM\_MIM\_RECORD  
Tape drive Service/Media Information Message (Log Page X '31')
- DEV\_DUMP RETRIEVED  
Device dump retrieved
- TAPE\_DRIVE\_CLEANNING  
Tape drive needs cleaning
- RESERVE\_CONFLICT  
Reservation conflict

## Detail data

Detail data is logged with the associated error that identifies the cause of the error. Detail data for the SIM\_MIM\_RECORD\_3590 or TAPE\_SIM\_MIM\_RECORD entries contain the raw data from Log Sense Page 31. Refer to the hardware reference manual for the format of this entry. All other error log entries use the following format for detail data:

```
Detail Data
SENSE DATA
aabb xxxx ccdd eeee eeee eeee eeee eeee ffgg hhxx ssss ssss ssss ssss ssss
ssss ssss ssss ssss ssss ....
```

### aa

Length of the command descriptor block (CDB).

### bb

SCSI target address.

### xx

Unused or reserved.

### cc

Start of CDB, *cc* is the operation code (byte 0).

### dd

Logical unit (byte 1) in the CDB.

### ee

Bytes 2 - 12 in the CDB.

### ff

Status validity field. If this field is 01, then a SCSI error was reported, and byte *gg* indicates the type of error. If this field is 02, then an adapter error was reported, and byte *hh* indicates the type of error.

## gg

This byte indicates the type of SCSI error that occurred.

- 02 CHECK CONDITION - Device reported a check condition.
- 08 BUSY STATUS - Target is busy.
- 18 RESERVATION CONFLICT - Target is reserved to another initiator.
- 22 COMMAND TERMINATED - Device terminated the command.
- 28 QUEUE FULL - Device command queue is full.

## hh

This byte indicates the type of adapter error that occurred. For parallel SCSI adapters, this byte is the `general_card` status code as defined in `/usr/include/sys/scsi.h`

- 01 HOST IO BUS ERROR - Host I/O bus error during data transfer.
- 02 SCSI BUS FAULT - SCSI bus protocol or hardware error.
- 04 COMMAND TIMEOUT - Command timed out before completion.
- 08 NO DEVICE RESPONSE - Target did not respond to selection phase.
- 10 ADAPTER HARDWARE FAILURE - Adapter indicated a hardware failure.
- 20 ADAPTER SOFTWARE FAILURE - Adapter indicated a microcode failure.
- 40 FUSE OR TERMINAL PWR - Blown terminator fuse or bad termination.
- 80 SCSI BUS RESET - Adapter indicated that SCSI bus was reset.

For FCP or SAS adapters, this byte is the `adapter_status` code as defined in `/usr/include/sys/scsi_buf.h`

- 01 HOST IO BUS ERROR - Host I/O bus error during data transfer.
- 02 TRANSPORT FAULT - Failure in the transport layer.
- 03 COMMAND TIMEOUT - Command timed out before completion.
- 04 NO DEVICE RESPONSE - Target did not respond to attempts to select it.
- 05 ADAPTER HARDWARE FAILURE - Adapter indicated a hardware failure.
- 06 ADAPTER SOFTWARE FAILURE - Adapter indicated a microcode failure.
- 07 WW NAME CHANGE - Adapter detected a new worldwide name for the device.
- 08 FUSE OR TERMINAL PWR - Blown terminator fuse or bad termination.
- 09 TRANSPORT RESET - Adapter detected an external SCSI bus reset.
- 0A TRANSPORT BUSY - The transport layer is busy.
- 0B TRANSPORT DEAD - The transport layer is inoperative.

## ss

If byte `gg` indicates a check condition, the `ss` byte is the sense data from the device. Refer to the appropriate device reference manual for the specific format and content of these bytes.

## Automatic dump facility

The device driver provides an automatic dump facility for devices that support reading a dump and indicating when a dump is available in device sense data. Whenever a check condition occurs and the sense data indicates that a dump is available, the device driver reads the dump from the device and stores it in the `/var/adm/ras` directory. A maximum of five dumps for each device are stored in this directory as

```
Atape.xmtx.dump1
Atape.xmtx.dump2
Atape.xmtx.dump3
```

**Note:** X is the device number, for example, `rmt0`.

When the device is first configured, the dump name is set to **dump1**. If more than three dumps occur, the driver starts over at **dump1**; therefore the last three dumps are always kept.

## Dynamic Runtime Attributes

Field issues can occur where customers must know which initiator on a host is holding a reservation in a drive or preventing the media from being unloaded. Also, they must correlate which special file name is used for the drive (such as **rmt2**). Sometimes this issue occurs over transport bridges and translators, thus losing any transport identifiers to help this effort. LTO5, 3592 E07, 3592 E08, and later physical tape drives support attributes set to the drive dynamically by a host. This function is called Dynamic Runtime Attributes (DRA). Atape supports the feature, starting from 12.5.9.0.

The feature is enabled by default. The drive attributes on the host are set during the open, close, device reset, and data path change only. So, it does not have impact on the tape and system performance. The error is ignored and is not returned to application, when the information failed to send. However, a device attribute named as "host\_attributes" is available to disable the DRA feature, when Atape runs with the virtual tape library for which the feature is not supported.

Run the **lsattr** command to display the attribute of host\_attributes setup.

On LTO tape drive:

```
# lsattr -El rmt1
host_attributes yes      Host Dynamic Runtime Attribute (LTO-5 and later only) True
```

On 3592 tape drive:

```
# lsattr -El rmt4
host_attributes yes      Host Dynamic Runtime Attribute for 3592-E07 and later True
```

To modify the attribute of "host\_attribute", run the **chdev** command.

```
# chdev -l rmt1 -a host_attributes=no
rmt1 changed
```

## Trace facility

The AIX trace facility is supported for the device driver. The trace event is identified with a *hookword*. The hookword that is used by the device driver is 326. The trace can be initiated at any time before an operation on a tape device.

Enter the following AIX command to start the trace.

```
trace -a -j 326
```

This command starts the trace in the background and collects only the trace events with the 326 hookword (*Atape* device driver).

Enter the following AIX command to stop the trace.

```
trcstop
```

This command stops the trace after the tape operations are run.

Enter the following AIX command to view the trace.

```
trcrpt > trace.out
```

This command formats the trace output into a readable form and places it into a file for viewing.

## Atape System Trace (ATRC) utility

The **atrc** trace utility is also installed with the device driver to start, stop, and format a device driver trace. To start the trace, enter the **atrc** command. To stop and format the trace, enter the **atrc** command again. The trace is formatted to an **atrc.out** AIX file in the current directory.

## Component tracing

Later releases of AIX 5.3 and above support component tracing. Unlike system tracing that must be started and stopped, component tracing by default is on all the time and runs continually.

To determine whether component tracing is available, run the command **ctctrl -q** to display a list of supported components with their default settings. You must have root authority to run this command. Refer to the AIX **ctctrl man** page for a complete description of the **ctctrl** command and parameters.

To dump and format the current component trace to a file (for example, **actrc.out**) into the current directory, run the following commands.

```
ctctrl -D -c Atape
trcrpt -l Atape -o actrc.out
```

The Atape component trace can also be retrieved from a system dump. This action eliminates the need to start the Atape system trace before a system dump or to re-create an AIX system dump when a system trace is not running. The AIX system dump is normally stored in the **/var/adm/ras** directory as a **vmcore.x.BZ** file, where x is a dump number 1, 2, and so on.

To retrieve and format the Atape component trace from a dump file (for example, **vmcore.1.BZ**) to a file (for example, **actrc.dump**) into the current directory, run the following commands.

```
dmpuncompress /var/adm/ras/vmcore.1.BZ
trcdead -c /var/adm/ras/vmcore.1
trcrpt -l Atape -o actrc.dump
```

## Atape Component Trace (ACTRC) utility

The **actrc** component trace utility is also installed with the device driver to dump and format the current Atape component trace. To dump and format the component trace, run the command **actrc**. The trace is formatted to an **actrc.out** file in the current directory.

## Tape drive service aids

The service aids described in the following sections are accessible through the AIX diagnostic subsystem by using the AIX **diag** command and in the **Task Selection** menu by selecting **IBM Tape Drive Service Aids**. Refer to [“Tape drive service aids details”](#) on page 36.

Tape drive service aids are also available by using the [“IBM Tape Diagnostic Tool \(ITDT\)”](#) on page 76.

## Tape drive service aids details

The following service aid utilities are installed with the device driver:

- Force Microcode Dump
- Read Dump
- Microcode Load
- Error Log Analysis
- Reset Drive
- Create an FMR Tape



## Force Microcode Dump

This utility forces a dump operation on the tape drive. After the dump operation is completed, the dump data can be transferred from the tape drive by using the **Read Dump** utility.

To access this utility -

1. Open the **Service Aids** window.
2. Select **Force Microcode Dump** from the **IBM Tape Drive Service Aids** window, then press **Enter**.
3. Select the device from the **IBM Tape Drive Selection** window, then press **Enter**.

The **Force Microcode Dump** operation starts, and a window opens when the operation is completed.

## Read Dump

This utility transfers the dump data from the device to a file, a diskette, or a tape cartridge.

To access this utility -

1. Open the **Service Aids** window.
2. Select **Read Dump** from the **IBM Tape Drive Service Aids** window, then press **Enter**.
3. Select the device from the **IBM Tape Drive Selection** window, then press **Enter**.
4. Enter the destination file name or device in the **Prompting for Destination** window. The default destination is the `/dev/rfd0` diskette drive. To transfer the dump data to a tape cartridge, enter the device name of the tape drive (for example, `/dev/rmt0`). To transfer the dump data to a file, enter the file name. Press **F7** to commit.

**Note:** On certain terminal types, it might be necessary to press **Esc** and the number **7** instead of **F7**.

The **Read Dump** operation starts, and a window opens when the operation is completed.

## Microcode Load

This utility downloads microcode to the device from a file or a diskette (AIX format only).

**Note:** To download the microcode from a DOS diskette, you must first use the AIX **dosread** command to transfer the file from the DOS diskette to the AIX file. Then, you can use the **Microcode Load** utility to download the AIX file to the tape drive.

To access this utility -

1. Open the **Service Aids** window.
2. Select **Microcode Load** from the **IBM Tape Drive Service Aids** window, then press **Enter**.
3. Select the device from the **IBM IBMTape Drive Selection** window, then press **Enter**.
4. Enter the source file name or device on the **Prompting for Source File** window. The default source is the `/dev/rfd0` diskette drive. To load from a file, enter the file name. Press **F7** to commit.
5. If the current microcode on a tape drive is Federal Information Processing Standard (FIPS) code, then a window opens and displays the following message.

Warning: The drive is currently using FIPS code. Press Enter to continue with downloading new drive code.

If you do not want to download the new code, press either **F3** to cancel or **F10** to exit without downloading new code. Otherwise, press **Enter** to continue with the download code procedure.

The **Microcode Load** operation starts, and a window opens when the operation is completed.

## Error Log Analysis

This utility displays and analyzes the system error log entries for a specific tape drive and can be used for problem determination. The type of error, the SCSI command, and the sense data (if applicable) are displayed for each entry in the error log (one screen at a time).

To access this utility -

1. Open the **Service Aids** window.
2. Select **Error Log Analysis** from the **IBM Tape Drive Service Aids** window, then press **Enter**.
3. Select the device from the **IBM Tape Drive Selection** window, then press **Enter**.
4. If entries are listed in the error log for the selected device, then the first entry is displayed. Press **Enter** to display the next entry.
5. After all entries are displayed, a window opens, and the operation is completed.

## Reset Drive

This utility resets the tape drive.

To access this utility -

1. Open the **Service Aids** window.
2. Select **Reset Drive** from the **IBM Tape Drive Service Aids** window, then press **Enter**.
3. Select the device from the **IBM Tape Drive Selection** window, then press **Enter**.

The **Reset Drive** operation starts, and a window opens when the operation is completed.

## Create an FMR Tape

This utility creates a field microcode replacement (FMR) cartridge tape by using the loaded functional microcode in the tape drive.

To access this utility -

1. Open the **Service Aids** window.
2. Select **Create an FMR Tape** from the **IBM Tape Drive Service Aids** window, then press **Enter**.
3. Select the device from the **IBM Tape Drive Selection** window, then press **Enter**.

The **Create an FMR Tape** operation starts, and a window opens when the operation is completed.

## Performance considerations

This chapter describes the parameters and issues that can affect the perceived performance of the tape drive. In general, AIX applications that operate at a file level to move data between disk storage devices and tape do not use the full capabilities of a high end tape device. The goal of this discussion is to give an overview of the data path components that are involved in moving data between disk storage devices and tape. The following chapter describes basic techniques and common utilities in a specific environment that can be used to understand how a device is performing. Performance issues that are encountered by advanced application developers are beyond the scope of this document.

- Refer to the hardware reference for the specific device for performance specifications.
- Refer to the application documentation for information on device-specific application configuration.
- Refer to the operating system documentation for information on disk storage device striping and other techniques for improving file system performance.

## Data path

The simplified model in [Figure 8 on page 39](#) shows the components that are involved in the data path for moving data at a file level between disk storage devices and tape.

Performance analysis must be approached by determining which component of the data path impacts performance. Typically, a performance problem can be isolated by looking at one leg of the data path at a time. The goal of this analysis is to confirm that the tape data path is not impacting the performance adversely.

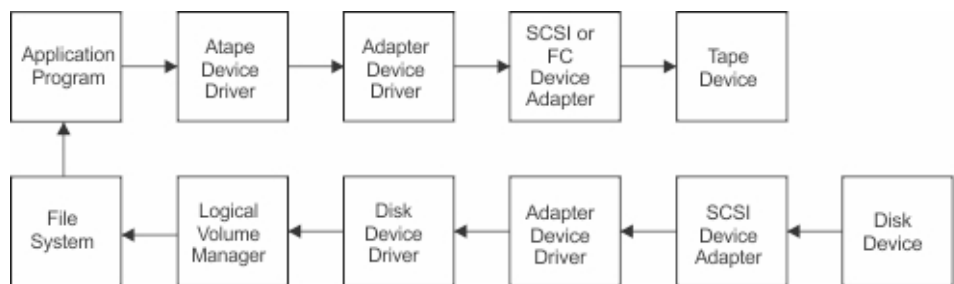


Figure 8. Data path for AIX device driver (Atape)

## Common AIX utilities

The most commonly reported cause for poor tape performance is the use of small block sizes or the modification of the installation defaults for the tape device.

**Note:** The device parameters must not be changed from the defaults for most applications.

The following guidelines typically result in good tape path performance for use with AIX utilities:

1. Hardware compression must be enabled for maximum performance if the data sent to the device is uncompressed.
2. The **block\_size** parameter must be set to variable (*block\_size=0*) and command or application parameters that are specified to a block size appropriate for the device.
3. Block sizes of 128 KB or greater must be used to improve performance.

## AIX iostat utility for tape performance

In releases of AIX 5.3 and earlier, the AIX **iostat** utility supports tape performance statistics in addition to other supported devices (such as disk). To determine whether the **iostat** utility supports the configured tape drives, run the command **iostat -p**. If the configured tape drives are supported, a list of configured tape drives are displayed with the statistics listed for each drive. Refer to the AIX **iostat man** page for a complete description of the **iostat** command and parameters. When the **Data Path Failover** feature is used, only the primary path for the tape drive is listed. The statistics apply to both the primary and alternative paths that are used.

## Before Support is called

System performance tuning is not a support responsibility. If tests indicate that raw tape performance is below specifications, record the exact failing command. Then, collect the output from the commands in Table 5 on page 39 before support is contacted.

Table 5. Error description	
Information	Command
Configuration	<b>lscfg -v</b>
Device parameters	<b>lsattr -E -l rmtN</b>
Error log. Call hardware support if errors are found for TAPE_ERR* or SCSI* error labels.	<b>errpt -a</b>
Driver version	<b>lspp -l Atape.driver</b>

Table 5. Error description (continued)	
Information	Command
Trace of failing command	Refer to <a href="#">“Trace facility”</a> on page 35

## Linux Tape and Medium Changer device driver

This chapter describes the IBM Linux Tape and Medium Changer device driver (`lin_tape`).

For tape diagnostic and utility functions, refer to [“IBM Tape Diagnostic Tool \(ITDT\)”](#) on page 76.

### Purpose

The `lin_tape` and medium changer device driver is designed to take advantage of the features that are provided by the IBM tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape operations (such as backup and restore) and medium changer operations (such as mount and unmount the cartridges), and also the advanced functions that are needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application.

### Data flow

The software that is described in this chapter covers the Linux device driver (`lin_tape` device driver) and the interface between the application and the tape device.

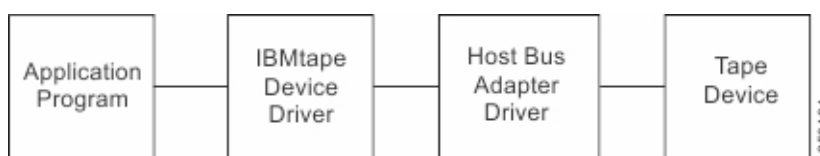


Figure 9. Data flow for Linux device driver (`lin_tape`)

## Product requirements

### Hardware requirements

For current hardware requirements, refer to the [“Hardware requirements”](#) on page 2.

### Software requirements

For current software requirements, refer to the [“Software requirements”](#) on page 2.

## Installation and Configuration instructions

The `lin_tape` device driver for Linux is provided in a source `rpm` package. The utility tools for `lin_tape` are supplied in binary `rpm` packages. Refer to [“Accessing documentation and software online”](#) on page 77. They are downloaded with the driver.

The following sections describe installation, configuration, uninstallation, and verification procedures for `lin_tape` and its utility tools. Refer to Linux documentation for `tar` command information and any Linux distribution that support `rpm` for `xpm` command information. You must have `root` authority to proceed with the installation of the driver. See the README file that can be downloaded with the driver at Fix Central. For information about downloading drivers, see [“Accessing documentation and software online”](#) on page 77.

This file contains the latest driver information and supersedes the information in this publication.

## Conventions used

In subsequent pages, you see file names with x.x.x in them. The x.x.x refers to the version of the driver, which changes as IBM releases new driver levels. Use the actual driver version numbers as you complete the instructions.

Commands that you are to type are indicated with a leading ">" character, which indicates the shell prompt.

**Note:** This procedure is run with tape diagnostic and utility functions. See [“IBM Tape Diagnostic Tool \(ITDT\)”](#) on page 76.

## Configuration limitations

Maximum supported number of tape devices	1024
Maximum supported number of HBA ports	16 (8 dual-port, 4 quad-port)
Maximum supported number of paths for a device (DPF/CPF)	16/16
Maximum LUNs per system	256

Every attached tape or library device uses a certain amount of resources. The user must consider resources such as physical memory and virtual space on the system, which might further limit the number of devices that can be attached.

## Components created during installation

The lin\_tape package consists of the device driver and a number of associated files. Components that are created during **lin\_tape** installation (from the **rpm** package) are listed in [Table 6 on page 41](#).

Table 6. Linux: Components created during lin_tape installation	
Component	Description
<b>/lib/modules/(Your system's kernel name)/kernel/drivers/scsi/ lin_tape.ko</b> <b>/lib/modules/(Your system's kernel name)/kernel/drivers/scsi/pfo.ko</b> Only when sfmp flag is used during build: <b>/lib/modules/(Your system's kernel name)/kernel/drivers/scsi/ sgmp.ko</b> <b>/lib/modules/(Your system's kernel name)/kernel/drivers/scsi/stmp.ko</b>	Device driver module for current kernel version
<b>/usr/bin/lin_taped</b>	lin_taped daemon
<b>/etc/lin_taped.conf</b>	lin_taped daemon configuration file
<b>/usr/share/doc/lin_tape/lin_tape_daemon.Readme</b> (for Red Hat) <b>/usr/share/doc/packages/lin_tape/lin_tape_daemon.Readme</b> (for SUSE LINUX)	Readme file for lin_taped daemon
<b>/usr/share/doc/lin_tape/lin_tape.Readme</b> (for RedHat) <b>/usr/share/doc/packages/lin_tape/lin_tape.Readme</b> (for SUSE LINUX)	Readme file for lin_tape driver

Table 6. Linux: Components created during lin_tape installation (continued)	
Component	Description
/usr/share/doc/lin_tape/copying (for Red Hat)	License documentation for lin_tape
/usr/share/doc/packages/lin_tape/copying (for SUSE LINUX)	

**Note:** On a Unified Extensible Firmware Interface secure boot enabled kernel, modules are required to be signed before they can be loaded. Refer to Table 6 on page 41 to identify the device driver modules and their location after installation. Proceed to sign the device driver modules by following the operating system instructions while keeping the modules at the same location.

## Installation procedure

If **lin\_tape** is already installed on your system, refer to the “[Updating procedure](#)” on page 43. This section assumes that you are installing the **lin\_tape** device driver onto a system where it is not currently installed.

If you are installing **lin\_tape** on a system that is running Linux for S/390® or Linux for zSeries, ensure that the OpenFCP adapter device driver **zfcp** is loaded in the kernel. Refer to “[Configuring Tape and Medium Changer devices on IBM System z models](#)” on page 44 for how to configure and install **zfcp**.

Make sure that the C/C++ development and kernel development packages are installed on your system. To install the **lin\_tape** driver with all the added value of the lin\_taped daemon, complete the following steps.

1. Download the appropriate level of the source **RPM** package to a directory of your choice on the Linux kernel for which you want to install it.
2. Run **rpmbuild --rebuild <filename>**, where: *<filename>* is the name of the RPM file. A binary RPM package is created for your kernel from the source RPM package. For example,

```
>rpmbuild --rebuild lin_tape-1.x.x.x.0-1.src.rpm
```

**Note:** For the current **lin\_tape** driver, it is possible to enable path failover for *st/sg* interfaces. See [Enabling path failover for st and sg interface](#) for details. If path failover is enabled and you want to also enable it for *st* or *sg* devices, an extra flag that is named *sfmp* must be added at this step:

```
>rpmbuild --rebuild -with sfmp lin_tape-1.x.x.x.0-1.src.rpm
```

3. Output from the build is printed to your screen. Near the end of the output, a line indicates the file name and location of your binary RPM package. For example, a line similar to the following is output to your screen.

```
Wrote: /root/rpmbuild/RPMS/s390x/lin_tape-1.x.x.x.0-1.s390x.rpm
```

4. To install the **lin\_tape** driver from the binary package, run **>rpm -ivh <filename>** For example,

```
>rpm -ivh /root/rpmbuild/RPMS/s390x/lin_tape-1.x.x.x.0-1.s390x.rpm
```

**Note:** For Ubuntu, run **alien -i --scripts <filename>** (starting with lin\_tape version 3.0.33).

5. To install the lin\_taped daemon, download it to your Linux file system and run rpm **-ivh** on the daemon RPM file. For example,

```
>rpm -ivh ../lin_taped-1.x.x.x.0-rhel9.s390x.rpm
```

**Note:** For Ubuntu, run **alien -i --scripts <lin\_taped\*.rpm>**, and then run **sudo update-rc.d lin\_tape defaults**.

## Updating procedure

If your current **lin\_tape** device driver was installed from an **rpm** package previously, you can uninstall the driver first, then install the newer version. For example,

On SLES and RHEL:

```
>rpm -e lin_taped  
>rpm -e lin_tape
```

Then, follow the installation procedure in the previous section.

On Ubuntu:

```
apt purge lin-taped  
apt purge lin-tape
```

Then, follow the installation procedure in the previous section.

**Note:** All tape devices that use the **lin\_tape** device driver must be closed and cannot be in use when **lin\_tape** is uninstalled.

## Querying the installed package

The query is supported for the **lin\_tape** device driver **rpm** package and SLES and RHEL only.

The installed **rpm** package can be queried by running the following commands to show information that is associated with the package.

To show information about **lin\_tape** -

```
>rpm -qi lin_tape
```

To show the file list, enter the command

```
>rpm -ql lin_tape
```

To show the states of files in the package, for example, *normal*, *not installed*, or *replaced* -

```
>rpm -qs lin_tape
```

To query **lin\_tape** package on Ubuntu:

```
dpkg-query --list lin-tape
```

## Configuring Tape and Medium Changer devices on Intel-compatible systems

Physically attach your tape and medium changer devices to your Linux server.

After the driver software is installed and a tape device is connected to the adapter, the device can be configured and made available for use. Access to the device is not provided until the device is configured.

If your system is attached to a tape library with the integrated router, before the QLogic driver is installed, set the host type of the router to *solaris* and make sure that the logical unit numbers of the control unit, medium changer, and the connected tape drives are contiguous. Otherwise, the QLogic device driver does not recognize all of the attached devices. To view the LUNs of attached devices, log on to the router and use the **fcShowDevs** command. If the LUNs are not contiguous, use the **mapCompressDatabase** command to delete the invalid LUNs and make the valid LUNs contiguous.

When you run the **lin\_tape** kernel module, it creates special files in the */dev* directory.

## Configuring Tape and Medium Changer devices on IBM System p models

Follow the same instructions as documented in the previous section. You must configure the Emulex Linux device driver if you have Fibre Channel tape devices that are attached to your System p (also known as pSeries) system.

## Configuring Tape and Medium Changer devices on IBM System z models

The Fibre Channel topology that is supported for System z® is point-to-point and fabric. Refer to the Linux on System z Fibre Channel documents for details on the supported configurations for Fibre Channel device attachment. The Linux Fibre Channel adapter device driver **zfcp** is available in the kernel that supports zSeries Fibre Channel Protocol. The zfcp device configuration methods in 2.6 (and higher) and 2.4 kernels are different. For 2.6 kernels and higher, refer to appropriate chapter in the Linux on System z document entitled *Linux on System z: Device Drivers, Features, and Commands*.

For 2.4 kernels, there are three ways to load the **zfcp** device driver to see the attached tape devices.

1. Create a **/etc/zfcp.conf** file and make a ram disk to statically attach tape devices into your system. You can use this method only if you have a persistent mapping in a SAN environment. Every time that you restart the system, the **zfcp** is automatically loaded and the tape devices can be seen from the system.

You must add the device map into this file. The following code is an example of **zfcp.conf**.

```
0xf1c0 0x1:0x5005076300402733 0x0:0x0000000000000000;\
0xf1c1 0x1:0x5005076300402733 0x0:0x0001000000000000
```

The **zfcp** device driver uses the "**map**" module parameter to recognize a physically attached tape device. **map** takes the following format,

```
map="<devno><port scsi-id>:<wwpn><unit-scsi-lun>:<fcp-lun>;..."
```

Where

### **devno**

The device number of the host bus adapter (16 bits, see `/proc/subchannels`). It is 0xf1c0 or 0xf1c1 in the previous example.

### **port scsi-id**

Linux internal SCSI ID assigned to the Fibre Channel port of the SCSI target device (32-bit, must not be 0, must be a unique one-to-one mapping for each worldwide port name. It is 0x1 in the previous example.

### **wwpn**

Worldwide port name that identifies the Fibre Channel port of the SCSI target device (64-bit). It is 0x5005076300402733 in the previous example.

### **unit scsi-lun**

Linux internal SCSI Logical Unit Number (32-bit). It is 0x0 in the previous example.

### **fcp-lun**

Logical Unit Number that is associated with the SCSI target device (64-bit). In the previous example, 0x0000000000000000 is the Logical Unit Number 0, and 0x0001000000000000 is the Logical Unit Number 1.

For tape attachment, each logical unit number must be associated with a unique devno. If you use the same devno numbers for several logical units, you can ensure that each `<unit-scsi-lun>` is unique. After **/etc/zfcp.conf** is created, run the following commands.

```
>mk_initrd>zipl
```

Then restart the system. After it is booted up, your tape device must be shown in **/proc/scsi/scsi** file.

2. Modify the **/etc/modules.conf** file to add the **zfcp** module parameters; then run the **depmod -A** and **modprobe zfcp** command.



**Note:** Do not use this choice together with the first one, otherwise it causes conflicts.

The **zfc** map in **/etc/modules.conf** always takes higher priority than the map in **/etc/zfc.conf**.

The following example demonstrates the **zfc** configuration in **/etc/modules.conf**.

```
options zfc map="\
0xf1c0 0x1:0x5005076300402733 0x0:0x0000000000000000;\
0xf1c1 0x1:0x5005076300402733 0x0:0x0001000000000000"
```

The map arguments are the same as the ones listed in for the **/etc/zfc.conf** file.

After the **/etc/modules.conf** file is modified, save and close it. Then, run the following command.

```
>depmod -A
>modprobe zfc
```

This action installs the **zfc** device driver and all of its prerequisite kernel modules. Now you can check the file **/proc/scsi/scsi** to see whether all of the attached tape devices are shown in this file. If not, then check the Fibre Channel connection, such as the fibre cables, or if the devices are powered on.

Then, run the following commands to install **zfc**.

```
>rmmod zfc
>modprobe zfc
```

3. Run the **modprobe zfc** command first, then dynamically add a tape device into the system after you physically attach a Fibre Channel tape device to the switch.

If you physically attach a tape device on the switch and **zfc** is already loaded, you do not need to restart the Linux system to add this entry in the **/proc/scsi/scsi** file. The **zfc** device driver provides an "add\_map" proc system entry under the directory **/proc/scsi/zfc** to dynamically add the device into the system. For example, to add two logical units from the example in Step 2 into the system, you can issue the following commands.

```
> echo "0xf1c0 0x1:0x5005076300402733 0x0:0x0000000000000000;\
0xf1c1 0x1:0x5005076300402733 0x0:0x0001000000000000"
> /proc/scsi/zfc/add_map
> echo "scsi add-single-device 0 0 1 0" > /proc/scsi/scsi
> echo "scsi add-single-device 1 0 1 1" > /proc/scsi/scsi
```

The **scsi add-single-device** takes four parameters, corresponding to the four parameters **scsi**, **Channel**, **Id**, and **Lun** in the **/proc/scsi/scsi** file. The value of **scsi** is 0 for the first devno, 1 for the second devno (if it is different from the first devno), and so on. The value of **Channel** can start from 0 for each different **scsi** value. The value of **Id** is the one you use for <unit scsi-lun> in the previous mapping. The value of **Lun** is the logical unit number of the target device, for example, the last number in the previous mapping. Currently, the **zfc** device driver does not support dynamically removing the attached devices. If you must remove the tape devices from the system, do **rmmod zfc**. Then, you can delete the entry in **/etc/modules.conf** and reload **zfc**, or reload **zfc** first and dynamically add the devices that you want. After you finished all the mapping, if you can see all of the attached tape devices in **/proc/scsi/scsi**, you successfully attached those devices to your system. Next, you can install the **lin\_tape** device driver. Refer to the [“Installation procedure”](#) on page 42 section for the instructions on how to install **lin\_tape**.

## Uninstallation procedure

**Note:** All tape devices that use the **lin\_tape** driver must be closed and cannot be in use when **lin\_tape** is uninstalled or the uninstall fails.

On SLES and RHEL:

Run the following command to uninstall **lin\_taped** before you uninstall **lin\_tape**:

```
> rpm -e lin_taped
```

Run the following command to uninstall **lin\_tape**:

```
>rpm -e lin_tape
```

On Ubuntu:

```
apt purge lin-taped  
apt purge lin-tape
```

## Tape drive, media, and device driver parameters

This chapter describes the parameters that control the operating modes of the IBM Linux Tape and Medium Changer device driver.

### Configuration parameters

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. The installation defaults are provided for all parameters initially. These parameters are kept on reopen, but are always restored back to the default values when the **lin\_tape** device driver is reinstalled.

**Note:** This procedure is completed with tape diagnostic and utility functions. See [“IBM Tape Diagnostic Tool \(ITDT\)”](#) on page 76.

The nonchangeable configuration parameters are

- Autoloading
- Density code
- Emulate autoloader
- Hook word
- Maximum block size
- Minimum block size
- Medium type
- Read SILI bit
- Record space mode
- Volume ID for logging
- Write protect

The changeable configuration parameters are

- Barcode length
- Block size
- Buffered mode
- Capacity scaling
- Compression
- Disable auto drive dump
- Disable SIM logging
- Dynamic attributes
- Logging
- Logical write protect
- Maximum SCSI transfer length
- Read past filemark
- Rewind immediate

- Trace
- Trailer labels
- Busy Retry

## Nonchangeable parameters

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. The nonchangeable parameters are detailed as follows.

### Autoloading

This parameter enables the autoloading feature of the device driver. It is disabled by default and cannot be changed.

### Capacity scaling

This parameter sets the capacity or logical length of the current tape. By reducing the capacity of the tape, the tape drive can access data faster at the expense of data capacity. Capacity scaling is not supported currently but might be supported in future releases of **lin\_tape**.

### Density code

This parameter is the density setting for the currently loaded tape. Some tape devices support multiple densities and report the current setting in this field. It cannot be changed by the application.

### Emulate autoloader

This parameter currently is not supported and is ignored.

### Hook word

This parameter is not supported in the **lin\_tape** device driver.

### Logical write protect

This parameter sets or resets the logical write protect of the current tape. This feature is not supported currently but might be supported in future releases of the **lin\_tape**.

### Maximum block size

This parameter is the maximum block size for the device.

### Minimum block size

This parameter is the minimum block size for the device.

### Medium type

This parameter is the media type of the current loaded tape. Some tape devices support multiple media types and different values are reported in this field.

### Read SILI bit

SILI bit currently is not supported due to limitations associated with the Linux environment. SILI bit support can be enabled in future releases of the **lin\_tape**.

## Record space mode

This parameter specifies how the device driver operates when a forward or backward space record operation encounters a filemark. Only the SCSI mode is supported by **lin\_tape**. When a forward or backward space record operation is issued to the driver and a filemark is encountered, the device driver returns **-1** and the *errno* variable is set to input/output error (EIO). On the forward space operation, the tape is left-positioned after the filemark (the end of tape side of the filemark). On the backward space operation, the tape is positioned before the filemark (the beginning of tape side of the filemark).

## Volume ID for logging

This parameter is the volume ID of the currently loaded tape. The **lin\_tape** device driver ignores this field.

## Write protect

This parameter is set to TRUE if the currently mounted tape is logically or physically write protected.

## Changeable parameters

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. The changeable parameters are detailed as follows.

### Barcode length

This parameter can be set to change the barcode length for a cartridge. For LTO cartridges the default is 8. It can be changed to 6 for LTO 1 and LTO 2 generation cartridges only. For 3592 cartridges the default is set at 6. It can be changed to 8. In the **/etc/modprobe.conf.local** file the following line must be added to reflect the desired change.

```
options lin_tape ibm3592_barcode=8
options lin_tape lto_barcode=6
```

### Block size

This parameter specifies the block size that is used for read and write operations. A value of zero means a variable block size. Any other value is a fixed block size. The installation default is zero (variable length block size). Refer to [“Maximum SCSI transfer length”](#) on page 50 for guidance.

### Buffered mode

This parameter specifies whether read and write operations must be buffered by the tape device. The default (recommended) value is TRUE.

### Capacity scaling

This parameter sets the capacity or logical length of the current tape on Enterprise Tape System 3590 or 3592 tape subsystems. By reducing the capacity of the tape, the tape drive can access data faster at the expense of data capacity. Capacity scaling can be set at 100% for the entire tape (which is the default), or set at 75%, 50%, or 25% of the 3590 tape cartridge and more available capacity scaling for the 3592 standard 300 GB rewritable data cartridge. Capacity scaling remains with the tape across mounts until it is changed.

#### Note:

1. The tape position must be at the start of the tape to change this parameter from its current value.
2. Changing this parameter destroys any existing data on the tape.
3. For 3592 media types, capacity scaling is supported only for the standard 300 GB rewritable data cartridge. Attempting to set capacity scaling that is not supported by a device or the current loaded

media always returns 100% and cannot be changed. For example, the 60 GB (Economy Data) cartridge for the IBM 3592 cannot be capacity-scaled and is always 100%.

## Compression

Hardware compression is implemented in the device hardware. This parameter turns the hardware compression feature On and Off. If compression is enabled, the effective performance can increase, based on the compressibility of the data.

The installation default is On (use compression).

## Disable auto drive dump

This parameter is provided in the **lin\_tape** version 1.2.2 or later. It is set to FALSE by default. If it is FALSE and the **lin\_taped** daemon is running and if an error occurs in the drive that creates a drive dump, the **lin\_tape** device driver automatically retrieves the drive dump and saves it under the `/var/log` directory by default. You can specify another directory in the `/etc/lin_taped.conf` file. The dump is labeled with a .dmp extension on the file. Refer to [“Configuring and running the lin\\_taped daemon” on page 62](#) for details.

## Disable SIM logging

This parameter is provided in the **lin\_tape** version 1.2.2 or later. It is set to FALSE by default. If it is FALSE and the **lin\_taped** daemon is running and SIM/MIM data is generated by the drive, the **lin\_tape** device driver automatically retrieves the data and saves it in a formatted text file under the `/var/log` directory by default. You can specify another directory in the `/etc/lin_taped.conf` file. Refer to [“Configuring and running the lin\\_taped daemon” on page 62](#) for details.

This capacity is not applicable to IBM Ultrium tape drives.

## Dynamic attributes

This parameter determines whether dynamic runtime attributes are attempted on open for supported drives. Default is 1 (on) meaning that the driver automatically attempts to set dynamic runtime attributes on open. This parameter can be changed to 0 (off) in the configuration file before the **lin\_tape** is loaded. It is recommended to keep on dynamic attributes unless it produces an unexpected problem in the environment.

## Logging (volume logging)

This parameter turns the volume information logging On or Off. With the **lin\_tape** version 1.2.2 and later, the **lin\_tape** device driver provides this support. It is set to On by default. If logging is On and the **lin\_taped** daemon is running, the **lin\_tape** device driver retrieves the full log sense data from the drive whenever a tape is unloaded, or the drive reaches a log threshold. The log file is saved in binary format under the directory `/var/log` by default. You can specify another directory in `/etc/lin_taped.conf` file. Refer to [“Configuring and running the lin\\_taped daemon” on page 62](#) for details.

**Note:** This parameter is volume logging, which is different from error logging. **lin\_tape** provides error logging whenever the **lin\_taped** daemon is running. Refer to [“Configuring and running the lin\\_taped daemon” on page 62](#) for details on error logging.

## Logical write protect

This parameter sets or resets the logical write protect of the current tape on Enterprise Tape System 3590 tape subsystems. The three types of logical write protect are associated protect, persistent protect, and write-once read-many (WORM) protect.

1. Associated protect remains only while the current tape is mounted or associated with the tape drive. It is reset when the tape is unloaded or the tape drive is reset.

2. Persistent protect remains or persists with the tape across mounts until it is reset.
3. WORM protect also remains with the tape across mounts, but unlike persistent protect it cannot be reset on the tape. After a tape is WORM protected, it can never be written on again.

**Note:** The tape position must be at the start of the tape to change this parameter from its current value.

## Maximum SCSI transfer length

In the **lin\_tape** drivers with level lower than 3.0.3, the maximum transfer length per device per SCSI command is 262144 bytes (256 KB) by default. Variable block read/write requests with transfer length greater than the maximum transfer length fails [*errno*: EINVAL]. When a fixed block size is defined, large write requests are subject to both the granularity of the block size and the maximum transfer length. For example, with a fixed block size of 80000 bytes and maximum transfer length of 262144, a write request for 400000 bytes (5 blocks of 80000 each) is written to tape in two transfers. The first transfer is 240000 bytes (3 blocks) and the second transfer is 160000 (the remaining two blocks). You can increase the maximum transfer length to enhance the data throughput. This procedure can be done with ITDT with the Query/Set Parameters option, or a customized STIOCSETP input/output control (*ioctl*) call. However, setting the transfer length greater than the default 256 KB does not guarantee a noticeable increase in data throughput. Maximum transfer length of 256 KB is highly recommended.

In **lin\_tape** driver with level 3.0.5 or higher and the open source driver **lin\_tape**, the maximum transfer length is defined as the minimum length that the host bus adapter and the tape drive can support. This number is greater than 256 KB. It cannot be changed by the STIOCSETP *ioctl* call any more.

## Read past filemark

If this parameter is set to true, when a fixed-length read operation encounters a filemark, it returns the number of bytes read before the filemark is encountered and positions the tape head after the filemark. If the **read\_past\_filemark** parameter is set to false, when the fixed-length read operation encounters a filemark, if data was read, the *read* function returns the number of bytes read, and positions the tape head before the filemark. If no data was read, then the *read* returns 0 bytes read and positions the tape head after the filemark.

This installation default is FALSE.

## Rewind immediate

This parameter sets the immediate bit for rewind commands. If it is set to On, the rewind tape operation runs faster, but the next command takes a long time to finish unless the physical rewind operation is complete. Setting this parameter reduces the amount of time it takes to close a device for a Rewind on Close special file.

The installation default is Off (no rewind immediate).

## Trace

This parameter turns the trace facility On or Off. With the **lin\_tape** version 1.2.2 and later, the **lin\_tape** device driver provides this support. It is set to On by default. If trace is On and the **lin\_taped** daemon is running, the **lin\_tape** device driver retrieves the trace from the driver if trace level is set to 1 or 2 in the **/etc/lin\_taped.conf** file. The trace file is saved under the directory **/var/log** by default. You can specify another directory in **/etc/lin\_taped.conf** file. Refer to [“Configuring and running the lin\\_taped daemon”](#) on page 62 for details.

## Trailer labels

If this parameter is set to On, then writing records past the early warning mark on the tape is allowed. The first write operation after detecting the early warning mark fails and the *errno* variable is set to ENOSPC. No data is written during the operation. All subsequent write operations are allowed to continue until the physical end of the volume is reached and *errno* EIO is returned.

If this parameter is set to Off, then writing records past the early warning mark is not allowed. *Errno* variable is set to ENOSPC.

The installation default is On (with trailer labels).

## Busy Retry

The parameter **busy\_retry** determines how many times to retry a command when the device is busy. Default is 0 (off), and can be set up to 480 in the configuration file before the lin\_tape is loaded.

## lin\_tape\_ignoreOEM

The parameter lin\_tape\_ignoreOEM stops OEM devices from connecting to the driver. Default is 0 (off), and can be set up to 1 in the configuration file before the lin\_tape is loaded.

## Special files

After the driver is installed and a device is configured and made available for use, access is provided through the special files. These special files, which consist of the standard Linux special files for devices, are in the **/dev** directory.

### Special files for the tape device

Each tape device has a set of special files providing access to the same physical drive but providing different attributes. [Table 7 on page 51](#) shows the attributes of the special files.

**Note:** The asterisk (\*) in **IBMtape\*** represents a number assigned to a particular device, such as **IBMtape0**.

For tape drives with attached medium changer devices, the **IBMchanger\*** special file provides a separate path for issuing commands to the medium changer. When this special file is opened, the application can view the medium changer as a separate device. Both the tape and changer special file can be opened at the same time.

Table 7. Linux: Special files for IBM tape devices	
Special file name	Rewind on Close
/dev/IBMTape*	YES
/dev/IBMTape*n	NO

### Special files for the Medium Changer device

After the driver is installed and a medium changer device is configured and made available for use, access to the robotic device is provided through the **IBMchanger** special file in the **/dev** directory. The asterisk (\*) represents a number that is assigned to a particular device, such as **IBMchanger0**. The term *IBMchanger* is used for a SCSI medium changer device. The **IBMchanger\*** special file provides a path for issuing commands to control the medium changer robotic device.

The file descriptor that results from opening the **IBMchanger** special file does not support the following operations.

- Read
- Write
- Open in Append mode
- Commands that are designed for a tape device

## Persistent naming support

**Lin\_tape** Persistent Naming is implemented through the Linux **udev** utility. **Udev** is a service that monitors changes in system hardware configuration and completes actions that are based on what devices are attached to the Linux system. It can be configured to create symbolic links (persistent names) to a device based on attributes that a driver exports for that device. The persistent name can then be used as the device name to open and complete IO to a tape drive or medium changer. This action makes it possible to reference a static name, such as

```
/dev/lin_tape/by-id/lin_tape4801101
```

This name is always associated with the same physical device, rather than being required to reference the device name **/dev/IBMtape0**, which can change names and become **/dev/IBMtape1** after the driver is reinstalled.

**Lin\_tape** exports several attributes that can be used as the basis to create persistent names. These attributes can be reported to the user through **udevadm info** on recent Linux kernels, or **udevinfo** on older Linux kernels. The **udevinfo** and **udevadm** are **udev** management tools. These tools can query the **udev** database for device information and properties that are stored in the **udev** base for help creating **udev** rules.

**Note:** **Udev**, **udevinfo**, and **udevadm** are not implemented or maintained by the **lin\_tape** driver. Refer to the man pages for **udevadm** or **udevinfo** for details on usage. These man pages are system specific and supersede all information in this document. For questions on these utilities, you must contact your Linux support representative.

An example is provided on **udev** for implementing a persistent name. The example must be customized to fit a user's needs and environment.

**Note:** Variations exist between kernels.

If a tape device is attached to the Linux system with worldwide port name 0x123456789ABCDEF0 with a current device name of **/dev/IBMtape0**, a user can run **udevadm** information to obtain information on exported attributes for this device. This procedure can be done as follows,

```
>udevadm info --attribute-walk --name /dev/IBMtape0
```

The output of this command includes something similar to

```
ATTRS{serial_num}=="123456789"  
ATTRS{ww_node_name}=="0x123456789ABCDEF1"  
ATTRS{ww_port_name}=="0x123456789ABCDEF0"
```

If you are using **udevinfo**, enter the previous command as

```
>udevinfo -a -p `udevinfo -q path -n /dev/IBMtape0`  
or  
>udevinfo -a -p /sys/class/lin_tape/IBMtape0  
(Both commands return the same information).
```

Also, on some kernels an attribute **ATTRS{xxx}** is replaced by **SYSFS{xxx}**. Furthermore, some kernels use a '=' (single equal sign) to indicate an attribute match and also an assignment. Whereas, other kernels use a '==' (double equal sign) for a match and '=' for assignment. Place the attribute from the attribute list into your rules file **exactly as it appears** in the attribute list, as described here.

The **ww\_port\_name** is in a rules file that assigns a symbolic link to a device that has the listed worldwide port name. The file typically is placed in **/etc/udev/rules.d**, but this location might be changed by the **udev\_rules** directive in the **/etc/udev/rules.conf** file. In this example, a file is created that is called **/etc/udev/rules.d/98-lin\_tape.rules** and write a single line to the file.

```
KERNEL=="IBMtape*", ATTRS{ww_port_name}=="0x123456789ABCDEF0",
```

```
SYMLINK="lin_tape/by-id/lin_tape4801101"
```



Assuming that the **udev** service is running and configured correctly, the user can install or reinstall **lin\_tape** with **modprobe**, and the symbolic link is created in the **/dev/lin\_tape/by-id** folder. One line must be added to the **98-lin\_tape.rules** file for each wanted symbolic link.

With **lin\_tape** version 2.2.0, a parameter, **persistent\_n\_device**, is added to support persistent naming of no rewind devices. The default value is 0 (off). To enable this support, complete the following steps.

1. Modify the **98-lin\_tape.rules** file to differentiate standard devices and no rewind devices. For example,

```
KERNEL=="IBMtape*[0-9]", ATTR{serial_num}=="1013000306", SYMLINK="lin_tape/by-id/IBMtape0"  
KERNEL=="IBMtape*n", ATTR{serial_num}=="1013000306", SYMLINK="lin_tape/by-id/IBMtape0n"
```

2. Stop the **lin\_taped** daemon.

```
lin_taped stop
```

3. Unload the **lin\_tape** driver from the memory.

```
modprobe -r lin_tape  
  
Only if sfmp was enabled unload stmp/sgmp:  
modprobe -r stmp  
modprobe -r sgmp  
  
modprobe -r pfo
```

4. Add the following line in your **/etc/modprobe.conf** or **/etc/modprobe.conf.local** file (or, if you are running RHEL 6 or higher, in your **/etc/modprobe.d/lin\_tape.conf** file).

```
options lin_tape persistent_n_device=1
```

5. Reload the **lin\_tape** driver into memory.

```
modprobe pfo  
modprobe lin_tape  
  
Only if sfmp was enabled load stmp/sgmp:  
modprobe stmp  
modprobe sgmp
```

**Note:** Wait at least 10 seconds between step 3 (**modprobe -r lin\_tape**) and step 5 (**modprobe lin\_tape**) in order for **udev** to correctly configure the devices.

6. Check that the devices are all correctly listed with the following command.

```
ls -l /dev/lin_tape/by-id/
```

7. Restart the **lin\_taped** daemon.

```
lin_taped start
```

## Control Path failover support for tape libraries

**Note:** The library control path failover feature code must be installed before control path failover support is enabled in the Linux **lin\_tape** device driver. Refer to [“Automatic failover”](#) on page 5 to determine which feature code is required for your machine type.

The Linux **lin\_tape** device driver control path failover support configures multiple physical control paths to the same logical library within the device driver. It also provides automatic failover to an alternative control path when a permanent error occurs on one path. This support is transparent to the running application.

### Configuring and unconfiguring path failover support

Control path failover support is not enabled automatically when the device driver is installed. The Linux **lin\_tape** device driver provides a driver parameter **alternate\_pathing** for you to enable the library

control path failover. To enable the failover support in the **lin\_tape** device driver software, you must do the following steps after the **lin\_tape** rpm package is installed:

1. `lin_taped stop` (stop the `lin_taped` daemon)
2. Unload the **lin\_tape** driver from the memory.

```
modprobe -r lin_tape

Only if sfmp was enabled unload stmp/sgmp:
modprobe -r stmp
modprobe -r sgmp

modprobe -r pfo
```

3. Add the following line in your **/etc/modprobe.conf** or **/etc/modprobe.conf.local** file (or, if you are running RHEL 6 or higher, in your **/etc/modprobe.d/lin\_tape.conf** file)

```
options lin_tape alternate_pathing=1
```

4. Reload the **lin\_tape** driver into memory.

```
modprobe pfo
modprobe lin_tape

Only if sfmp was enabled load stmp/sgmp:
modprobe stmp
modprobe sgmp
```

5. `lin_taped start` (restart `lin_taped` daemon)

You can ignore the Unresolved symbols in `/lib/modules/<your kernel name>/drivers/scsi/lin_tape.ko` message after the **depmod** command. You can check whether the **lin\_tape** driver recognized multiple control paths for your library by reading the **/proc/scsi/IBMchanger** file.

- `cat/proc/scsi/IBMchanger`

If your library lists "Primary" or "Alternate" under "FO Path", you successfully enabled the control path failover feature for your library. If "NA" is listed under "FO Path", then the control path failover is not enabled. After control path failover support is enabled, it remains set until the **lin\_tape** driver is reloaded with the **alternate\_pathing** driver parameter set to OFF. The path failover setting is retained even if the system is rebooted. If you want to turn off the control path failover feature in the **lin\_tape** device driver, you can complete the following steps.

1. `lin_taped stop`
2. Unload the **lin\_tape** driver from the memory.

```
modprobe -r lin_tape

Only if sfmp was enabled, unload stmp/sgmp:
modprobe -r stmp
modprobe -r sgmp

modprobe -r pfo
```

3. Delete the following line in your **/etc/modprobe.conf** file.

```
options lin_tape alternate_pathing=1
```

4. Reload the **lin\_tape** driver into memory.

```
modprobe pfo
modprobe lin_tape

Only if sfmp was enabled load stmp/sgmp:
modprobe stmp
modprobe sgmp
```

5. `lin_taped start` (restart `lin_taped` daemon)

## Enabling path failover for the sg interface

Through a collaboration effort that is named **join driver**, control path failover is supported through the sg interface, by using the latest **lin\_tape** driver version on RHEL over Intel only.

**Note:** It is important to review hardware and software requirements before path failover is enabled. See [“Path failover and load balancing”](#) on page 4.

To enable it, the **lin\_tape\_as\_sfmp** parameter must be set at `/etc/modprobe.d/lin_tape.conf` as follows:

```
options lin_tape alternate_pathing=1
options lin_tape lin_tape_as_sfmp=1
```

If **lin\_tape** is already installed, follow the [“Uninstallation procedure”](#) on page 45.

Follow the [“Installation procedure”](#) on page 42 by using

```
-with sfmp at rpmbuild.
```

## Primary and alternative paths

When **lin\_tape** is loaded into kernel memory, the first logical medium changer device that **lin\_tape** sees in the system is the primary path for that medium changer. The other logical medium changers that **lin\_tape** attached for the same medium changer are configured as alternative paths. The device driver supports up to 16 physical paths for a single device. The primary and alternative path information can be obtained by the following command.

- `cat /proc/scsi/IBMchanger`

An example of a `/proc/scsi/IBMchanger` file:

- lin\_tape version: 3.0.3
- lin\_tape major number: 253

*Table 8. Attached changer devices*

Number	Model	SN	HBA	FO Path
0	03584L22	IBM1234567	qla2xxx	Primary
1	03584L22	IBM1234567	qla2xxx	Alternate
2	03584L22	IBM1234567	qla2xxx	Alternate

The labeling of a logical device as either a primary or alternative path is for information only, to

- Identify the actual number of physical devices that are configured on the system and a specific logical device that is associated with them. Only one logical device is labeled as the primary path for each physical device. However, multiple logical devices can be labeled as an alternative path for the same devices.
- Provide information about which logical devices configured on the system have path failover support enabled.

The numbers listed in [Table 8 on page 55](#) are the ones used for IBMchanger special files at `/dev` directory (see [Special Files for the Medium Changer device](#)). An attempt to open a device file name not listed at `/dev` directory will fail. Per file systems handling and due to path removal, usually a device file name is deleted after a device closes or before a device opens. Using [Persistent naming support](#) will maintain device file names listed under `ls -l /dev/lin_tape/by-id/`.

When **lin\_tape\_as\_sfmp** is set, sg paths can be queried through *pfo* paths as follows:

```
cat /sys/class/pfo/*/paths
```

Example output:

```
pfo10 sg=/dev/sg10 st=none sf=yes fo=no wwnn=0000013400140405 type=changer
2:0:1:1 up last - - -
3:0:1:1 - - - -
```

There is only one sg device file name per device that uses all the paths for this device.

## Querying primary and alternative path configuration

You can show the primary and alternative path configuration for all devices by reading the `/proc/scsi/IBMchanger` file, as explained in [“Primary and alternative paths” on page 55](#).

## Disabling and enabling primary and alternative paths

When you load the **lin\_tape** device driver with the **alternate\_pathing** parameter to be ON, by default, all the available paths for a physical device are enabled.

If it is necessary to disable a path and not run path failover (for example, because of maintenance), run commands to disable and then later enable the primary and alternative paths.

The commands to enable and disable primary and alternative paths are tape diagnostic and utility functions.

**Note:** See [“IBM Tape Diagnostic Tool \(ITDT\)” on page 76](#).

## Data Path failover and load balancing support for tape drives

Data path failover support is not enabled automatically when the device driver is installed. The Linux **lin\_tape** device driver provides a driver parameter **alternate\_pathing** for you to enable the data path failover.

To enable the failover support in the **lin\_tape** device driver software, you must complete the following steps after the **lin\_tape rpm** package is installed.

```
>lin_taped stop (stop the lin_taped daemon)
>xmmod lin_tape (unload the lin_tape driver from the memory)
```

If you have IBM 3592 tape drives, add the following line in your `/etc/modprobe.conf` or `/etc/modprobe.conf.local` file (or, if you are running RHEL 6 or higher, in your `/etc/modprobe.d/lin_tape.conf` file).

```
options lin_tape alternate_pathing=1
```

If you have IBM LTO tape drives, the library must have a path failover feature code. The data path failover license keys are needed to enable the failover if you are running LTO2 drives or if you are running LTO3 drives with old levels of drive code. DPF keys do not need to be added if you are running the latest drive code on LTO3 or higher drives.

Add the following line in your `/etc/modprobe.conf` or `/etc/modprobe.conf.local` file (or, if you are running RHEL 6 or higher, in your `/etc/modprobe.d/lin_tape.conf` file).

```
options lin_tape alternate_pathing=1 dpf_keys="abcdefghijklmnop"
```

abckdefghijklmnop is an example of a data path failover feature key. If you have multiple libraries and multiple data path failover feature keys, input your keys as follows.

```
dpf_keys="key1;key2;..."
```

Save the file, then run the following commands.

```
>depmod
>modprobe lin_tape (re-load the lin_tape driver into memory)
>lin_taped (re-start lin_taped daemon)
```

You can ignore the Unresolved symbols in `/lib/modules/<your kernel name>/drivers/scsi/lin_tape.ko` message after the **depmod** command. You can check whether the **lin\_tape** driver recognized multiple paths for your tape drive by reading the **/proc/scsi/IBMtape** file.

```
>cat /proc/scsi/IBMtape
```

If your tape drive lists Primary or Alternate under F0 Path, you successfully enabled data path failover feature for your tape drive. If NA is listed under F0 Path, the data path failover is not enabled. After the path failover support is enabled, it remains set until the **lin\_tape** driver is reloaded with the **alternate\_pathing** driver parameter set to OFF. The path failover setting is retained even if the system is rebooted. If you want to turn off the data path failover feature in the **lin\_tape** device driver, you can run the following steps.

```
>lin_taped stop
modprobe -r lin_tape

Only if sfmp was enabled unload stmp/sgmp:
modprobe -r stmp
modprobe -r sgmp

modprobe -r pfo
```

Delete the following line in your **/etc/modules.conf** file: `options lin_tape alternate_pathing=1`.

```
modprobe pfo
modprobe lin_tape

Only if sfmp was enabled load stmp/sgmp:
modprobe stmp
modprobe sgmp

>lin_taped start
```

## Enabling path failover for *st* and *sg* interface

Through a collaboration effort that is named **join driver**, control path failover is planned to be supported through *st* and *sg* interfaces by using the latest **lin\_tape** driver version on RHEL over Intel only.

**Important:** Review hardware and software requirements before path failover is enabled. See [“Path failover and load balancing”](#) on page 4.

To enable it, the **lin\_tape\_as\_sfmp** parameter must be set at `/etc/modprobe.d/lin_tape.conf`:

```
options lin_tape alternate_pathing=1
options lin_tape lin_tape_as_sfmp=1
```

If **lin\_tape** is already installed, follow the [“Uninstallation procedure”](#) on page 45.

Follow [“Installation procedure”](#) on page 42 by using

```
-with sfmp at rpmbuild.
```

## Primary and alternative paths

When the **lin\_tape** device driver is loaded into kernel memory with path failover support enabled, the first logic device that **lin\_tape** sees always becomes the primary path. The other logical devices that **lin\_tape** sees are configured as the alternative paths. The device driver supports up to 16 physical paths for a single device.

The primary and alternative path information can be obtained by the following command.

```
>cat /proc/scsi/IBMtape
```

The following is an example of a `/proc/scsi/IBMtape`:

```
lin_tape version: 3.0.3
lin_tape major number: 253
```

*Table 9. Attached tape devices*

Number	Model	SN	HBA	FO Path
0	03592	IBM1234567	qla2xxx	Primary
1	03592	IBM1234567	qla2xxx	Alternate

The labeling of a logical device as either a primary or alternative path is for information only and is used for following purpose:

- Identify the actual number of physical devices that are configured on the system and a specific logical device that is associated with them. Only one logical device is labeled the primary path for each physical device. However, many (multiple) logical devices can be labeled as an alternative path for the same device.
- Provide information about which logical devices configured on the system have path failover support enabled.

The numbers listed in [Table 9 on page 58](#) are the ones used for IBMchanger special files at `/dev` directory (see [Special files for the tape device](#)). An attempt to open a device file name not listed at `/dev` directory will fail. Per file systems handling and due to path removal, usually a device file name is deleted after a device closes or before a device opens. Using [Persistent naming support](#) will maintain device file names listed under `ls -l /dev/lin_tape/by-id/`.

When `lin_tape_as_sfmp` is set, `st` and `sg` paths can be queried through `pfo` paths as follows:

```
cat /sys/class/pfo/*/paths
```

Example output:

```
pfo9 sg=/dev/sg9 st=/dev/st0 sf=yes fo=no wwnn=500507630f04be07 type=tape
2:0:1:0 up last - wwpn=500507630f44be07
3:0:1:0 - - - -
```

Only one `st` and one `sg` device file name per device can use all the paths for that device.

## Querying primary and alternative path configuration

You can show the primary and alternative path configuration for all devices by reading the `/proc/scsi/IBMtape` file, as explained in [“Primary and alternative paths” on page 57](#).

## Disabling and enabling primary and alternative paths

If it is necessary to disable a path and not run path failover (for example, because of maintenance), run commands to disable and then enable the primary and alternative paths.

The commands to enable and disable primary and alternative paths are tape diagnostic and utility functions.

**Note:** See [“IBM Tape Diagnostic Tool \(ITDT\)” on page 76](#).

## Tape Reserve Type

This parameter causes `lin_tape` to issue SCSI-3 persistent reserves to a tape drive whenever a reservation is attempted. Persistent reserves are automatically issued if data path failover is used, and

therefore setting the parameter is unnecessary. This parameter can be set only when **lin\_tape** is installed. To set it, add the following line to **/etc/modprobe.conf** or **/etc/modprobe.conf.local** (or, if you are running RHEL 6 or higher, to **/etc/modprobe.d/lin\_tape.conf**):

```
options lin_tape tape_reserve_type=persistent
```

If alternate pathing is not enabled persistent reserve is not issued automatically, so this parameter is needed only if you want to use persistent reserve with **alternate\_pathing** disabled.

## Open source device driver - lin\_tape

The **lin\_tape** device driver is the new device driver for the Linux 2.6 kernels to replace the closed-source driver IBMtape. In most respects, it behaves the same as the closed-source IBMtape device driver. This section covers significant differences between the IBMtape driver and the **lin\_tape** driver.

### Comparing IBMtape and lin\_tape

Table 10 on page 59 compares the names for various components of the **IBMtape** and **lin\_tape** device drivers.

Table 10. Comparing IBMtape and lin_tape		
Component	IBMtape	lin_tape
Driver name	IBMtape	lin_tape
Module name	IBMtape.ko	lin_tape.ko
Special files	/dev/IBMtape0 /dev/IBMchanger0, and so on.	No change
Proc entry	/proc/scsi/IBMtape /proc/scsi/IBMchanger	No change
Daemon name	IBMtaped	lin_taped
Daemon configuration file	/etc/IBMtaped.conf	/etc/lin_taped.conf
Daemon trace files	/var/log/IBMtape.trace /var/log/IBMtape.errorlog	/var/log/lin_tape.trace /var/log/lin_tape.errorlog

Lin\_tape **join driver** installs **pfo.ko**, an extra lin\_tape module that is idle for non-sfmp configurations, and **blacklist-pfo.conf** at the location **/etc/modprobe.d** for pfo module load control.

### Installation

Installation of the **lin\_tape** driver is the same as for the **IBMtape** driver, except that **IBMtape** must be replaced with **lin\_tape** in the installation instructions. Refer to “[Installation and Configuration instructions](#)” on page 40 for details.

The **lin\_tape** driver cannot be installed if the **IBMtape** driver is already installed. If the **IBMtape** driver is installed, first uninstall the **IBMtape** driver, and then install the **lin\_tape** driver. With RHEL4 and SLES10, driver removal also requires a reboot of the server, since the **IBMtape** driver module is "permanent" in these distributions.

### Driver parameters and special device files

The driver parameters are not changed for the **lin\_tape** driver. However, it is important that the module parameters, such as **alternate\_pathing** and **dpf\_keys**, must now be applied to the **lin\_tape** module,

instead of the **IBMtape** module. For example, in the `/etc/modprobe.conf` or `/etc/modprobe.conf.local` file (or, if you are running RHEL 6 or higher, the `/etc/modprobe.d/lin_tape.conf` file), add the following line for LTO library's path failover:

```
options lin_tape alternate_pathing=1 dpf_keys="abcdefghijklmnop"
```

abckdefghijklmnop is an example of a data path failover feature key.

The special device files for the **lin\_tape** driver are the same as for the **IBMtape** driver. Refer to [“Special files for the tape device” on page 51](#) and [“Special files for the Medium Changer device” on page 51](#) for details on special device files.

## Taking devices offline and completing maintenance

Input and output must be quiesced and all driver handles must be closed before a **lin\_tape** device is taken offline. It is recommended to remove the **lin\_tape** driver module and then the HBA driver module before maintenance is done or the physical topology of the tape drive or library environment is changed. **Lin\_tape** can be removed by the following command at the shell prompt.

```
lin_taped stop
modprobe -r lin_tape

Only if sfmp was enabled unload stmp/sgmp:
modprobe -r stmp
modprobe -r sgmp

modprobe -r pfo
```

Likewise, the HBA module can be removed by the following command, where "HBA\_driver" is your specific HBA driver.

```
rmmod <HBA_driver>
```

After the maintenance is complete and the environment is ready to issue input and output, the drivers must be reinstalled in reverse order. This procedure is typically done by

```
modprobe <HBA_driver>
modprobe pfo
modprobe lin_tape

Only if sfmp was enabled load stmp/sgmp:
modprobe stmp
modprobe sgmp

>lin_taped start
```

## Path failover support

Path failover support in **lin\_tape** is the same. However, with the **lin\_tape** driver, failover support is provided through the `lin_taped` daemon. If the `lin_taped` daemon is not running, neither control path failover nor data path failover is attempted. The `lin_taped` daemon is started automatically when the **lin\_tape** driver is loaded.

To check whether the `lin_taped` daemon is running, run the following command.

```
lin_taped status
```

This command indicates whether the `lin_taped` daemon is running. If the `/proc/scsi/IBMtape` and `/proc/scsi/IBMchanger` files indicate "NA" for "FO Path", this answer indicates that failover support for that device is not enabled. If all other settings are correct, but "FO Path" is incorrectly indicating "NA", confirm that the `lin_taped` daemon is running.

For details about path failover support, refer to [“Control Path failover support for tape libraries” on page 53](#) and [“Data Path failover and load balancing support for tape drives” on page 56](#).



## lin\_taped daemon

The `lin_taped` daemon uses the same command-line arguments as the `IBMtaped` daemon. The `lin_taped` configuration file is the same as the `IBMtaped` configuration file, but is renamed to **`lin_taped.conf`**. Refer to “Configuring and running the `lin_taped` daemon” on page 62 for information.

## System-managed encryption

### Configuring device drivers

**Note:** System-managed encryption (SME) on Linux requires that the `lin_taped` daemon is running.

The device driver SME settings can be set for all drives at once with the **`default_sys_encryption_proxy`** and **`default_sys_encryption_write`** module options.

If no options are specified in the registry, the driver uses the default values for the parameters.

- The default value for **`default_sys_encryption_proxy`** is 1.

This value causes the device driver to handle encryption key requests, if the drive is set up for system-managed encryption. This value does not need to be changed. A value of 0 causes the device driver to ignore encryption key requests for system-managed encryption drives, and is not desirable.

- The default value for **`default_sys_encryption_write`** is 2.

This value causes the device driver to leave the encryption write-from-BOP settings alone. It does not turn on or turn off encryption writing, but instead uses the settings that are already in the drive. If encryption is not set up previously, then the drive writes unencrypted data. A value of 0 causes the device driver to write unencrypted data. A value of 1 causes the device driver to write encrypted data.

The module options can be specified in the **`/etc/modprobe.conf`**, **`/etc/modprobe.conf.local`**, or **`/etc/modprobe.d/lin_tape.conf`** files, the same as other **`lin_tape`** module parameters.

For example, to turn on SME to write/read encrypted data,

1. Add the following line:

```
options lin_tape default_sys_encryption_write=1
```

2. Then, run the following commands.

```
>lin_taped stop
>modprobe -r lin_tape

Only if sfmp was enabled unload stmp/sgmp:
>modprobe -r stmp
>modprobe -r sgmp

>modprobe -r pfo
Wait at least 5 seconds
>modprobe pfo
>modprobe lin_tape

Only if sfmp was enabled load stmp/sgmp:
>modprobe stmp
>modprobe sgmp

>lin_taped start
```

The default settings are used to initialize the settings for all connected drives.

To modify the settings for individual drives, the settings are **`sys_encryption_write`** and **`sys_encryption_proxy`**. They have the same definitions and values as the similarly named "default" parameters, except that the settings apply only to individual drives.

These settings are available as part of the sysfs infrastructure. For each drive, there are two files, named **`sys_encryption_write`** and **`sys_encryption_proxy`**, in the `/sys/class/lin_tape/{DEVICE}/` directory, where: `{DEVICE}` is the device name, such as `IBMtape0`. The contents of these files indicate

the current setting for the parameter for that particular drive. The setting can be changed by writing a different value for the parameter to the file.

For example, to change the **sys\_encryption\_write** setting for IBMtape0 to ON (which has a value of 1), enter the following at a command line.

```
echo 1">/sys/class/lin_tape/IBMtape0/sys_encryption_write
```

**Note:** The driver encryption parameters for individual drives are not persistent between loads of the **lin\_tape** driver. If you remove the **lin\_tape** driver, and then reload it, the individual settings for all drives are the same as the "default" settings in **/etc/modprobe.conf**, **/etc/modprobe.conf.local**, or **/etc/modprobe.d/lin\_tape.conf**.

The **ibmekm.conf** file, which contains the configuration that the EKM servers use, is installed in the **/etc/** directory when **lin\_tape** is installed. Instructions for modifying this file are found within the file itself.

## Querying tape drive configuration

The following is an example of the output when the drive is configured for system-managed encryption, with encryption turned on.

```
issuing query encryption status...
encryption capable.....Yes
encryption method.....METHOD_SYSTEM
encryption state.....ON
```

## Problem determination

A set of tools is provided with the device driver to determine whether the device driver and the tape device are functioning correctly.

## Tracing driver modules

By default, the driver prints minimal kernel trace messages to the system log at **/var/log/messages**. The minimal information includes notification that a device is recognized or taken offline and also the most serious of error conditions. If a more verbose trace is wanted, the variable **/sys/bus/scsi/drivers/lin\_tape/lin\_tape\_debug** must contain the value 1. This procedure can be accomplished in one of two ways -

- Add the following line to **/etc/modprobe.conf**, **/etc/modprobe.conf.local**, or **/etc/modprobe.d/lin\_tape.conf**.

```
options lin_tape lin_tape_debug=1
```

Then reinstall **lin\_tape**. This action causes the **lin\_tape\_debug** variable to be set every time **lin\_tape** is loaded.

- Issue the following command from the shell.

```
echo 1 > /sys/bus/scsi/drivers/lin_tape/lin_tape_debug
```

This action causes the **lin\_tape\_debug** variable to be set only until **lin\_tape** is uninstalled or until the variable is set back to 0.

## Configuring and running the lin\_taped daemon

Starting with **lin\_tape** version 1.2.5, the **lin\_tape** device driver provides an error diagnostic daemon (**lin\_taped**) which provides the following capabilities:

1. Error logging and tracing

2. When drive dumps, log sense data, or SIM/MIM error information is created by the tape drive, the daemon automatically retrieves that data and saves it to the hard disk drive on your Linux system.
3. Failover and load balancing
4. Encryption

Because `lin_taped` requires a minimal amount of system resource and because it provides these necessary diagnostic capabilities, IBM recommends that you leave the daemon always enabled.

## Installing `lin_taped`

`lin_taped` is automatically installed at `/usr/bin/lin_taped` when you install the **lin\_tape** device driver with the **rpm** or **tar** package. Refer to “Installation and Configuration instructions” on page 40 for instructions on installing the **lin\_tape** device driver.

## Configuring `lin_taped`

You can customize the operation of `lin_taped` by modifying its configuration file, which is at **/etc/lin\_taped.conf**. The daemon reads only the configuration file when it starts; so if you modify the configuration file, stop the daemon, and restart it so that your modifications are recognized by the daemon.

### Tracing:

Three levels of tracing are supported for the `lin_taped` daemon. `lin_taped` tracing is a complement to, but is different from, tracing of the kernel module that is described in “Tracing driver modules” on page 62. The `lin_taped` tracing levels are defined as follows:

- 0** With tracing set to 0, `lin_taped` records minimal tracing.
- 1** With tracing set to 1, `lin_taped` records information that is associated with each `ioctl` called. If a device error occurs and SCSI sense data is obtained from the device, a subset of that sense data is also recorded. The default setting for tracing.
- 2** With tracing set to 2, `lin_taped` records tracing messages for each SCSI command. If a device error occurs and SCSI sense data is obtained from the device, all sense data is also recorded. This tracing level is used only when a specific problem is being diagnosed due to the potential for huge amounts of data that is generated.

Set the `lin_tapeTrace` variable in the **/etc/lin\_taped.conf** file to 0, 1, or 2, depending on what level of tracing you want. If the `lin_tapeTrace` variable is set to an invalid number, the `lin_taped` daemon does not start.

Tracing information is written to a file named **/var/log/lin\_tape.trace**, by default. Information is written into the file until it is 1 MB in size, by default. After 1 MB of information is written, the file is archived (using the Linux **ar** command) into file **lin\_tape.a** in the same directory. In the archive, the file name is renamed to **lin\_tape.trace.timestamp**, where *timestamp* reflects the time that the file was archived.

You can change the directory to which the tracing information is written or the default maximum size of the trace file by modifying settings in the **lin\_taped.conf** file. Refer to the instructions in the **lin\_taped.conf** file for details.

### Error logging:

`lin_taped` records certain error messages from the **lin\_tape** device driver in a file named **/var/log/lin\_tape.errorlog**, by default. Information is written into the file until it is 1 MB in size, by default. After 1 MB of trace information is written, the file is archived (with the Linux **ar** command) into file **lin\_tape.a** in the same directory. In the archive, the file name is renamed to **lin\_tape.errorlog.timestamp**, where *timestamp* reflects the time that the file was archived.

You can change the directory to which the error logging information is written or the default maximum size of the error log file by modifying settings in the **lin\_taped.conf** file. Refer to the instructions in the **lin\_taped.conf** file for details.

Whenever the `lin_taped` daemon is running, error logging is enabled if tracing is enabled. Following is an example an error log record.

```
IBMtape0---E0001 Tue Sep 10 14:04:57 2002
Scsi Path : 03 00 00 00
CDB Command : 01 00 00 00 00 00
Status Code : 08 00 00 01
Sense Data : 70 00 04 00 00 00 00 58 00 00 00 00 00 00 00 FF 0B
              C4 77 00 00 00 06 01 40 00 00 00 00 00 00 00 01 00
              10 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00
              00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Description : Hardware Error
```

The first line indicates the tape device special file name and the device serial number, and the timestamp when the error message was recorded. "Scsi Path" is the SCSI path for this logical unit. It matches the order of the `scsi/Channel/Id/Lun` information in the `/proc/scsi/scsi` file. "CDB Command" is the command data block of the SCSI command. "Status Code" is the returned result from the Linux SCSI middle layer device driver (`scsi_mod.o`). The 4 bytes represent `driver_byte`, `host_byte`, `msg_byte`, and `status_byte`. "Sense Data" is the full SCSI sense data that is returned from the target. "Description" is a person-readable text string that is obtained by parsing the sense key field of the sense data.

The following circumstances are not logged in the `lin_tape.errorlog` file:

1. Sense key is 0, and the sense data indicates an overlength or an underlength read, or encountering a file mark or the end of data
2. Sense key is 2, and the ASC/ASCQ indicates that the device is becoming ready
3. Sense key is 6, indicating a unit attention
4. Sense key is 8, and the ASC/ASCQ indicates the end of data

### Volume logging:

The `lin_tape` device driver retrieves the full log sense data from the tape drive whenever the drive reaches a log threshold, or a tape is unloaded from the drive, or the drive is reset through an application. This data is stored in binary in a file named `lin_tape.timestamp.log`, where: `lin_tape` is the device special file (for example, `lin_tape1`, `lin_tape2`) and `timestamp` reflects the time that the file was created. Each time log sense data is obtained, it is written to a new file. Use the appropriate tape drive hardware reference manual to decode the log sense data.

The volume logging data is stored in the `/var/log` directory by default. You can specify another directory in the `/etc/lin_taped.conf` file.

There are two configuration parameters in the `/etc/lin_taped.conf` file that you can tailor to affect the number of log sense files that are kept on your system.

- `lin_tapeMaxLogSenseFiles`, which has a value of 0 or a positive decimal number.
- `lin_tapeAutoLogSenseFileOverWrite`, which has a value of 0 or 1.

By default, `lin_tapeMaxLogSenseFiles` is 0 and `lin_tapeAutoLogSenseFileOverWrite` is 1, which means that every time log sense data is created, it is written to a new file.

If `lin_tapeMaxLogSenseFiles` is 0, `lin_tapeAutoLogSenseFileOverWrite` is ignored, and each time log sense data is obtained, it is written to a new file.

If `lin_tapeMaxLogSenseFiles` is a positive number and `lin_tapeAutoLogSenseFileOverWrite` is 0, each time log sense data is created, `lin_taped` writes that data to a file until `lin_tapeMaxLogSenseFiles` is created. Then, `lin_taped` stops creating new files, even if new log sense data is produced.

If `lin_tapeMaxLogSenseFiles` is a positive number and `lin_tapeAutoLogSenseFileOverWrite` is 1, each time log sense data is created, `lin_taped` writes that data to a file until `lin_tapeMaxLogSenseFiles` is created. Then, when new log sense data is detected, `lin_taped` deletes the oldest log sense file and creates a new file with the newest log sense data. Thus, only the newest data is kept.

### Automatically retrieving a drive dump:

If a condition occurs in the drive such that a drive dump is created, `lin_taped` retrieves the drive dump and saves it in a file named `lin_tapex.timestamp.dmp`, where `lin_tape` is the device special file (for example,

lin\_tape1, lin\_tape2) and **timestamp** reflects the time that the file was created. Each time a drive dump is obtained, it is written to a new file. The IBM service organization might request that you forward drive dumps to them for analysis.

The drive dumps are stored in the **/var/log** directory by default. You can specify another directory in the **/etc/lin\_taped.conf** file.

There are two configuration parameters in the **/etc/lin\_taped.conf** file that you can tailor to affect the number of drive dumps that are kept on your system.

- **lin\_tapeMaxDumpFiles**, which can have a value of 0 or a positive decimal number.
- **lin\_tapeAutoDriveDumpFileOverWrite**, which can have a value of 0 or 1.

By default, **lin\_tapeMaxDumpFiles** is 0 and **lin\_tapeAutoDriveDumpFileOverWrite** is 1, which means that every time a drive dump is obtained, it is written to a new file.

If **lin\_tapeMaxDumpFiles** is 0, **lin\_tapeAutoDriveDumpFileOverWrite** is ignored, and each time a drive dump is obtained, it is written to a new file.

If **lin\_tapeMaxDumpFiles** is a positive number and **lin\_tapeAutoDriveDumpFileOverWrite** is 0, each time a dump is obtained, lin\_taped writes that data to a file until **lin\_tapeMaxDumpFiles** is created. Then, lin\_taped stops creating new files, even if new drive dumps are produced.

If **lin\_tapeMaxDumpFiles** is a positive number and **lin\_tapeAutoDriveDumpFileOverWrite** is 1, each time a dump is obtained, lin\_taped writes that data to a file until **lin\_tapeMaxDumpFiles** is created. Then, when a new drive dump is detected, lin\_taped deletes the oldest drive dump file and creates a new file with the newest drive dump data. Thus, only the newest data is kept.

#### **Automatically retrieving SIM/MIM data:**

If a condition occurs in the drive such that a drive SIM/MIM data is created, lin\_taped retrieves the data and save it in a file named **lin\_tapex.timestamp.simmim**, where *lin\_tapex* is the device special file (for example, lin\_tape1, lin\_tape2) and **timestamp** reflects the time that the file was created. Each time SIM/MIM data is obtained, it is written to a new file. The IBM service organization might request that you forward SIM/MIM data to them for analysis.

The SIM/MIM data is stored in the **/var/log** directory by default. You can specify another directory in the **/etc/lin\_taped.conf** file.

There are two configuration parameters in the **/etc/lin\_taped.conf** file that you can tailor to affect the number of SIM/MIM files that are kept on your system.

- **lin\_tapeMaxSimMimDataFiles**, which can have a value of 0 or a positive decimal number.
- **lin\_tapeAutoSimMimDataOverWrite**, which can have a value of 0 or 1.

By default, **lin\_tapeMaxSimMimDataFiles** is 0 and **lin\_tapeAutoSimMimDataOverWrite** is 1, which means that every time SIM/MIM data is obtained, it is written to a new file.

If **lin\_tapeMaxSimMimDataFiles** is 0, **lin\_tapeAutoSimMimDataOverWrite** is ignored, and each time SIM/MIM data is obtained, it is written to a new file.

If **lin\_tapeMaxSimMimDataFiles** is a positive number and **lin\_tapeAutoSimMimDataOverWrite** is 0, each time SIM/MIM data is obtained, lin\_taped writes that data to a file until **lin\_tapeMaxSimMimDataFiles** is created. Then, lin\_taped stops creating new files, even if new SIM/MIM data is created.

If **lin\_tapeMaxSimMimDataFiles** is a positive number and **lin\_tapeAutoSimMimDataOverWrite** is 1, each time SIM/MIM data is obtained, lin\_taped writes that data to a file until **lin\_tapeMaxSimMimDataFiles** is created. Then, when new SIM/MIM data is detected, lin\_taped deletes the oldest SIM/MIM file and creates a new file with the newest SIM/MIM data. Thus, only the newest data is kept.

#### **Selective tracing:**

**Lin\_tape** provides facilities by which you can disable and enable tracing, error logging, auto-retrieving drive dumps, and auto-retrieving SIM/MIM data. You can selectively enable or disable them through an

application program, which uses the `STIOC_SETP ioctl`. These settings persist until the device driver is restarted, or the host system is rebooted.

The parameters and their definitions are as follows -

#### **trace**

This parameter is set to On by default, which enables **lin\_tape** tracing of activities and error logging on a particular tape drive. Set this parameter to off to stop tracing and error logging.

#### **logging**

This parameter is set to On by default and enables logging of log sense data. Setting this flag to Off suppresses volume logging for this device.

#### **disable\_sim\_logging**

This parameter controls the logging of SIM/MIM data for a device. By default it is set to Off, which causes SIM/MIM data to be logged. Set this flag to On to suppress the logging of SIM/MIM records.

#### **disable\_auto\_drive\_dump**

This parameter controls the saving of drive dumps for a device. By default it is set to Off, which causes drive dumps to be saved. Set this flag to On to suppress the saving of drive dumps.

#### **Running lin\_taped:**

If you are running the **lin\_tape** device driver, version 1.4.1 or higher, after installing **lin\_tape** `lin_taped` starts running even if your system does not have a tape device attached. If you add a new tape device into your Linux system, `lin_taped` automatically creates a special file under the `/dev` directory. If you are running the **lin\_tape** device driver, version 1.3.x or less, `lin_taped` does not automatically start if there is no tape device attached. After you attach a new tape device, you must start the `lin_taped` daemon.

You can start `lin_taped` from the command line. `lin_taped` takes zero or more of the parameters as listed in the following command.

```
lin_taped [start stop restart status]
```

#### **lin\_taped or lin\_taped start**

Starts the daemon. If there is already a `lin_taped` running, the new one is aborted. (Use **lin\_taped restart** if `lin_taped` is already running.)

#### **lin\_taped stop**

Terminates the daemon and frees all the resources that are associated with the daemon. When the daemon is stopped, no information is saved.

#### **lin\_taped restart**

Terminates the currently running daemon and starts a new one. The new daemon reads the `/etc/lin_taped.conf` file. This command is used after the `/etc/lin_taped.conf` file is modified while `lin_taped` is running.

#### **lin\_taped status**

Prints a message on **stdout** to indicate whether the daemon is running or not.

**Note:** If you run `rmmod lin_tape` command to remove the **lin\_tape** device driver from the running kernel, you must stop the `lin_taped` daemon first; otherwise you get a **Device or Resource Busy** error.

### **Reservation conflict logging**

When the device driver receives a reservation conflict on a tape drive command, it logs the conflict to the kernel debug buffer (which is typically echoed to `/var/log/messages`). Before the error is logged, the device driver determines whether a SCSI Persistent Reservation is active on the target tape drive. If it is, it gets the reserving host initiator WWPN (worldwide port name). If successful, the device driver posts the message

```
lin_tape: reserve held by xxxxxxxx
```

to the debug buffer. To prevent multiple identical entries in the error log, subsequent reservation conflicts from the same reserving host WWPN are not logged.

## Devices not reported at `/proc/scsi/IBMchanger` or `/proc/scsi/IBMtape`

After **lin\_tape** is installed, following the [Installation and Configuration instructions](#), this command shows all tape devices and their device index:

```
cat /proc/scsi/IBM*
```

If any or some devices that are expected to show up are not there, confirm that they are attached to Linux SCSI layer by:

```
cat /proc/scsi/scsi
```

If the devices do not show up there, review your HBA configuration and device attachment. If they are there, make sure that they are supported devices at SSIC. See [Hardware requirements](#).

If files `/proc/scsi/IBMchanger` and `/proc/scsi/IBMtape` do not exist, review [Installation and Configuration instructions](#), and confirm **lin\_taped** is running by using:

```
lin_taped status
```

and also that **lin\_tape** is correctly installed on SLES and RHEL by using:

```
rpm -qa lin_tape
```

.

On Ubuntu:

```
apt list --installed | grep lin-tape
```

## Windows Tape and Medium Changer device driver

This chapter describes the hardware requirements, software requirements, and installation notes for the Microsoft Windows device drivers for IBM tape devices.

### Purpose

The Windows tape and medium changer device driver is designed to take advantage of the features that are provided by the IBM tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape operations (such as backup and restore) and medium changer operations (such as mount and unmount the cartridges), and to the advanced functions needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application.

### Data flow

The software that is described here covers the Windows device driver and the interface between the application and the tape device.

[Figure 10 on page 67](#) illustrates a typical data flow process.

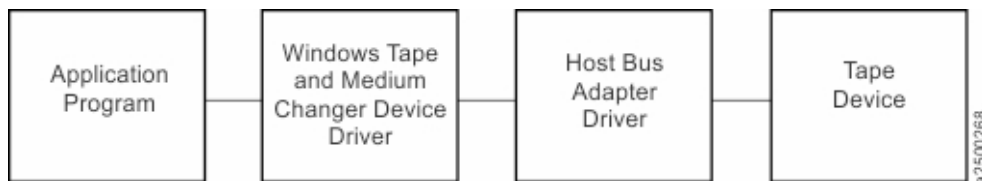


Figure 10. Data flow for Windows device driver (IBMTape)

# Product requirements

## Hardware requirements

Refer to the [“Hardware requirements” on page 2](#) for the latest hardware that is supported by the IBM tape device driver.

## Software requirements

For current software requirements, refer to the [“Software requirements” on page 2](#).

**Note:** Limited support for customers who have Microsoft Windows Server 2016 extended support from Microsoft only.

# Installation and configuration instructions

This section includes instructions for installing and configuring the Windows tape and medium changer device driver on Windows Server 2019, 2022 and 2025.

The recommended procedure for installing a new version of the device driver is to uninstall the previous version (see [“Uninstalling the device drivers” on page 70](#)).

## Windows Server 2019, 2022 and 2025 instructions

This section describes how to install, remove, and uninstall the Windows tape and medium changer device drivers on Windows Server 2019, 2022 and 2025.

## Installation overview

The installation process consists of the following steps:

1. Verify that the hardware and software requirements are met.
2. Install the host bus adapters and drivers.
3. Shut down the system.
4. Connect the tape and medium changer devices to the host bus adapters.
5. Power<sup>®</sup> on the tape and medium changer devices.
6. Restart the system.
7. Log on as **Administrator**.
8. Install and configure the devices and device drivers with the installation application.

## Installation procedures

These procedures make the following assumptions:

- No other driver is installed that claims the tape and medium changer devices.
- If you are updating the device driver from a Microsoft certified version to an uncertified version, you must first uninstall the certified driver. Refer to [“Uninstalling the device drivers” on page 70](#).
- The host bus adapter is installed, configured properly, and is running supported microcode and driver levels.
- Device drivers are installed with Windows Server 2019, with V6.2.6.8 or later until V7.0.1.4; Windows Server 2022, with V7.0.1.1 or later; Windows Server 2025, with V7.0.1.5 or later.

**Note:** The latest driver level to include support for Windows Server 2016 is V7.0.0.8.

**Note:** The latest driver level to include support for Windows Server 2019 is V7.0.1.4.



- Different registry keys are created to configure different parameters. They can be at System\CurrentControlSet\Services\ and the subkeys are created depending on the Windows Server version. With Windows 2016, they are:

- ibmcg2k16
- ibmtp2k16
- ibmtpbs2k16

Starting with Windows 2019 and later , subkeys are the same, they are:

- ibmcgbs
- ibmtpbs

Starting with V7.0.1.5, subkeys are stored under System\CurrentControlSet\Services\ibmXXXX\Parameters.

Refer to this list when in doubt of the registry key's name and instructions that involve modifying the registry. Caution and a backup are advised due to the registry's nature.

- Drivers are identified by the following conventions, where *nnnn* refers to a version of the driver. If there is more than one version, use the current one.

- Windows Server 2019 for extended 64-bit architectures (Intel EM64T and AMD64),

IBMTape.x64\_w19\_nnnn\_WHQL\_Cert.zip

- Starting with V7.0.1.1 all the supported OS versions for each driver version share the same files.

IBMTape.nnnn-x64\_WHQL\_Cert.zip

To install the device drivers, follow this procedure.

1. Log on as **Administrator**.
2. Download the appropriate driver. Refer to “[Accessing documentation and software online](#)” on page 77.
3. Extract the driver package to a hard disk drive directory of your choice, other than the root directory.
4. Ensure that the tape and medium changer devices are connected to your host bus adapter and configured properly by locating the devices in Device Manager.
5. Double-click either **install\_exclusive.exe** or **install\_nonexclusive.exe**.
  - **install\_exclusive.exe** The driver issues automatic reserves on open. It also prevents multiple open handles from the host to a drive from existing at the same time, as is required by applications such as Tivoli Storage Manager. This driver is also required for the failover feature to work as it uses persistent reservation (by default).
  - **install\_nonexclusive.exe** The driver permits open handles from the host to a drive to exist at the same time, as is required by applications such as Microsoft Data Protection Manager (DPM).

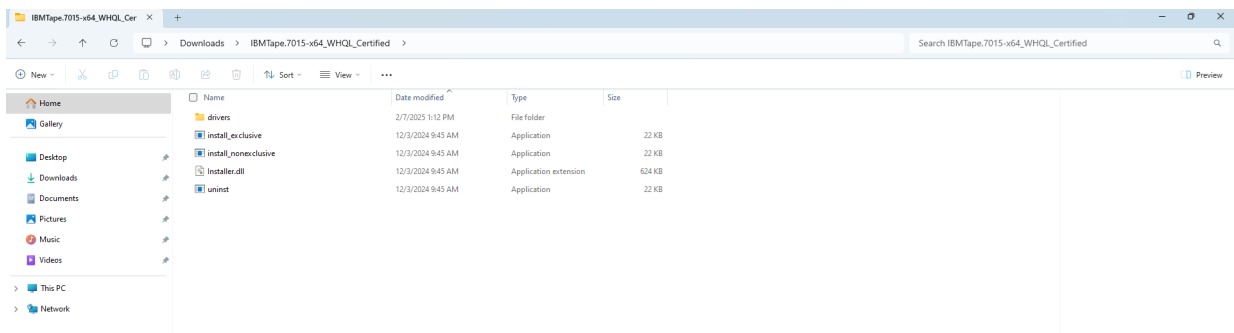


Figure 11. Installation application in Windows Explorer

**Note:**

a. More installation features are available through the command prompt interface (CLI), which include

- Installing only the tape or medium changer device drivers (**-t** or **-c**)
- Running in debug mode, which creates the file **debug.txt** in the driver package directory (**-d**)
- Running in silent mode, which suppresses messages that require user intervention, but only with Microsoft-certified IBM drivers (**-s**)
- Disabling DPF from installation (**-f**)
- Enabling Persistent Reserve from installation if DPF is disabled (**-p**)
- Disable Media Polling (**-m**).
- Disable Dynamic Runtime Attributes (**-a**).
- Exclude devices from being claimed by the driver (**-e**)
- Install checked version of tape and changer device drivers (**-x**), available on V7.0.1.5 and later versions

To install the device drivers with any of these features, instead of double-clicking the installation executable file, open a command prompt window and cd to the driver package directory. For the usage information, type `install_exclusive.exe -h` or `install_nonexclusive.exe -h` at the prompt.

b. If the Windows **Found New Hardware Wizard** begins during installation, cancel the wizard. The installation application completes the necessary steps.

6. To verify that the tape and medium changer devices and drivers are installed correctly, follow the instructions in [“Verifying correct attachment of your devices” on page 183](#).

## Device removal or disable procedure

If you must remove a device, or if you are altering the hardware configuration, you must uninstall or disable the device first.

1. Right-click **My Computer**, select **Manage** to open the Computer Management Console, and click **Device Manager**.
2. Right-click the device that you want to uninstall and select **Uninstall ...**. If you want to disable the device without uninstalling it, you can select **Disable**.
3. You are prompted to confirm the uninstallation. Click **OK**.
4. In Device Manager, under **System devices**, right-click **Changer Bus Enumerator** and select **Uninstall**.
5. In Device Manager, under **System devices**, right-click **Tape Bus Enumerator** and select **Uninstall**.

**Note:** This removal procedure removes the device from the device tree, but it does not uninstall the device driver files from your hard disk.

## Uninstalling the device drivers

To uninstall the device drivers from the system, which includes deleting the system files and deallocating other system resources, complete the following steps.

1. Quiesce all activity on the tape and medium changer.
2. Double-click **uninst.exe** in the driver package.

**Note:** This action removes all the files in the system directories that were created during the installation of the device driver. It does not delete the compressed file or the files that were extracted from the compressed file. If you want to remove these files, you must delete them manually. For v.6.2.5.3 and later, it is not required to manually remove the devices at the Device Manager.

3. Restart the system.

## Configuring limitations

The driver limitation for the supported number of tape devices is 1024. Every installed device uses a certain amount of resources. The user must also consider other resources, such as physical memory and virtual space on the system before you attempt to reach the limits. Also, be aware of Microsoft limitations. One known article is <http://support.microsoft.com/kb/310072> (ID: 310072). Be aware of any Windows specific version limitations.

## Persistent Naming Support on Windows Server 2019, 2022 and 2025.

The Windows tape driver has an option for enabling device object names that persist across restarts of the operating system. For example, if your tape drive has the name `\\.\tape4801101` and the persistent naming option is used, then `\\.\tape4801101` is reserved for use by that device after an operating system restart.

Complete the following steps to enable this feature.

1. Add a DWORD value to the registry called **PersistentNaming** and assign it a value 1
  - On Windows Server 2016:  
HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\ibmtp2k16
  - On Windows Server 2019 and Windows Server 2022 (For IBMTape 7.0.1.4 and before) :  
HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\ibmtp
  - On any supported Windows version starting on IBMTape 7.0.1.5 :  
HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\ibmtp\Parameters
2. Restart your system. Then, the system writes information to the registry to associate the Worldwide Node Name from Inquiry p. 0x83 with the persistent name used by the operating system.
  - If the Worldwide Node Name is unavailable, or the drive is a virtual (that is, emulated) drive, then the device serial number is used rather than the Worldwide Node Name.
  - If the **PersistentNaming** option is not specified in the registry, then your devices might not be able to claim the same device name after restart or driver initialization.

You can find registry subkeys with persistent naming information.

- On Windows Server 2016:  
HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\ibmtpbs2k16
- On Windows Server 2019 and Windows Server 2022(For IBMTape 7.0.1.4 and before):  
HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\ibmtpbs
- On any supported Windows version starting on IBMTape 7.0.1.5 :HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\ibmtpbs\Parameters

Alternately, you can use the Windows Device Manager to examine the device number to determine that persistent naming is enabled on your host. Persistent names contain tape device numbers that are based at 4801101 (which is the decimal equivalent of hexadecimal 0x49424D and ASCII "IBM").

If two physical paths exist to a drive and different Windows device names are required (which happens, for example, when two different HBAs are connected to the drive and Data Path failover is disabled), the first discovered path claims the persistent device name. Any subsequent paths that connect to the same device receive names according to the order in which they are discovered by the Windows Device Manager.

**Note:** Persistent naming is not set by default. For disabling it, set the **PersistentNaming** value to 0 and restart the system.

## Control Path failover support for tape libraries

To take advantage of Windows Control Path failover (CPF) support, the appropriate feature code must be installed. Refer to [“Automatic failover” on page 5](#) for what feature code might be required for your machine type.

### Configuring and unconfiguring Control Path failover support

Control Path failover is enabled automatically when the device driver is installed by default (install\_exclusive.exe). It can be disabled from installation with the `-f` CLI option. Or, it can be disabled or reenabled for the entire set of attached medium changers by modifying the registry.

1. Open the **reg** folder of the driver package.
2. Double-click **DisableCPF.reg** or **EnableCPF.reg**.
3. Reboot the system. This action is necessary for any registry modification to take effect.

### Querying primary and alternative path configuration

To check whether the control path failover is enabled in the device driver and show the primary and alternative paths, use the tape diagnostic and utility tool.

### Checking disablement of Control Path failover setting

If you disabled the control path failover in the device driver's setting by double-clicking the **DisableCPF.reg** file and rebooting your system, you can go into the registry by issuing the Windows **regedit** command to confirm that CPF is disabled. Look for FailoverDisabled exists and is set to 1

- On Windows Server 2016:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmcg2k16
- On Windows Server 2019 and Windows Server 2022 (For V7.0.1.4 and before):  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmcg
- On any supported Windows version starting on IBMTape 7.0.1.5 :  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmcg\Parameters

This line indicates that CPF is disabled in the driver. This setting takes effect only after your system is rebooted.

## Data Path failover support for tape drives

To take advantage of Windows Data Path failover (DPF) support, the appropriate feature code must be installed. Refer to [“Automatic failover” on page 5](#) for what feature code might be required for your machine type.

### Configuring and unconfiguring Data Path failover support

Data Path failover is enabled automatically when the device driver is installed by default (install\_exclusive.exe). It can be disabled from installation with the `-f` CLI option. Or, it can be disabled or reenabled for the entire set of attached drives or medium changers by modifying the registry.

1. Open the **reg** folder of the driver package.
2. Double-click **DisableDPF.reg** or **EnableDPF.reg**.
3. Reboot the system. This action is necessary for any registry modification to take effect.

For latest LTO generation 3 code and later LTO generation tape drives, a license key feature for the library hardware is required.

**Note:** For LTO generation 3 or lower, or for tape drives that require a data path license key on the host side to enable DPF, the device driver looks for a file that is called %system\_root%\IBM\_DPF.txt for the key. %system\_root% is the drive letter where Windows is installed, typically C (for example, C:\IBM\_DPF.txt). The file contains the key on a single line, with no spaces and no other text on the line. If multiple keys are required, place each key in the file on its own line. The driver looks for this file at initialization. If the file contains a valid DPF license key, the DPF feature is enabled and any eligible devices have multi-path support.

## Reserve Type if DPF is disabled

If DPF is disabled, SCSI-2 reserve is used by default to handle the reservation on tape drives.

If Persistent Reserve is wanted rather than SCSI-2 reserve, **ReserveTypePersistent.reg** enables it (ReserveTypeRegular.reg disables it and then SCSI-2 reserve is used). Or, -p CLI option installation (only if -f was used to disable DPF) enables Persistent Reserve from installation.

**Note:** If DPF is not disabled, Persistent Reserve is used.

## Querying primary and alternative path configuration

To check whether the data path failover is enabled in the device driver and show the primary and alternative paths, you can use the tape diagnostic and utility tool.

## Checking disablement of Data Path failover setting

If you disabled the data path failover in device driver's setting by double-clicking the **DisableDPF.reg** file and rebooting your system, you can go into the registry by issuing the Windows **regedit** command to confirm that DPF is disabled. Look for FailoverDisabled exists and is set to 1

- On Windows Server 2016:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp2k16
- On Windows Server 2019 and On Windows Server 2022 (For V7.0.1.4 and before):  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp
- On any supported Windows version starting on IBMTape 7.0.1.5:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp\Parameters

This line indicates that DPF is disabled in the driver. This setting takes effect only after your system is rebooted.

If you enabled Persistent Reserve on DPF disabled, look for ReserveType exists and is set to 1

- On Windows Server 2016:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp2k16
- On Windows Server 2019 and On Windows Server 2022 (For V7.0.1.4 and before):  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp
- On any supported Windows version starting on IBMTape 7.0.1.5:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp\Parameters

If you use **ReserveTypeRegular.reg**, then the ReserveType value is deleted.

## Problem determination

There is a debug version of the device driver that can be used if you encounter problems. The debug version of the driver issues DbgPrint messages at various places during device driver execution. To capture these messages, you must start a debugger or use a tool like **Debug View** from Sysinternals Suite, available from <http://technet.microsoft.com/sysinternals/>.

## Installing the debug version

### Prior V7.0.1.5 instructions

1. Quiesce all activity on the tape and medium changer.
2. Exit all applications that are using the tape and medium changer devices.
3. If the driver is installed, complete the **Uninstall the devices** procedure that is described in a previous section.
4. Locate the **\checked** folder for the device driver level that you are running. This folder is in the highest level directory of the driver package. It contains checked versions of the tape and medium changer device drivers.
  - a. on Windows Server 2016. **ibmtpxxyyy.sys**, and **ibmcgxyyy.sys**, where:
    - i) xx = **ft** for the filter driver, **bs** for the bus driver, or **blank** for the base driver
    - ii) yyy = **2k16**. It also has the **.inf** and **.cat** files.
  - b. on Windows Server 2019 and on Windows Server 2022. **ibmtpxx.sys**, and **ibmcgxx.sys**, where:
    - i) xx = **ft** for the filter driver, **bs** for the bus driver, or **blank** for the base driver. It also has the **.inf** and **.cat** files.
5. Copy all the files from the **\checked** folder to the highest level directory of the driver package, overwriting the files in there. A previous backup of the files or having a copy of the original driver package is recommended.
6. Start **Dbgview.exe** to capture debug messages. **Dbgview** can be downloaded from the Microsoft website. Make sure that **Capture Kernel**, **Enable Verbose Kernel Output**, and **Capture Events** are checked at the **Capture** menu.
7. Complete the Installation procedure.
8. Issue commands to the driver. You see debug messages on **Dbgview** from IBMTpBus, IBMCgBus, tape, or mcd. For example, you might see **IBMTpBus: ENT:** tag output. If you do not see something similar, an error might have happened during the checked driver installation or no driver activity might have occurred.

### For V7.0.1.5 and later instructions

1. Quiesce all activity on the tape and medium changer.
2. Exit all applications that are using the tape and medium changer devices.
3. If the driver is installed, complete the **Uninstall the devices** procedure that is described in a previous section.
4. Open a terminal and navigate to **IBMTape.nnnn-x64\_WHQL\_Cert** folder path, where the **install\_exclusive.exe** is located.
5. Execute the following command **.\install\_exclusive.exe -x**; where the **-x** flag means the debug version of the drivers are going to be installed.
6. Wait until the installation finishes.
7. Start **Dbgview.exe** to capture debug messages. **Dbgview** can be downloaded from the Microsoft website. Make sure that **Capture Kernel**, **Enable Verbose Kernel Output**, and **Capture Events** are checked at the **Capture** menu.
8. Complete the Installation procedure.
9. Issue commands to the driver. You see debug messages on **Dbgview** from IBMTpBus, IBMCgBus, tape, or mcd. For example, you might see **IBMTpBus: ENT:** tag output. If you do not see something similar, an error might have happened during the checked driver installation or no driver activity might have occurred.

## Restoring the non-Debug version

To restore the non-debug version of the driver, complete the following steps.

1. Quiesce all activity on the tape and medium changer devices.
2. Exit all applications that are using the tape and medium changer devices.
3. Uninstall the debug driver version, which can be accomplished by running **uninst.exe**, within the driver's package folder.

**Note:** For v.6.2.5.3 and later, there is no need to manually remove the devices at Device Manager.

4. Restart the system.
5. When the system is back, install the non-debug driver version.

## Reservation conflict logging

When the device driver receives a reservation conflict during open or after the device is opened, it logs a reservation conflict in the Windows eventlog. Before the error is logged, the device driver issues a **Persistent Reserve In** command. This action determines whether a SCSI Persistent Reservation is active on the reserving host to get the reserving host initiator WWPN (worldwide port name). If successful, the device driver logs this information as follows.

```
Reserving host key: kkkkkkkk WWPN: xxxxxxxx
```

Where *kkkkkkkk* is the actual reserve key and *xxxxxxx* is the reserving host initiator WWPN.

After initially logging the reserving host WWPN, subsequent reservation conflicts from the same reserving host WWPN are not logged. This action prevents multiple entries in the error log until the reserving host WWPN is different from the one initially logged. Or, the device driver reserved the device and then another reservation conflict occurs.

## Max retry busy

Complete the following steps to enable this feature.

1. Add a DWORD value to the registry called **MAXBusyRetry** and assign it a value between 1 and 480 at:
  - On Windows Server 2016:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp2k16
  - On Windows Server 2019 and Windows Server 2022 (V7.0.1.4 and before):  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp
  - On any supported Windows version starting on V7.0.1.5 :  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp\Parameters
2. Reboot your system. Then, when a device gets the **device busy** error, the command is retried up to *MAXBusyRetry* times.

## Checking if Data Path failover is enabled

If the tape diagnostic utility is not available, you can confirm that Data Path failover is enabled by installing the Debug version of the device driver. Refer to [“Problem determination” on page 73](#). While the Debug device driver is installing, look for the message: `tape.CreateTapeDeviceObject: DPF INSTALLED`.

## Checking if Control Path failover is enabled

If the tape diagnostic utility is not available, you can confirm that Control Path failover is enabled by installing the Debug version of the device driver. Refer to [“Problem determination” on page 73](#). While the Debug device driver is installing, look for the message: `FtlValidateCpfKey` and confirm that the status is `True`.

# IBM Tape Diagnostic Tool (ITDT)

---

This chapter describes the IBM Tape Diagnostic Tool.

## Purpose

The IBM Tape Diagnostic Tool (ITDT) is available in two versions:

- Standard Edition (ITDT-SE) - The command line version.
- Graphical Edition (ITDT-GE) - The GUI version for the following platforms:
  - Microsoft Windows operating systems
  - Linux operating systems

Both versions provide the user with a single diagnostic program for tapeutil applications. Both SE and GE contain tapeutil functions with SE also providing scripting capability.

**Note:** The term *tapeutil* is a synonym for the tool that is delivered with the device driver. For example, this tool is named **tapeutil** on UNIX operating systems; it is named **ntutil** on Microsoft Windows operating systems.

ITDT supports IBM tape and library devices claimed by the IBM Tape Device Driver and devices that use the generic Operating System driver. By default the IBM Tape Device Driver is used but only if the IBM Tape Device Driver is not installed or the device is not claimed by the IBM Tape Device Driver. The generic Operating System driver is used when available.

The available advanced operations that are provided by the IBM Tape Diagnostic Tool are completed on tape drives and tape libraries. By using this function, you can complete maintenance tasks and run diagnostic tasks to determine tape drive issues. This action reduces product downtime and increases productivity.

The IBM Tape Diagnostic Tool is designed to

- Run quick or extended diagnostic tests on tape drives.
- Start tape library self-test operations.
- Retrieve dumps from tape drives and libraries.
- Run a firmware update on tape drives or libraries.
- Online Check for Firmware updates for tape drives or libraries (ITDT-GE).
- Test the performance of the environment by completely writing a cartridge and measuring performance.
- Verify tape drive compression.
- Measure system performance.
- Retrieve and display cartridge usage information.
- Verify the encryption environment.

This test is used to verify whether data on the cartridge was written encrypted.

- Scan the system to discover all supported tape and library devices.
- Run a connection test (scan).

This test is used to verify that all devices are attached properly.

- Run a standard test to check whether the tape device is defective and output a pass/fail result.

**Note:** When this test is completed, all data on the cartridge is overwritten.

- Run a full write function.

This function writes the entire cartridge, overwriting all previous data with a selectable block size that contains either compressible or incompressible data and then outputs performance data.

**Note:** When this test is completed, all data on the cartridge is overwritten.



- Run a system test.

Write different block sizes with compressible and incompressible data and then outputs performance data.

**Note:** When this test is completed, all data on the cartridge is overwritten.

- Run a tape usage function to retrieve statistical data and error counters.
- Run HD-P functions like discovery.
- Physical Copy/Data migration and verification of cartridges.
- Log and Dump File analysis.

ITDT-SE provides the most important functions of the previous tapeutil tools. As an extension of the current tapeutil variants, the set of operations and functions available with ITDT-SE is identical across all supported operating systems (unless a function is not available on a particular system).

Dedicated device drivers for tapes and libraries can be installed on the target system and an application is installed that uses the tape/library devices. When this configuration exists, ITDT-SE can coexist with the application so that when the application disables the device internally, ITDT-SE can run the diagnostic tests on that device.

## Accessing ITDT

IBM maintains the latest levels of the ITDT tool chain and documentation on the Internet at <http://www.ibm.com/support/fixcentral>.

One option to access ITDT is through “Accessing documentation and software online” on page 77. When an IBM driver is downloaded, there is a corequisite to download ITDT.

This portal gives access to the download area where the following procedure guides you to the correct download:

1. In the **Product Group** menu, select **Storage Systems**.
2. In the **Product Family** menu, select **Tape Systems**.
3. In the **Product Type** menu, select **Tape device drivers and software**.
4. In the **Product** menu, select **Tape Diagnostic Tool (ITDT)**.
5. Select your platform and press **Continue**.

## Accessing documentation and software online

IBM maintains the latest levels of System Storage tape drive and library device drivers and documentation on the Internet. Fix Central is a portal where you can download the latest version of drivers for most of the IBM tape products. In this download area, you find menus that can guide you to find what you need.

To access to tape Device Drivers or software downloads, click

[http://www-933.ibm.com/support/fixcentral/?productGroup0=System%20Storage&productGroup1=ibm/Storage\\_Tape&productGroup2=Tape%20drivers%20and%20software&productGroup3=ibm/Storage\\_Tape/Tape%20device%20drivers](http://www-933.ibm.com/support/fixcentral/?productGroup0=System%20Storage&productGroup1=ibm/Storage_Tape&productGroup2=Tape%20drivers%20and%20software&productGroup3=ibm/Storage_Tape/Tape%20device%20drivers). Choose your platform, then click

**Continue**. Alternately, you can follow these steps.

1. Access the Fix Central URL at <http://www.ibm.com/support/fixcentral>.
2. In the **Product Group** menu, select **System Storage**.
3. From the **System Storage** menu, select **Tape systems**.
4. In the **Tape systems** menu, select **Tape drivers and software**.
5. In the **Select from Tape drivers and software** menu, select **Tape device drivers**. You are also able to get tools such as ITDT.
6. The **Platform** menu displays. Select the platform that you are looking for, then click **Continue** to view what is available for the selected platform.

7. In the next screen, there is a list of the latest Tape device drivers or Tape Diagnostic tools versions. Select the package that you need, then click **Continue**.

**Note:**

- The latest driver has the highest number extension. Numeric sequence numbers are in each level of the device and library driver. So, for example, for AIX the number is Atape.11.7.5.0.bin. As newer levels of a driver are released, a higher numeric sequence is assigned. When a driver is downloaded, ITDT also appears as a recommended download if you selected the option of co-requisites and prerequisite.
  - As of January 31, 2012, each IBM client that accesses Fix Central (whether through their employees or other authorized representatives) is required to have an individual IBM ID to download fixes (some exemptions might apply). The registration is quick and simple and provides users with a customized experience to better serve their needs.
8. After you click **Continue**, you can choose the way that you want to download the fix or fixes. You can either use your browser (HTTP), the Download Director ([http://www6.software.ibm.com/dldirector/doc/DDfaq\\_en.html#Q\\_A1](http://www6.software.ibm.com/dldirector/doc/DDfaq_en.html#Q_A1)), or bulk FTP option. The bulk FTP option is a web service where you can download a package with the FTP command or other download commands like WGET.
- Note:**
- To use the Download Director, ensure that the Java™ Runtime Environment is installed on your computer. For information, see [http://www6.software.ibm.com/dldirector/doc/DDfaq\\_en.html](http://www6.software.ibm.com/dldirector/doc/DDfaq_en.html). Leave the check box for prerequisites and co-requisites selected. Click **Continue**.
  - When you select the download option before **Continue** is pressed, you can change the way that you want to download the fix or fixes from the **Change download options** link at the top of the page.
9. Sign in, then click **Submit**.
10. Click **I agree** on the **Terms and Conditions** screen to continue.
11. A list of files and downloadable files displays, including links to Documentation, Programming guides, User guides, and readme files.

**Note:** Some plain text files might require you to right-click, then select **Save Link As...** to download the file.

Table 11 on page 78 documents each driver by name and description.

Table 11. Driver descriptions	
Driver	Description
Atape.n.n.n.n.bin	AIX Device Driver (Atape)
/lin_tape_source-lin_taped/lin_tape-x.x.x.x-x.src.rpm.bin	Linux Device Driver (lin_tape) source code
/lin_tape_source-lin_taped/lin_taped-x.x.x-dist.arch.rpm.bin	Linux lin_taped daemon program
/IBMtape.n.n.n.n.bin	Solaris Device Driver (IBMtape)
/IBMTape.arch_wXX_nnnn.zip	Windows Server 20XX Driver on arch (x86, x64) where XX denotes the version (such as, 2008, 2012, 2019)

Table 11. Driver descriptions (continued)

Driver	Description
<b>Note:</b> <ol style="list-style-type: none"> <li>1. Valid for Windows 2008 and Windows 2008 r2</li> <li>2. <b>dist</b> indicates a Linux distribution. <b>arch</b> indicates a machine architecture (for example, i386, ia64, s390).</li> <li>3. The <i>n.n.n.</i> or <i>n.n.n.n</i> strings are replaced with digits to reflect the version of each driver.</li> </ol>	

- For details on supported tape attachment, refer to the **System Storage Interoperation Center** website - <http://www.ibm.com/systems/support/storage/config/ssic/>.
- Information concerning supported Fibre Channel host bus adapters (HBAs) and associated HBA device drivers, firmware, and BIOS levels can be obtained from the **System Storage Interoperation Center** website - <http://www.ibm.com/systems/support/storage/config/ssic/>.
- The **IBM\_Tape\_Driver\_IUG.pdf** file contains the current version of the *IBM Tape Device Drivers: Installation and User's Guide*, which can be found here: <http://www-01.ibm.com/support/docview.wss?rs=577&iud=ssglS7002972>. You can also use the links that are located under **Related Information** to get quick access to the download section for the tape device drivers.
- The **IBM\_Tape\_Driver\_PROGREF.pdf** file contains the current version of the *IBM Tape Device Drivers: Programming Reference*, which can be found here: <http://www-01.ibm.com/support/docview.wss?uid=ssglS7003032>.
- For the current information for the device driver you are using, consult the readme file (not files) included in the download of your device driver.

**Use the links listed below for quick access to the download section for every tape Device Driver platform.**

- **AIX.** Path: **Storage Systems > Tape Systems > Tape device drivers and software > Tape device drivers > AIX**, or [http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage\\_Tape/Tape+device+drivers&release=1.0&platform=AIX&function=all](http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage_Tape/Tape+device+drivers&release=1.0&platform=AIX&function=all).
- **Linux.** Path: **Storage Systems > Tape Systems > Tape device drivers and software > Tape device drivers > Linux**, or [http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage\\_Tape/Tape+device+drivers&release=1.0&platform=Linux&function=all](http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage_Tape/Tape+device+drivers&release=1.0&platform=Linux&function=all).
- **Solaris.** Path: **Storage Systems > Tape Systems > Tape device drivers and software > Tape device drivers > Solaris**, or [http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage\\_Tape/Tape+device+drivers&release=1.0&platform=Solaris&function=all](http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage_Tape/Tape+device+drivers&release=1.0&platform=Solaris&function=all).
- **Windows.** Path: **Storage Systems > Tape Systems > Tape device drivers and software > Tape device drivers > Windows**, or [http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage\\_Tape/Tape+device+drivers&release=1.0&platform=Windows&function=all](http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage_Tape/Tape+device+drivers&release=1.0&platform=Windows&function=all).
- **Windows WHQL.** Path: **Storage Systems > Tape Systems > Tape device drivers and software > Tape device drivers > Windows WHQL**, or [http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage\\_Tape/Tape+device+drivers&release=1.0&platform=Windows+WHQL&function=all](http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~ST~Tapedevicedriversandsoftware&product=ibm/Storage_Tape/Tape+device+drivers&release=1.0&platform=Windows+WHQL&function=all).

## Supported systems

ITDT is developed to support various versions of different platforms and tape products. For the latest support, refer to “Introduction” on page 2 and Inter-operation Center website product support at <http://www.ibm.com/systems/support/storage/config/ssic/>.

# IBM Tape Diagnostic Tool - Standard Edition

## Installing ITDT - Standard Edition

This section describes the installation procedures for the Standard Edition of ITDT in various operating systems.

Before the IBM Tape Diagnostic Tool Standard Edition (ITDT-SE) is used with the IBM Tape Device Driver, we recommend upgrading to the latest available IBM Tape Device Driver level.

The [U] Tapeutil option works only if the IBM Tape Device Driver is installed. The **-force-generic-dd** startup option to bypass the IBM Tape Device Driver has no effect on the [U] Tapeutil option commands. They still use the IBM Tape Device Driver if it is installed.

In a System Managed Encryption setup, the Encryption test [E] always exits with NO DRIVER SPECIAL FILE when ITDT-SE is started with **-force-generic-dd**.

### Installing ITDT-SE on AIX operating systems

ITDT-SE on AIX operating systems has two installation modes and can be installed by one of the following methods.

1. Installation in the AIX software repository:

- a. Download **tape.itdt.<version>.bff**
- b. Change to the download folder and run the **inutoc** command.
- c. To install the file, run the following command.

```
installp -acd tape.itdt.<version>.bff all
```

Or

```
smitty install
```

2. Installation independently.

- a. Download **install\_itdt\_se\_<OS>\_<version>** to a directory of your choice.
  - **install\_itdt\_se\_Aix\_<version>** is for AIX® operating systems.
- b. Run the following command to make **install\_itdt\_se\_<OS>\_<version>** executable.

```
chmod 700 install_itdt_se_Aix_<version>
```

3. To run without parameters, run the following command.

```
install_itdt_se_Aix_<version>
```

Or

```
./install_itdt_se_Aix_<version>
```

depending on your operating system.

**Note:** ITDT-SE can be used only by a user with root access rights.

### Installing ITDT-SE on Microsoft Windows operating systems

To install ITDT-SE on Microsoft Windows operating systems, complete the following steps.

**Note:** ITDT-SE can be used only by a user with administrator rights.

1. Download **install\_itdt\_se\_WindowsX86\_64\_<version>.exe** to a directory of your choice.
2. Run the following command:

```
install_itdt_se_WindowsX86_64_<version>.exe
```

## Installing ITDT-SE on IBM System i

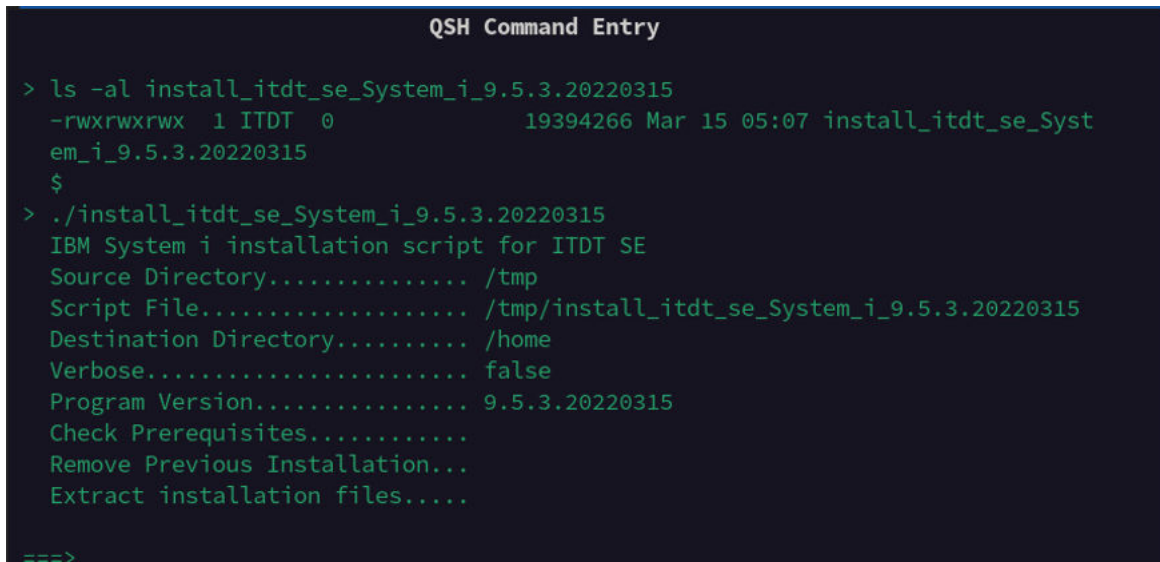
ITDT's installation package for IBM System i is a single file for a command-line based installation either in the Qshell or BASH environment. For this reason, the file needs to be manually transferred to the system i host.

### Procedure

To install ITDT-SE on IBM System i, complete the following steps.

**Remember:** ITDT-SE can be used only by a user with QSECOFR access rights.

1. Download **install\_itdt\_se\_System\_i\_version** to a directory of your choice.
2. Transfer the installation file to a folder within the IBM System i host IFS file system, such as /tmp.
3. Connect to the IBM System i host by using any terminal emulation program.
4. Run the QSH command and change the directory to the folder where the ITDT installation program is stored. See [#unique\\_152/unique\\_152\\_Connect\\_42\\_fig\\_jmv\\_qvh\\_ltb](#) on page 81 for reference.



```
QSH Command Entry

> ls -al install_itdt_se_System_i_9.5.3.20220315
-rwxrwxrwx  1 ITDT  0                19394266 Mar 15 05:07 install_itdt_se_Syst
em_i_9.5.3.20220315
$
> ./install_itdt_se_System_i_9.5.3.20220315
IBM System i installation script for ITDT SE
Source Directory..... /tmp
Script File..... /tmp/install_itdt_se_System_i_9.5.3.20220315
Destination Directory..... /home
Verbose..... false
Program Version..... 9.5.3.20220315
Check Prerequisites.....
Remove Previous Installation...
Extract installation files.....

===>
```

Figure 12. QSH command entry

5. Run the following command to make **install\_itdt\_se\_System\_i\_version** an executable file:  
`chmod 700 install_itdt_se_<OS>_version`
6. Run the installation file.

**Tip:** You can also use the following parameters as needed:

- --help displays help message.
- --verbose displays verbose information.
- --uninstall uninstalls ITDT.
- --version displays version information.

### Results

After successful installation, the ITDT program is located in the following folder: /home/ITDT.

## Installing ITDT-SE on other supported operating systems

To install ITDT-SE on other supported operating systems, complete the following steps.

**Note:** ITDT-SE can be used only by a user with root access rights, except for the Mac OS, which requires the user to have the minimum of read/write access to the device file.

1. Download **install\_itdt\_se\_<OS>\_<version>** to a directory of your choice.
  - **install\_itdt\_se\_Aix\_<version>** is for AIX operating systems
  - **install\_itdt\_se\_Linuxx86\_64\_<version>** is for Linux operating systems on X86\_64 hardware
  - **install\_itdt\_se\_Linuxpowerpc64\_<version>** is for Linux operating systems on pSeries
  - **install\_itdt\_se\_Linuxpowerpc64le\_<version>** is for Linux operating systems on pSeries Little Endian.
  - **install\_itdt\_se\_Linuxs390x\_<version>** is for Linux operating systems on zSeries
  - **install\_itdt\_se\_MacOS\_<version>** is for Mac OS operating systems.
2. Run the following command to make **install\_itdt\_se\_<OS>\_<version>** executable.

```
chmod 700 install_itdt_se_<OS>_<version>
```

3. To run without parameters, run the following command

```
install_itdt_se_<OS>_<version>
```

or

```
./install_itdt_se_<OS>_<version>
```

depending on your operating system.

Optionally you can use these parameters:

- **-h** [--help] Show help message
- **-v** [--verbose] Print the extracted files
- **-p** [--path] Path for extraction (default is current directory)
- **-t** [--test] Show package content

## Starting ITDT - Standard Edition

This section describes the startup procedures for the Standard Edition of ITDT in various operating systems.

### Starting ITDT-SE on Solaris operating systems

1. If the IBM Tape Device Driver is not used, if it is not installed, or you want to be able to run **-force-generic-dd**, configure the *sgen* driver.

A script **sgen\_solaris\_conf.sh** is included in the ITDT-SE package. This script allows you to configure the *sgen* generic SCSI driver that is shipped with Solaris.

ITDT-SE requires that the *sgen* device driver is configured so that the devices you want to work with can be found.

**Note:** For system security reasons, always reset the *sgen* device driver settings after you finish working with ITDT-SE, by using the `sgen_solaris_conf.sh` script.

To configure the *sgen* driver, start the **sgen\_solaris\_conf.sh** script with root access.

The following command-line options are available on the *SGEN* driver configuration screen:

1. Check driver: This option checks if the driver **/kernel/drv/sgen** is available.
2. List driver settings: This option shows the current activated devices.
3. New driver configuration: This option shows the screen that is used to create a new driver configuration (see Step 2).

- 4. Stop *sgen* driver: This option stops the driver (that is, `rem_drv sgen`).
- 5. Start *sgen* driver: This option stops and starts the *sgen* driver.
- 6. Exit program: This option closes the shell script.

2. Enter option 3 to create a new driver configuration.

The following command-line options are available on the **New Configuration** screen. Use these options to configure the *sgen* driver:

- 1. List targets: This option shows the targets in current configuration.
- 2. Define device types: This option defines drive and changer types.
- 3. or 5. Add targets: This option adds targets to the list.

**Note:** Option 3 allows for the addition of individual devices one at a time. Option 5 allows for the addition of a range of devices, eliminating the need to add many devices one by one.

- 4. or 6. Remove targets: This option removes targets from the list.

**Note:** Option 4 allows for the removal of individual devices one at a time. Option 6 allows for the removal of a range of devices, eliminating the need to remove many devices one by one.

- 7. Save configuration: This option saves the modifications.
- 8. Back to Main Menu: This option returns to the main menu.

3. After the *sgen* driver is configured, enter command-line option 8 to go back to the main menu.

4. On the *SGEN* driver configuration screen, enter command-line option 5. This option starts the *sgen* driver. New devices are found by using the definitions that are completed in Step 2.

5. After the new devices are found, enter option 6 to exit the **`sgen_solaris_conf.sh`** script.

**Note:** For Fibre Channel Host Bus Adapters (HBAs), special handling is required. Attached devices must be configured with their WWPN in the **`sgen.conf`** file. This task must be done manually. It is not completed by using the `sgen_solaris_conf.sh` script.

The following code is an example how to add those devices:

```
Run the command "cfgadm -al" to get the WWPN number(s).
.....
c4                                fc-private  connected  configured
c4::5005076302401924             tape        connected  configured
.....
Add the WW-PN number(s) into the sgen.conf file.
name="sgen" parent="fp" target=0 lun=0 fc-port-wwn="5005076302401924";
name="sgen" parent="fp" target=0 lun=1 fc-port-wwn="5005076302401924";

If you have finished the editing, the sgen driver has to be restarted.
Please enter "update_drv sgen".
```

The **Read Attribute** command is 16-byte CDB. On Solaris 10, IBMtape can detect what maximum CDB length is supported from HBA attributes and set the supported CDB. For Solaris 9, CDB16 must be enabled for ITDT-SE to work correctly. This procedure can be done by adding the entry of

```
cdb16_support=1
```

at the first line in

```
/usr/kernel/drv/IBMtape.conf
```

and reload the IBMtape driver again.

To start ITDT-SE, run the following command:

```
./itdt
```

At first start, read the User License Agreement.

- Press **Enter** to scroll the license screens forward or b followed by **Enter** to go back.

- Type **i** if you agree to the terms of license or **q** followed by **Enter** to quit the application.

During the initial program start, the input and output directories are created.

- Input directory: default directory for firmware files during Firmware Update
- Output directory: directory that contains the result files, dump files, and log files after tests are run

ITDT-SE does not make changes outside the installation directory.

### ***Starting ITDT-SE on Windows operating systems***

At first program start, the license text is displayed and the input and output directories are created. Start ITDT by running the following command from the directory "ITDT" that was created during the installation (confirm to run with administrator rights):

```
itdt.exe
```

Read the User License Agreement.

- Press **Enter** to scroll the license screens forward or **b** followed by **Enter** to go back.
- Type **i**, if you agree to the terms of license or **q** followed by **Enter** to quit the application.

ITDT-SE does not create any registry entries or make changes outside the installation directory.

During installation a subdirectory "ITDT" was created which contains the ITDT program file and 2 subdirectories initially:

- "License" directory with license files in different languages.
- "Scripts" directory for extra scripts.

When the program is run for the first time, two more subdirectories are created in the "ITDT" folder:

- "Input" directory for firmware files to be uploaded to devices.
- "Output" directory for generated log and dump files.

To remove ITDT from your system, erase the ITDT directory. Any log and dump files in the subdirectories are also erased when you do so.

### ***Starting ITDT-SE on i5/OS operating systems***

To use ITDT-SE to update firmware or pull dumps from a tape drive inside a tape library, make sure that the drives are varied online in STANDALONE MODE by completing the following steps.

1. Issue the command **WRKMLBSTS**. Identify the library and drives you want to work with. Note their names (for example, TAPMLB01, TAP01, TAP02).
2. Deallocate the corresponding drives by using option 6.
3. Vary OFF the TAPMLB by using option 2.
4. Enter the following command:

```
WRKCFGSTS *DEV TAP*
```

Identify the drives that were noted in Step 1 (for example, TAPMLB01, TAP01, TAP02) and vary them ON by using option 1.

5. Start the iSeries Qshell environment with the following command:

```
QSH
```

6. Change to the folder /home/ITDT with the following command:

```
cd /home/ITDT
```

7. Start ITDT with the following command.



```
./itdt
```

8. Update Firmware and pull dumps. See [“Firmware Update” on page 96](#) and [“Dump” on page 95](#).
9. When firmware updates and dumps are complete, enter the following command:

```
WRKCFGSTS *DEV TAP*
```

10. Vary off the TAPs that you worked with by using option 2.
11. Issue the command **WRKMLBSTS**. Identify the library and drives you worked with.
12. Vary on the TAPMLB by using option 1.
13. Press **F5** to refresh the screen. The TAPs belonging to the TAPMLB shows up.
14. Allocate the TAPs back to the TAPMLB by using option 4 or 5.

### ***Starting ITDT-SE on other supported operating systems***

To start ITDT-SE, run the following command

```
./itdt
```

At first start, read the User License Agreement:

- Press **Enter** to scroll the license screens forward or b followed by **Enter** to go back.
- Type i, if you agree to the terms of license or q followed by **Enter** to quit the application.

During the initial program start, the input and output directories are created.

- Input directory: default directory for firmware files during Firmware Update
- Output directory: directory that contains the result files, dump files, and log files after tests are run

ITDT-SE does not make changes outside the installation directory.

### **Generic Operating System driver with ITDT-SE**

When the IBM Tape Device Driver cannot be used, ITDT-SE can be forced to use generic Operating System driver devices with the parameter **-force-generic-dd**.

The parameter is supported in the interactive and scripting mode of ITDT-SE.

For information, go to:

- [“Standard Edition - Program options” on page 118](#)
- [“Device file names - device addressing” on page 108](#)
- [“Scan” on page 91](#)

### **Standard Edition - known issues and limitations**

This section describes the known issues and limitations of the ITDT-SE program.

#### ***AIX operating systems***

The following are the known scan limitations:

- Only devices that have the device state "available".

For FC and SAS devices, ID and LUN greater than 999 are not displayed; they are shown as ###.

When logged in through telnet, backspace might not work - an escape sequence is inserted and the input is ignored after **Enter** is pressed.

## HP-UX operating systems

Verify that the following patches are installed before ITDT-SE is started.

- PA-Risc: At least these patches:
  - ld and linker tools cumulative patch
  - libc cumulative patch
- Itanium/IA-64: All regular patches and the following patches:
  - VxVM 3.5~IA.014 Command Patch
  - VxVM 3.5~IA.014 Kernel Cumulative Patch
  - Aries cumulative patch
  - linker + fdp cumulative patch

**Note:** ITDT-SE is emulated by Aries (a binary emulator that transparently emulates 32-bit and 64-bit HP-UX PA-RISC applications on HP-UX IA-64 machines).

On HP-UX11.iV3 systems, tape libraries that are operated through the drive's control path (no control path failover) might disappear from the Device List after a **[F] Firmware Update** on the controlling drive. It is recommended to complete repeated **[S] Scan** operations to make the library reappear in the device list.

## Linux operating systems

ITDT-SE on Linux requires `glibc 2.2.5` or later.

**Note:** On an SLES9 s390x (64 bit) configuration, you might experience a SCSI CMD TIMEOUT when the [T] option is run with the IBM Tape Device Driver.

For SUSE SLES9 on zSeries, ensure that the kernel update `SUSE-SA:2007:035` is installed.

## Solaris operating systems

Rescan might take 6-8 minutes, depending on the numbers of host adapters and devices attached.

The known scan limitations: SCSI ID 0-255, LUN 0-10.

If the IBM Tape Device Driver is not installed on Solaris 10, tape devices might not be found during scan although they are configured in `sgen.conf`. When this event occurs, complete the following steps to configure the devices:

1. Check the current driver bindings for IBM tape drives and changers by entering the following commands:

```
# egrep "scsiclass,01" /etc/driver_aliases      (for drives)
# egrep "scsiclass,08" /etc/driver_aliases      (for changers)
```

2. Modify the `/etc/driver_alias` file to comment all lines not starting with `sgen` and containing identification of your drives and changers. Examples:

```
#st "scsiclass,01"                (all tape drives)
#st "scsiclass,01.vIBM.pULT3580-TD4" (IBM tape drive model ULT3580-TD4)
#st "scsiclass,08"                (all changers)
#st "scsiclass,08.vIBM.p3573-TL"   (IBM changer model 3573-TL)
```

3. Check that the configured drives are not configured for `st` driver by entering the following command:

```
# cfgadm -al
```

If the tape drive is claimed by `st` device driver, an entry with `cxx: rmt/y`, is displayed, for example:

```
c11::rmt/0      tape      connected      configured      unknown
```

4. Add `sgen` driver aliases with one of the following commands:

```
# update_drv -a -i '"scsiclass,01.vIBM.pULT3580-HH4"' sgen
(adds sgen alias for IBM drive, model ULT3580-HH4)

# update_drv -a -i '"scsiclass,01"' sgen
(adds sgen alias for all drives attached to the system)

# update_drv -a -i '"scsiclass,08.vIBM.pULT3581-TA2"' sgen
(adds sgen alias for IBM changer, model ULT3581-TA2)

# update_drv -a -i '"scsiclass,08"' sgen
(adds sgen alias for all changers attached to the system)
```

5. Check that the drives and changers are now configured with the following command:

```
# cfgadm -al
```

6. If the drives or changers are not listed in the output of 'cfgadm -al', reboot the system and verify the list of configured devices with the command:

```
# cfgadm -al
```

## Windows operating systems

After a firmware update, devices might disappear. This issue is a known Windows problem.

Repeated **Scan** operations can help to rediscover the device.

When applications are turned on Windows while ITDT-SE is running, an extra ESC character might appear on the input line. When this issue occurs, the input is ignored after **Enter** is pressed.

If you are using Adaptec SCSI Host Bus adapters, ensure that you are using the latest Adaptec Host Bus Adapter Drivers instead of the drivers that are shipped with the Windows operating system.

On Microsoft Windows systems where the maximum transfer size is limited to less than 64 kB, the **Dump** and **Firmware Update** operations do not work.

## i5/OS operating systems

The Tape Drive must be varied online. If the tape drive is operated through a tape library, the library must be varied offline. See [“Starting ITDT-SE on i5/OS operating systems” on page 84](#) for details.

As the library is varied offline, the Encryption Test does not deliver decrypted data in a Library Managed Encryption environment.

IO adapters without a dedicated IOP do not support commands with 16 bytes. These adapters are called IOP-less and can be identified with the command; **WRKHDWRSC \*STG**. Adapters where the CMBxxx and DCxxx resource names are the same (like both 5774) are IOP-less. If it is IOP, then the CMBxxx is a different type from the DCxxx.

Therefore, the following ITDT commands and test sequences can fail on IOP-less adapters attached devices.

- Encryption Verification Test
- Physical Copy with more than one partition
- **Erase** command
- **Change Partition** command
- **Read/write -attribute** command

ITDT-SE on i5/OS V5R4 requires the following PTF installed.

- PTF: SI25023 Release: 540 Abstract: OSP-MEDIA-TAPE-THREADS-MSGCEE0200-T/QTAHRMGR QTARDCAP FAILS

The [U] Tapeutil option is not available for i5/OS with this release as all the underlying operations require the IBM Tape Device Driver to be installed.

FC 5912 SAS HBA support is only for POWER6 and V6R1 configurations that are attached to LTO Gen 4 HH tape drives (No support for LTO 3 HH SAS).

## **All supported operating systems**

This section describes the known issues and limitations of the ITDT-SE program on all other supported operating systems.

Prevent/Allow Medium Removal is missing as a Tape Drive option. But, it can still be completed by using the [56] Prevent/Allow Medium Removal option for tape libraries while the tape device is opened.

## **User Interface issues**

If you press the arrow keys on most UNIX operating system consoles, the input is ignored after **Enter** is pressed.

When the **Tab** key is pressed as an input string for field data, the user interface is corrupted.

Make sure that field input does not start with a number followed by space and extra text. This input is interpreted as an entry to a specific row in the field. To avoid this issue, use an underscore character ( \_ ) instead of the space character.

## **Command timeout**

There is no instant operation termination upon SCSI command timeout; for example, when the SCSI cable is unplugged after POST A is started.

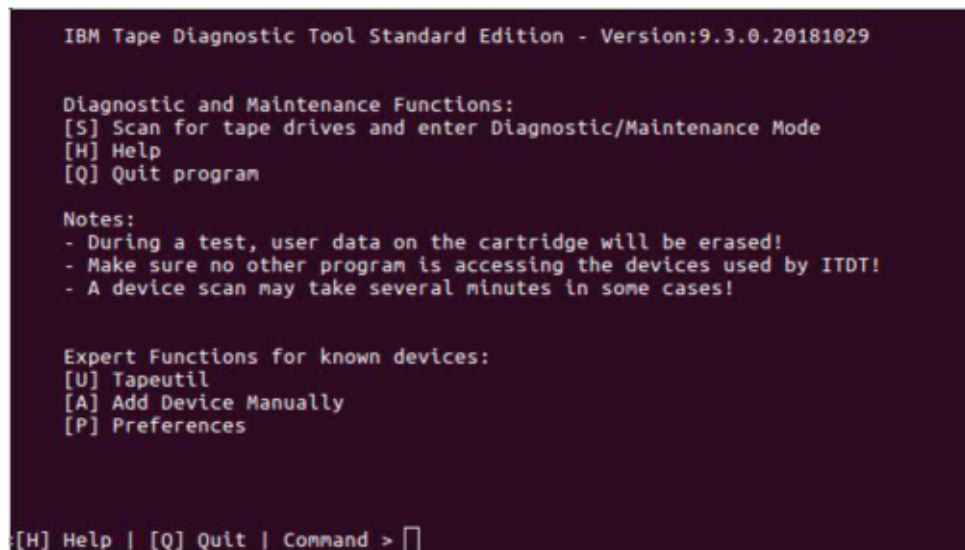
When a command timeout condition occurs, ITDT might still continue to complete more operations (like unmounting the cartridge) instead of instantly terminating with a timeout condition.

## **TS3310 tape libraries**

Library Firmware Update with ITDT-SE and ITDT-GE is not supported by the TS3310 libraries. Update the firmware by using the Web User Interface for those libraries.

## **Standard Edition - Start menu commands**

After program startup, ITDT-SE displays the start screen menu.



```
IBM Tape Diagnostic Tool Standard Edition - Version:9.3.0.20181029

Diagnostic and Maintenance Functions:
[S] Scan for tape drives and enter Diagnostic/Maintenance Mode
[H] Help
[Q] Quit program

Notes:
- During a test, user data on the cartridge will be erased!
- Make sure no other program is accessing the devices used by ITDT!
- A device scan may take several minutes in some cases!

Expert Functions for known devices:
[U] Tapeutil
[A] Add Device Manually
[P] Preferences

[H] Help | [Q] Quit | Command > 
```

Figure 13. Start screen menu

The following commands are available on the start screen menu:

- [S] Scan for tape drives and enter Diagnostic/Maintenance Mode

Opens the screen for the Scan function (refer to [“Standard Edition - Scan menu commands”](#) on page 89).

- [H] Help

Help starts and displays the available online help.

- [Q] Quit program

Quits the program.

- [U] Tapeutil

Opens the screen for the *Tapeutil* operation commands. These commands are the standardized *tapeutil* functions with most of the options available that were available with the previous *tapeutil* functions (refer to [“Standard Edition - Tapeutil menu commands”](#) on page 107).

- [A] Add Device Manually

Opens the screen for specifying a device manually instead of using the Scan function.

- [P] Preferences

Opens the dialog where default program settings can be defined and altered.

## Standard Edition - Scan menu commands

When ITDT-SE is used after **S** is entered on the start screen, the Scan function starts and displays the first device list screen.

IBM Tape Diagnostic Tool Standard Edition - Device List									
	Host	Bus	ID	LUN	Model	Serial	Fware	Changer	[#]
0	2	0	8	0	ULT3580-HH6	1013000112	E4J1		
1	3	0	0	0	03592E05	00001365271	1F05		
2	3	0	1	0	ULT3580-HH5	9068030258	D8D5	78G8705_LL0	
3	3	0	2	0	ULT3580-HH6	10WT023884	E6R3	78G8705_LL0	
4	3	0	2	1	3573-TL	78G8705_LL0	A.60		
5	5	0	0	0	ULTRIUM-HH5	1068048260	D8D5		
6									
7									
8									
9									

Please select a device by entering the number in the leftmost column of the list followed by enter.

[S] Scan

[H] Help [M] Main Menu [V] Toggle View  
 <[Q] Quit | [P] Previous | [N] Next | Line # | Command >

Figure 14. Scan screen menu

IBM Tape Diagnostic Tool Standard Edition - Device List									
	Host	Bus	ID	LUN	Model	Serial	Ucode	Changer	[#]
0	4	0	6	0	ULT3580-TD4	1310000299	7381		
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									

[S] Scan      [T] Test      [D] Dump      [F] Firmware Update  
 [E] Encryption    [W] Full Write    [U] Tape Usage    [O] Other...  
 <[H] Help : [Q] Quit : Line # : Command > \_

Figure 15. Scan menu

To select a device, enter a number from the leftmost column, then click **Enter**.

Entering the **M** command returns to the **Start Screen** menu. Entering the **V** command toggles between displaying the physical device address and the driver name.

IBM Tape Diagnostic Tool Standard Edition - Device List									
	Host	Bus	ID	LUN	Model	Serial	Fware	Changer	[#]
0	2	0	8	0	ULT3580-HH6	1013000112	E4J1		X
1	3	0	0	0	03592E05	00001365271	1F05		
2	3	0	1	0	ULT3580-HH5	9068030258	D8D5	78G8705_LL0	
3	3	0	2	0	ULT3580-HH6	10WT023884	E6R3	78G8705_LL0	
4	3	0	2	1	3573-TL	78G8705_LL0	A.60		
5	5	0	0	0	ULTRIUM-HH5	1068048260	D8D5		
6									
7									
8									
9									

[S] Scan      [T] Health Test    [D] Dump      [F] Firmware Update  
 [Y] sYstem Test    [J] eJect Cartrid. [A] CleAning Stat.  
 [H] Help    [M] Main Menu    [V] Toggle View    [O] Other Functions...  
 <[Q] Quit | [P] Previous | [N] Next | Line # | Command >

Figure 16. Scan screen second menu

This screen contains the **S**, **T**, **D**, **F**, **Y**, **J**, and **A** commands. Entering the **O** command displays the second device list screen. The second screen contains the **W**, **U**, **K**, **L**, **I**, **E**, **C**, and **R** commands.

The following commands are described:

- S - [“Scan” on page 91](#)
- T - [“Health Test” on page 93](#)
- D - [“Dump” on page 95](#)
- F - [“Firmware Update” on page 96](#)
- Y - [“System Test” on page 98](#)
- J - [“Eject Cartridge” on page 100](#)
- A - [“Cleaning Statistics” on page 100](#)

- O - “Other Functions” on page 100

IBM Tape Diagnostic Tool Standard Edition - Device List

	Host	Bus	ID	LUN	Model	Serial	Ucode	Changer	[#]
0	2	0	0	0	03592E06	00001326807	2471	13F00141003	
1	2	0	0	1	3577-TL	13F00141003	0030		
2	7	0	38	0	ULT3580-HH3	1H10003495	93G7		
3									
4									
5									
6									
7									
8									
9									
10									
11									

[Y] sYstem Test [L] Library Test [I] Manual Inspect  
[R] Return [C] Config. TCP/IP  
<[H] Help : [Q] Quit : + : - : Line # : Command >

Figure 17. More scan options

IBM Tape Diagnostic Tool Standard Edition - Device List

	Host	Bus	ID	LUN	Model	Serial	Fware	Changer	[#]
0	2	0	8	0	ULT3580-HH6	1013000112	E4J1		X
1	3	0	0	0	03592E05	00001365271	1F05		
2	3	0	1	0	ULT3580-HH5	9068030258	D8D5	78G8705_LL0	
3	3	0	2	0	ULT3580-HH6	10WT023884	E6R3	78G8705_LL0	
4	3	0	2	1	3573-TL	78G8705_LL0	A.60		
5	5	0	0	0	ULTRIUM-HH5	1068048260	D8D5		
6									
7									
8									
9									

[W] Full Write [U] Tape Usage [K] Check LTFS Readiness  
[L] Library Test [I] Lib. Media Screening  
[E] Encryption [C] Config. TCP/IP  
[H] Help [M] Main Menu [V] Toggle View [R] Return  
<[Q] Quit | [P] Previous | [N] Next | Line # | Command >

Figure 18. More scan options

- W - “Full Write” on page 100
- U - “Tape Usage” on page 102
- K - “Check LTFS Readiness” on page 103
- L - “Library Test” on page 103
- I - “Library Media Screening” on page 103
- E - “Encryption” on page 104
- C - “Configure TCP/IP” on page 105
- R - “Return” on page 107

## Scan

The Scan function [S] is used to discover all supported tape and library devices that are attached to the computer system so that they can be selected for the subsequent ITDT-SE operations. The scan function also serves as a connection test that can be used to verify correct attachment of the devices.

Make sure that no other program is accessing the devices that are used by ITDT-SE. For example, stop the backup jobs that are accessing the devices when ITDT-SE is used, or if not sure, stop the entire backup application.

After ITDT-SE is started, type S followed by **Enter** to activate the scan function.

Depending on the operating system and the number of attached drives, the scan can take several minutes. See “Standard Edition - known issues and limitations” on page 85 for details.

During the scan operation, a bar in the lower left edge of the screen shows that the scan operation is still in progress.

When the scan is finished, the first device list screen is displayed.

IBM Tape Diagnostic Tool Standard Edition - Device List

	Host	Bus	ID	LUN	Model	Serial	Ucode	Changer	[#]
0	1	186	10	0	03592E05	00001365102	1E0B	1340038040B	X
1	1	186	10	1	03584L22	1340038040B	8750		
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									

[S] Scan [T] Test [D] Dump [F] Firmware Update  
 [E] Encryption [W] Full Write [U] Tape Usage [O] Other...  
 <[H] Help | [Q] Quit | Line # | Command >

Figure 19. Device List screen

IBM Tape Diagnostic Tool Standard Edition - Device List

	Host	Bus	ID	LUN	Model	Serial	Fware	Changer	[#]
0	2	0	8	0	ULT3580-HM6	1013000112	E431		X
1	3	0	0	0	03592E05	00001365271	1F05		
2	3	0	1	0	ULT3580-HM5	0068030258	0805	7868705_LL0	
3	3	0	2	0	ULT3580-HM6	10W7023884	E6R3	7868705_LL0	
4	3	0	2	1	3573-TL	7868705_LL0	A.60		
5	5	0	0	0	ULTRIUM-HM5	1068048260	0805		
6									
7									
8									
9									

[S] Scan [T] Health Test [D] Dump [F] Firmware Update  
 [V] sYstem Test [J] eject Cartrid. [A] CleAning Stat.  
 [H] Help [M] Main Menu [V] Toggle View [O] Other Functions...  
 <[Q] Quit | [P] Previous | [N] Next | Line # | Command >

Figure 20. Device List screen

The first device list screen shows all detected devices and the connection information (host adapter number, bus number, SCSI/FCP ID, and LUN or driver name) along with product data (Model name, Unit Serial number, Microcode revision). For drives that are attached to a library, the Changer column shows the serial number of the changer the drive is attached to.

Scrollable data is indicated by "VVVVVVVVV" in the bottom border of the table. To view the non-displayed entries, type + and press **Enter**.

**Note:** For fast down scrolling, type + followed by a space and the number of lines to scroll down, then press **Enter**. Alternately, type N and press **Enter** to scroll down one page.



To scroll back, use - instead of +.

**Note:** For fast up (backward) scrolling, type - followed by a space and the number of lines to scroll up. Press **Enter**, or type P and press **Enter** to scroll up one page.

If no devices appear or if devices are missing in the list, make sure that

- ITDT-SE is running with administrator/root rights.
- The devices are properly attached and powered on.
- Linux: The devices must be attached at boot time.
- i5/OS: Only tape drives are detected.
- Solaris, when no IBM tape device driver is in use:
  - Ensure that sgen is correctly configured.

```
file /kernel/drv/sgen.conf
```

is correctly configured (see [“Starting ITDT-SE on Solaris operating systems”](#) on page 82).

- Solaris 10, see (see [“Standard Edition - known issues and limitations”](#) on page 85)
- More than 10 devices displayed - scroll down the Device List.

ITDT-SE uses the IBM Tape Device Driver for its operations. If no IBM Tape Device Driver is installed, the generic device driver for the operating system is used instead. On Microsoft Windows, any Tape Device Driver that is installed is used.

If you must bypass the IBM Tape Device Driver for diagnostic purposes, start ITDT-SE with the following command.

```
itdt -force-generic-dd
```

**Note:** For operating system-specific information on how to use this command, see the corresponding Initial Startup sections.

When the wanted device is displayed, select the device for test. Only one device can be selected at a time.

## Health Test

The Health Test function [T] checks if the tape device is defective and outputs a pass/fail result.



**Attention:** The health test function erases user data on the cartridge that is used for the test.

For the library or autoloader test, the Library Test [L] must be selected.

### Note:

1. The test can take from 15 minutes up to 2 hours.
2. The test runs only on tape drives, not on autoloaders or libraries.

To complete the test function, it is recommended that a new or rarely used cartridge is used. Scaled (capacity-reduced) cartridges must not be used to test the device.

To test tape drives within a library, the library must be in online mode.

1. Start ITDT-SE, then type S and press **Enter** to scan for the devices.
2. Select the device that you want to test by entering its number and press **Enter**.
3. Type T followed by **Enter** to activate the test.

If no cartridge is inserted, ITDT-SE prompts to insert a cartridge. Either insert a cartridge and press **Enter** or stop the test by entering C followed by **Enter**.

**Note:** If ITDT-SE detects data on the cartridge, the Device Test screen displays a message (as shown in [Figure 21](#) on page 94).

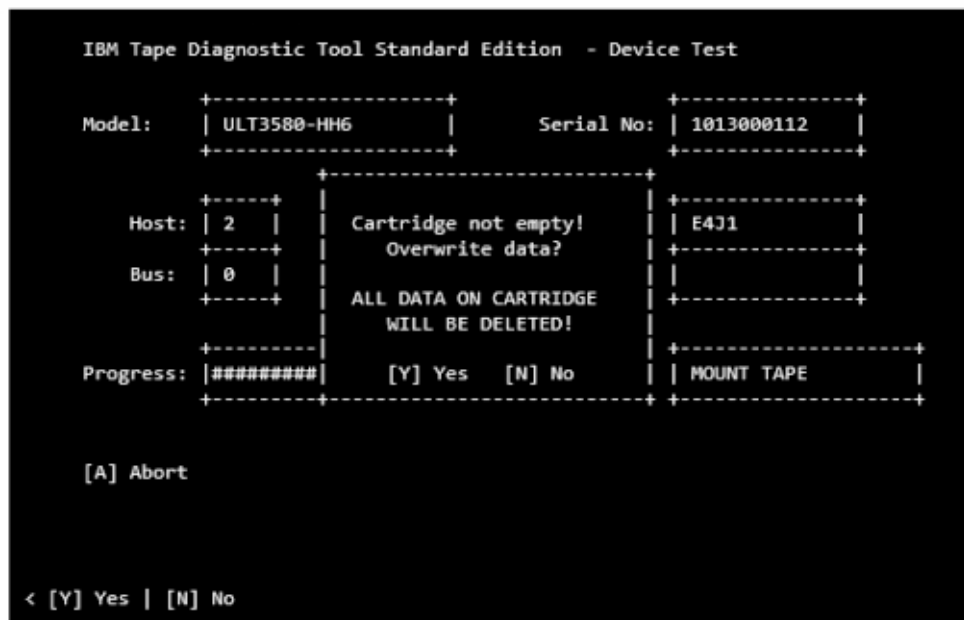


Figure 21. Data Delete question

Type Y followed by **Enter** to continue the test if you are sure that data on the cartridge can be overwritten. If you are unsure, type N followed by **Enter** to stop the test.

During the test, the program shows a progress indicator in the form of a bar of number signs (#) (1) that shows the progress of a single subtest and also a description of that subtest. The user might stop the test by selecting the [A] Abort option (exception: POST A).

During the test, a progress indicator (1) is shown on the test screen. Messages from the test steps are shown in the **Status** field (2).

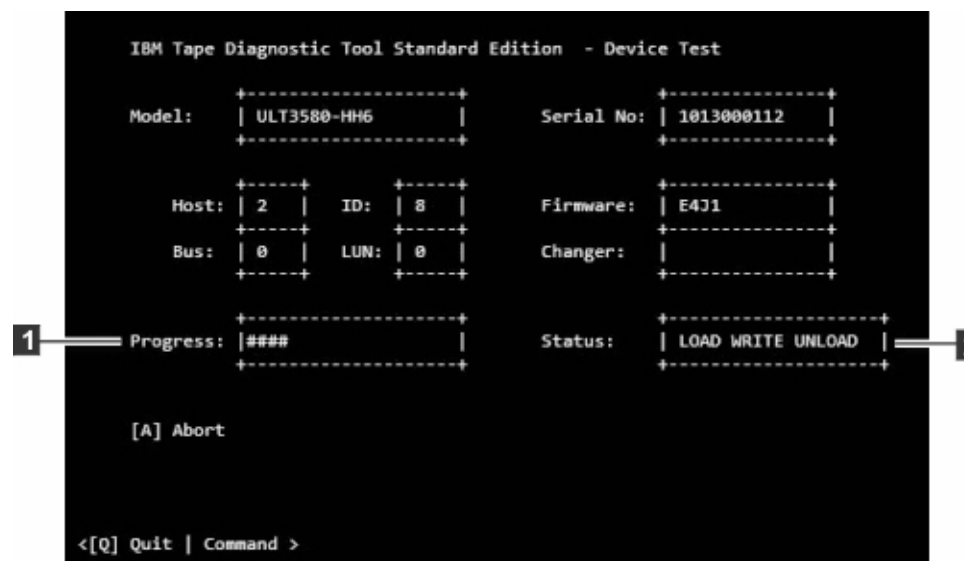


Figure 22. Test running

The test sequence contains the following steps.

1. Initialize Device
2. Read Thermal Sensor (might get skipped)
3. Mount Medium
4. [Medium Qualification] - only if the previous step indicated this requirement

5. Load/Write/Unload/Read/Verify
6. POST A
7. Performance Test (run 2 times if first run failed with performance failure)
8. Unmount Medium
9. Read Thermal Sensor (might get skipped)
10. Get FSC
11. Get Logs

The test can be stopped by typing A followed by **Enter** at any time except during the POST test, which is not interruptible.

**Note:** It might take some time until the test stops.

```

IBM Tape Diagnostic Tool Standard Edition - Device Test

Model:  +-----+      Serial No: +-----+
        | ULT3580-HH6 |          | 1013000112 |
        +-----+      +-----+
Host:   +-----+      ID:   +-----+      Firmware: +-----+
        | 2 |      | 8 |          | E4J1 |
        +-----+      +-----+
Bus:   +-----+      LUN:  +-----+      Changer:   +-----+
        | 0 |      | 0 |          |      |
        +-----+      +-----+
Result: +-----+      Code: +-----+
        | PASSED |          | OK |
        +-----+      +-----+

Bin Log: +-----+      Dumps: +-----+
        | 1013000112.blz |      |      |
        +-----+      +-----+
Text Log: +-----+      |      |
        | 1013000112.txt |      |      |
        +-----+      +-----+
[R] Return to device list

<[H] Help | [Q] Quit | Command >
  
```

Figure 23. Test results

When all subtests are finished, ITDT-SE shows a screen that displays the attachment and device information as in the first device list screen. It also shows the test result and failure information in the code field. The screen also shows the output files that were generated during the test run. The files might be requested by the IBM Support Center.

If you want to use other ITDT-SE functions, type R followed by **Enter** to return to the first device list screen. Otherwise, type Q followed by **Enter** to exit the program.

## Dump

Complete the following steps to start the Dump [D] process.

1. Start ITDT-SE, then type S and press **Enter** to scan for the devices.
2. Select the device that you want to retrieve a dump from by entering its number and pressing **Enter**.
3. Type D and press **Enter** to start the dump retrieval for the selected device. The ongoing dump process is completed (it takes less than 1 minute).

```

IBM Tape Diagnostic Tool Standard Edition - Dump

+-----+
Model: | ULT3580-HH6 | Serial No: | 1013000112 |
+-----+

+-----+
Host: | 2 | ID: | 8 | Firmware: | E4J1 |
+-----+
Bus: | 0 | LUN: | 0 | Changer: | |
+-----+

+-----+
Files: | 1013000112.002.a | Note:
      | 1013000112.002.b |   *.a - before "Force Dump"
      +-----+         *.b - after "Force Dump"
                        *.dump.blz - full data package
+-----+
Log: | 1013000112.002.dump.blz | Status: | PASSED |
+-----+

[R] Return to Device List
<[H] Help | [Q] Quit | Command >

```

Dump file names

Log file names

a2500237

Figure 24. Dump

When the dump process is completed on a tape library or autoloader other than the 3584/TS3500/TS4500, the Dump function stores 1 log file in the output folder of the program (\*.blz). For the 3584/TS3500/TS4500, a dump file (\*.a) is stored in the output folder.

**Note:** When the Dump function is completed for tape libraries or autoloaders other than the 3584/TS3500/TS4500, the log file contains only Log Sense and Mode Sense pages, while a Drive or 3584/TS3500/TS4500 dump contains much more diagnostic information.

Retrieve the files from the ITDT-SE output subdirectory that was created during the installation. The following are examples of the directory:

- Example output directory (Windows): c:\ITDT\output
- Example output directory (UNIX): /home/user/ITDT/output
- Example output directory (i5/OS): /home/ITDT/output

(On the IFS) use FTP or the System i Navigator to transfer the file

If you want to use other ITDT-SE functions, type R followed by **Enter** to return to the device list; otherwise, type Q followed by **Enter** to exit the program.

## Firmware Update

The Firmware Update [F] upgrades the firmware of tape drives and tape libraries.

```

IBM Tape Diagnostic Tool Standard Edition - Firmware Update

Host Bus ID LUN Model Serial Fware Changer
+-----+-----+-----+-----+-----+-----+
| 0 | 0 | 0 | 0 | ULT3580-TD4 | 1300001016 | 8197 | IBM0600_LL0 |
+-----+-----+-----+-----+-----+-----+

FW Dir: | /tmp/itdt/ITDT/input/ |
+-----+-----+-----+-----+-----+

Content: +-----+-----+-----+-----+-----+-----+
| 0 | IBM_3573_6_90.fbi | | | Dir OK: | true |
| 1 | df080109b_8192.FC.ro | | | +-----+
| 2 | df080109g_8197.FC.ro | | |
| 3 | IBM_3573_6_70.fbi | | |
| 4 | | | |
| 5 | | | |
| 6 | | | |
| 7 | | | |
| 8 | | | |
| 9 | | | |
+-----+-----+-----+-----+-----+

[F] FW Directory
[D] Dir Refresh
[H] Help
[R] Return
[C] Continue

[Q] Quit | + | - | [N] Next | [P] Previous | Command >

```

Figure 25. Firmware Update screen

**Note:**

The following site is available for the latest firmware files: <https://www.ibm.com/support/fixcentral>.

1. Press the button **Select Product**
2. Product Group: **System Storage**
3. **Tape systems**
4. **Tape drives or Tape auto loaders and libraries**
5. Select your product accordingly
6. Press **Continue**
7. Select the related fix and download the file(s).

Download the files to the ITDT-SE input subdirectory that was created during the installation. The following are examples of the directory:

```

Example input directory (Windows): c:\ITDT\input
Example input directory (Unix): /home/user/ITDT/input
Example input directory (i5/OS): /home/ITDT/input
(on the IFS) use FTP or the i-Series Navigator to transfer the file

```

To do a Firmware Update, complete the following steps:

1. Start ITDT-SE, then type S and press **Enter** to scan for the devices.
2. Select the device that you want to update by typing the number of the device and pressing **Enter**.
3. Type F and press **Enter** to display the **Firmware Update** screen.
4. To select the needed firmware update, complete one of the following steps:
  - If the downloaded firmware file is listed in the **Content** field of the **Firmware Update** screen, type the corresponding line number and press **Enter**.
  - If the firmware file is stored in a directory other than **FW Dir**, type F followed by a space and the fully qualified path to the directory that contains the firmware file, then press **Enter**.

For example, enter the following to change the firmware directory (UNIX):

```
f /home/user/firmware
```

- If no files are displayed in the **Content** field, check the **Dir OK** field on the right side of the screen. It indicates true if the directory exists, false otherwise.
- If the content of the displayed **FW Dir** changed, type D and press **Enter** to refresh the directory content.

**Note:** The selected file name is reset to the first item (#0) after the Refresh function is used.

- If the displayed directory contains more files than the files shown, type + and press **Enter** to scroll down the list. For fast down scrolling type + followed by a space and the number of lines to scroll down then press **Enter**. To scroll back, use - instead of +.

Scrollable data is indicated by "VVVVVVVVVVVVVVVVVV".

IBM Tape Diagnostic Tool Standard Edition - Firmware Update

Host	Bus	ID	LUN	Model	Serial	Fw	Changer
0	0	0	0	ULT3580-TD4	1300001016	8197	IBM0600_LL0

FW Dir: /tmp/Firmware/

Content:

0	df070308a_7381.SAS.ro
1	df070417d_74H4.FC.ro
2	df080319o_83KF.FC.ro
3	df070308a_7381.FC.ro
4	df070417d_74H4.ro
5	df070308a_7381.ro
6	df070308d_7384.FC.ro
7	df070308d_7384.ro
8	df070417d_74H4.SAS.ro
9	df080402p_842G.FC.ro

Dir OK: true

[F] FW Directory  
[D] Dir Refresh  
[H] Help  
[R] Return  
[C] Continue

[Q] Quit | + | - | [N] Next | [P] Previous | Command >

VVVV = more data

Figure 26. Scrollable Data screen

5. After the firmware file is selected, type C and press **Enter** to continue.
6. Before the firmware update is started, make sure the file that is displayed in the **FW File** field is the correct file.
  - If the correct file is displayed, proceed to the next step.
  - If the correct file is not displayed, type C and press **Enter** to change the selected firmware file. Go to Step 4.

**Note:** The selected file name is reset to the first item in the list when you return to that dialog from the Start Update dialog.

7. If you decide to run the firmware update, type S and press **Enter** to start the firmware update.

During the firmware update, a firmware update progress screen is displayed.



**Attention:** Once started, do not interrupt the firmware update.

The firmware update usually takes 3-5 minutes, but it can take up to 45 minutes for libraries. If you decide not to run the firmware update, type R and press **Enter** to return to the Device List.

**Note:** If ITDT-SE detects FIPS-certified drive firmware, it displays a warning dialog. Before you continue, ensure that you use FIPS-certified firmware to update the drive.

8. After completion, the **Status** field on the lower right side indicates PASSED if the firmware was updated successfully and FAILED otherwise.

Type R and press **Enter** to return to the Device List.

## System Test

The System Test [Y] is a short test that completes the following steps:

- Reveals system performance bottlenecks. Compressible data throughput values can reveal bandwidth limitations that are caused by the system, cabling, or HBA.
- Measures performance variations across the different block sizes to find the ideal block size for the system configuration.

The System Test runs only on tape drives, not on autoloaders or libraries. To complete a System Test on tape drives within a library, the library must be in online mode.

1. Start ITDT-SE, type S, and press **Enter** to scan for the devices.
2. Type Y and press **Enter** to start the System Test.

ITDT-SE then switches to the System Test screen. If no cartridge is inserted, ITDT-SE prompts to insert a cartridge. Either insert a cartridge and press **Enter** or stop the test by typing C followed by **Enter**.

**Note:** If ITDT-SE detects data on the cartridge, it shows the System Test screen, and displays the following message.

```
Cartridge not empty!
Overwrite data?
```

Type Y followed by **Enter** to continue the test if you are sure that data on the cartridge can be overwritten. If you are unsure, type N followed by **Enter** to stop the test.

The System Test is completed as follows:

- a. System Test determines the amount of data to write for each supported blocksize (a percentage of the cartridge is written for each blocksize)
- b. The test determines the maximum supported blocksize of the system – maximum is 8 MiB (generic 1 MiB).
- c. Next, the System Test writes the calculated size with supported block sizes in powers of two down to 64 kB (at maximum 8192, 4096, 2048, 1024, 512, 256, 128, 64), first with incompressible data, then with compressible data. Five different block sizes are written.
- d. At the end of the test, a summary screen is displayed.

IBM Tape Diagnostic Tool Standard Edition - System Test					
Model	Serial		Fware		Result:
ULT3580-TD4	F09789A000		89B2		PASSED
Compressible	Transfer Size (KB)	Data Size (MB)	Elapsed Time (s)	Data Rate (MB/s)	Code:
No	512	953	12.684	75.1734	OK
No	256	953	13.245	71.9894	
No	128	953	13.26	71.9174	
No	64	953	13.256	71.9391	
Yes	512	953	4.501	211.842	Text Log:
Yes	256	953	4.922	193.722	1013000112.004.txt
Yes	128	953	6.279	151.875	
Yes	64	953	9.002	105.935	Bin Log:
					1013000112.004.blz

Figure 27. System Test results

"Compressible = Yes" means that the data written was just zeros so that the data is compressed by the drive with a maximum compression ratio. "Compressible = No" means that a data pattern was written that the drive almost cannot compress at all. If the compression ratio is 1, the drive was not able to compress the data (equivalent to 1:1 compression ratio). If the compression ratio is 94.0, the drive

was able to do 94:1 compression, meaning that 94 bytes in the original data is compressed to 1 byte on the medium. 100.0 means 100 bytes is compressed down to 1 byte on the medium.

The System Test can be stopped by typing A followed by **Enter** at any time.

**Note:** It can take some time until the System Test stops.

If you want to use other ITDT-SE functions, type R followed by **Enter** to return to the device list. Otherwise, press Q followed by **Enter** to exit the program.

### ***Eject Cartridge***

The Eject Cartridge [J] function unloads a cartridge from a tape drive.

1. Start ITDT-SE, then type S and press **Enter** to scan for the devices.
2. Select the device that you want to unload a cartridge from by entering its number and pressing **Enter**.
3. Type J and press **Enter** to unload a cartridge.

### ***Cleaning Statistics***

Cleaning Statistics [A] retrieves statistical data about cleaning actions. Some devices do not support this function.

1. Start ITDT-SE, type S, and press **Enter** to scan for the devices.
2. Select the device that you want to retrieve cleaning statistics from by entering its number and pressing **Enter**.
3. Type A and press **Enter** to start the cleaning statistics retrieval for the selected device.

### ***Other Functions***

Other Functions [O] - type O followed by **Enter** to display a screen with the following commands:

- W - [“Full Write” on page 100](#)
- U - [“Tape Usage” on page 102](#)
- K - [“Check LTFS Readiness” on page 103](#)
- L - [“Library Test” on page 103](#)
- I - [“Library Media Screening” on page 103](#)
- E - [“Encryption” on page 104](#)
- C - [“Configure TCP/IP” on page 105](#)
- R - [“Return” on page 107](#)

### ***Full Write***

The Full Write [W] function writes the entire cartridge with a specified block size either with compressible or incompressible data and output performance data.



**Attention:** The Full Write function erases data on the cartridge that is used for the test.

#### **Note:**

1. The Full Write function takes approximately 2 hours when incompressible data is written, less time for compressible data.
2. The Full Write function runs only on tape drives, not on autoloaders or libraries.

The Full Write test can be used to

- Demonstrate that the drive can write the full amount of data on a cartridge.
- Identify system issues with compression.

Drive data compression is always turned on during the full write. When run with compressible data, the output shows the compression rate. If the compression rate is higher than 1.0 but the system does not



appear to be able to compress data on the cartridge, check the device driver and software settings to see whether they disable compression.

1. After ITDT-SE is started, type S followed by **Enter** to activate the device scan.
2. Select the device that you want to write to by entering its number and press **Enter**.
3. Type W and press **Enter** to start the full write.

ITDT-SE then switches to the Full Write screen. If no cartridge is inserted, ITDT-SE prompts to insert a cartridge. Either insert a cartridge and press **Enter** or stop the test by typing C followed by **Enter**.

**Note:** If ITDT-SE detects data on the cartridge, it shows the Full Write screen, and displays the following message:

```
Cartridge not empty!  
Overwrite data?
```

Type Y followed by **Enter** to continue the test if you are sure that data on the cartridge can be overwritten. If you are unsure, type N followed by **Enter** to stop the test.

4. The system prompts for entry of a transfer size between 32 KB and the maximum block size that is supported by the system (maximum value is 8 MiB). This action is a check for the type of supported block size that is completed. Enter the appropriate values for your system.

**Note:** Values of 16 KB and 32 KB are not tested in cases where the capability of a system supports higher block sizes.

5. Select the type of data to write, either **[C] Compressible** or **[I] Incompressible**.

During the full write, the program shows a progress indicator in form of a bar of number signs (#) that shows the progress of the full write.

The full write can be stopped by typing A followed by **Enter** at any time.

**Note:** It can take some time until the full write stops.

If all write operations are finished, ITDT-SE shows a screen that displays the compression ratio (1) and the write performance (shown in 2 as the Data Rate) for the selected block size. If an error occurred during the full write, data is only written partially.

"Compressible = Yes" means that the data written was just zeros so that the data is compressed by the drive with a maximum compression ratio. "Compressible = No" means that a data pattern was written that the drive almost cannot compress at all. If the compression ratio is 1, the drive was not able to compress the data (equivalent to 1:1 compression ratio). If the compression ratio is 94.0, the drive was able to do 94:1 compression, meaning that 94 bytes in the original data is compressed to 1 byte on the medium. 100.0 means 100 bytes is compressed down to 1 byte on the medium.

If you want to use other ITDT-SE functions, type R followed by **Enter** to return to option 4 the device list. Otherwise, type Q followed by **Enter** to exit the program.

IBM Tape Diagnostic Tool Standard Edition - Full Write Result

Host	Bus	ID	LUN	Model	Serial	Firmware	Changer
5	0	1	0	ULT3580-TD4	F09789A000	89B2	1300213_LLC

Compressible	Transfer Size (KB)	Data Size (MB)	Elapsed Time (s)	Data Rate (MB/s)	Comp. Ratio
No	256	405411	5319.36	76.2144	0.99

Result:

PASSED	Bin Log:	F09789A000.005.dump.blz
Code:	Text Log:	F09789A000.004.txt
OK	Dumps:	F09789A000.003.a
		F09789A000.003.b

[R] Return to device list

<[Q] Quit | [R] Return>

a2500243

Figure 28. Full Write results

## Tape Usage

The Tape Usage [U] function retrieves statistical data and error counters from a cartridge.

IBM Tape Diagnostic Tool Standard Edition - Tape Usage

Model	Serial No.
ULT3580-TD4	1300000206

Tape Usage:

Thread Count	585	Data Sets Read	104461
Data Sets Written	2546209	Read Retries	6
Write Retries	22	Unrecovered Read Err.	0
Unrecovered Write Err.	0	Suspended Reads	0
Suspended Writes	12	Fatal Suspended Reads	0
Fatal Suspended Writes	0		

Code:	OK	Result:	PASSED
-------	----	---------	--------

[R] Return to Device List

<[H] Help | [Q] Quit | Command >

a2500244

Figure 29. Tape Usage screen

1. After ITDT-SE is started, type S followed by **Enter** to activate the device scan.
2. Select the device that you want to test by entering its number and press **Enter**.
3. Type U followed by **Enter** to start the tape usage log retrieval. ITDT-SE then switches to the tape usage screen. If no cartridge is inserted, ITDT-SE prompts to insert a cartridge. Either insert a cartridge and press **Enter** or stop the test by entering C followed by **Enter**.

During the get logs operation, the program shows a progress indicator in form of a bar of number signs (#) that shows the progress of a single suboperation and a description of that operation.

When all suboperations are finished, ITDT-SE shows a **Tape Usage** completion screen. The **Status** field on the lower right side indicates PASSED if the log retrieval completed successfully and ABORTED otherwise.

## Check LTFS Readiness

The Check LTFS Readiness test analyzes the operating system and tape drive environment to ensure that the IBM linear tape file system can be installed. This test checks the operating system version, the tape device driver version, the tape drive firmware, and the LTFS HBA requirements.

**Note:** The tape drive firmware must be at least version: C7RC (for LTO 5), C974 (for LTO 6), and 36A5 (for TS1140 and TS1150). The LTFS Readiness Check requires an empty data cartridge. The maximum transfer size is detected at the beginning of the test. If it is under 512kB, a warning is displayed. If it is under 256kB, the test fails.

The LTFS Readiness Check can return with result FAILED and one of the following error codes.

Table 12. Codes and root causes	
Code	Root causes
LTFS CDB LENGTH	Unsupported SCSI 16 Byte command.
LTFS CARTRIDGE TYPE	Cartridge Density Code is not supported.
LTFS XFER SIZE	Supported block size is too less.
LTFS PARTITION SUPPORT	Partitioning commands are not supported.
LTFS FW LEVEL	Unsupported Tape Device firmware level.
LTFS DATA TRANSFER SIZE	Data Integrity Test failed.
LTFS DD VERSION	IBM Tape Device Driver version is not supported.
LTFS OPERATING SYSTEM	Operating system is not supported.

## Library Test

The Library Test [L] starts and monitors the library-internal self-test. This test runs only on libraries and autoloaders, not on tape drives.

1. Start ITDT-SE, type S, and press **Enter** to scan for the devices.
2. Type 0 and press **Enter** to display the second device list screen.
3. On the second device list screen, type L and press **Enter** to start the Library Test.

A Device Test screen is displayed and a functionality test on the tape library is completed.

At the end of the test, a results screen is displayed.

The Library Test can be stopped by typing A followed by **Enter** at any time.

**Note:** It can take some time until the Library Test stops.

If you want to use other ITDT-SE functions, type R followed by **Enter** to return to the device list; otherwise press Q followed by **Enter** to exit the program.

## Library Media Screening

Library Media Screening [L] generates dumps for each drive and cartridge within a library. It runs only on libraries (except TS3500/TS4500) and auto-loaders, not on tape drives.

First, the test tries to read dump files from each drive that is installed from the library. After that, the customer must select one drive for loading the cartridges.

All cartridges of the I/O and storage slots are moved - one after the other - from their source to the selected drive. A dump is taken and moved back to the source address.

In the result screen, the dumps taken and the count of dumps are displayed.

## Encryption

The Encryption [E] function is used to verify whether data on the cartridge was written encrypted. It reads both decrypted and raw data from the cartridge into two separate files on disk. The user can then verify that the data differs to ensure that encryption worked.

The Encryption function does not provide a Write - Read test.

The Encryption function is supported only on encryption enabled drives. It requires that an encryption infrastructure, including the Encryption Key Manager (EKM), is properly set up. An encrypted data cartridge must be used.

The Encryption function is supported for the following encryption environments:

- System Managed: IBM tape device driver must be installed and in use by ITDT to read decrypted data
- Library Managed
- Application Managed: Only raw encrypted data is read (result file \*.ENC)

**Note:** On i5/OS, media changers and media changer operations are not supported by this release of ITDT-SE. To test a tape drive inside a library, the tape drive must be varied online and the tape library must be varied offline (see [“Starting ITDT-SE on i5/OS operating systems”](#) on page 84 for details). As the library is varied offline, the Encryption function does not deliver decrypted data in a Library Managed Encryption environment.

1. After ITDT-SE is started, type S followed by **Enter** to activate the device scan.
2. Select the device that you want to test by entering its number and press **Enter**.
3. Type E and press **Enter** to start the encryption test. ITDT-SE then switches to the **Encryption Verification** screen. On this screen, the system requires the entry of the number of the start record and the amount of data (in KB) to be read.
4. Type S followed by a space and the start record number, then press **Enter** to enter the start record number. Type L followed by a blank and the data length, then press **Enter** to enter the data length, maximum 100000 KB.

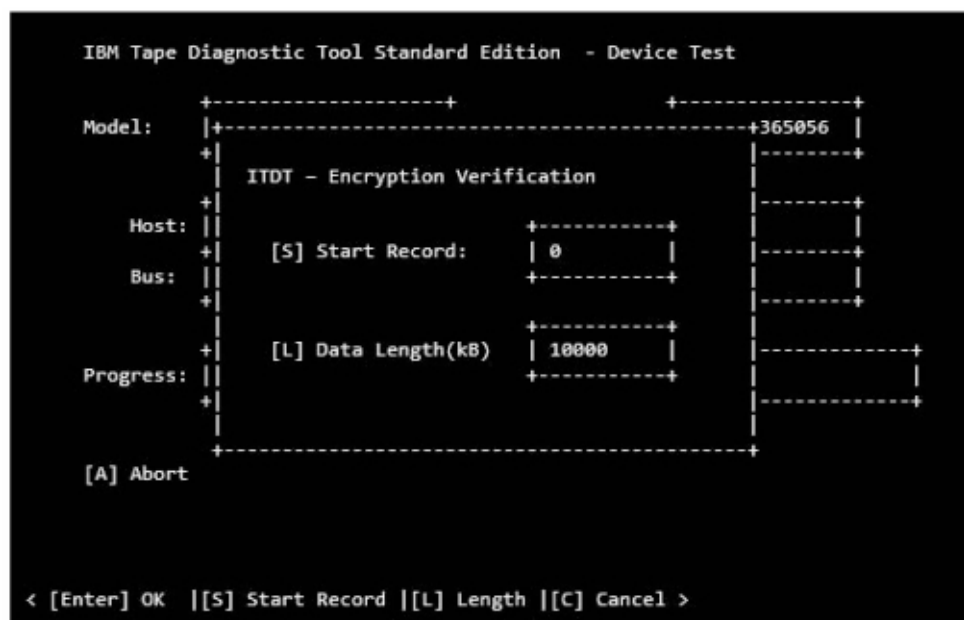


Figure 30. Encryption Start screen

5. If you entered the values correctly, press **Enter** to start the encryption.

During the encryption, the program shows a progress indicator in form of a bar of number signs (#) that shows the progress of a single subtest and information about that subtest.

The Encryption function can be stopped by typing A followed by **Enter** at any time.

**Note:** It can take some time before the Encryption function stops.

If all encryption operations are finished, ITDT-SE shows a screen that displays the **Status** field on the lower left side that indicates PASSED if the encrypted test completed successfully and ABORTED otherwise.

The screen also shows the output files that were generated during the Encryption function:

- file *serial# .n.ENC* contains the raw encrypted data
- file *serial# .n.DEC* contains the decrypted data

Table 13 on page 105 defines the abort codes.

Table 13. Abort code definitions	
ABORT CODE	ROOT CAUSE
LOCATE FAILED	Start position as requested by the user was not reached
MEDIUM NOT ENCRYPTED	ITDT detected medium as non-encrypted
NO DRIVER SPECIAL FILE	System-Managed environment, but generic device file is used instead of IBM device driver special file
DRIVE ENCRYPTION DISABLED	Mode Sense detected disabled drive encryption
UNEXPECTED DATA	<ul style="list-style-type: none"><li>• Set Raw read mode failed</li><li>• One of the commands failed</li></ul>
END OF MEDIUM	End of medium that is encountered before the specified amount of data was read
END OF DATA	End of data that is encountered before the specified amount of data was read
READ FAILED	
ENCRYPTION ERROR	
INVALID PARAMETER	User entered data length of 0 kB
FILE IO ERROR	The hard drive that ITDT is installed on might have run out of space.

If you want to use other ITDT-SE functions, type R followed by **Enter** to return to the device list. Otherwise, type Q followed by **Enter** to exit the program.

## Configure TCP/IP

Configure TCP/IP [C] configures the Ethernet settings of LTO 5, TS1140, and later drives. For those drives, the current settings are read and displayed and can be changed.

**Note:** LTO drives have one port with 2 sockets and TS1140 and later drives have two ports and 4 sockets can be configured. Configuring the Ethernet must not be done in a TS3500/TS4500. Although the ports can be configured, it is ineffective.

1. Start ITDT-SE, type S, and press **Enter** to scan for the devices.
2. Select a device from the list (just the ones that are listed are supported) by entering the number, then press **Enter**.
3. Type 0 and press **Enter** to display the second device list screen.
4. On the second device list screen, type C and press **Enter** to open the **TCP IP** screen.

ITDT-SE switches to the **TCP IP** screen, reads the data configuration and displays it in a table.

```

IBM Tape Diagnostic Tool Standard Edition - TCP IP

Host Bus ID LUN Model Serial Fware Device
+-----+-----+-----+-----+-----+-----+
| 1 | 0 | 4 | 0 | ULT3580-TD9 | 1013000376 | R8MF |
+-----+-----+-----+-----+-----+-----+

Port Sock DHCP IP Address Mask [#]
+-----+-----+-----+-----+-----+-----+
| 0 | 1 | 1 | 0 | 1.2.3.4 | 23 | |
| 1 | 1 | 2 | 0 | | 0 | X|
| 2 | | | | | | |
| 3 | | | | | | |
+-----+-----+-----+-----+-----+-----+

[S] Set [C] Cancel

[H] Help [M] Main Menu
<[Q] Quit [P] Previous [N] Next Line # Command >

```

Figure 31. TCP/IP screen: Query configuration

Select the Port/Socket that you want to alter by entering its number ( 0..1 for LTO5+ and 0..3 for TS1140+) and press **Enter**.

To alter the values for the selected Port/Socket type S and press **Enter**. The current values are loaded.

```

ITDT- Configure TCP/IP

Host Bus ID LUN Model Serial Fware Device
+-----+-----+-----+-----+-----+-----+
| 1 | 0 | 4 | 0 | ULT3580-TD9 | 1013000376 | R8MF |
+-----+-----+-----+-----+-----+-----+

Port          +-----+
              | 1 |
+-----+
Socket        | 2 |
+-----+
[1] DHCP      | 0 |
+-----+
[2] IP Address |
+-----+
[3] Subnet Mask Length | 0 |
+-----+

[A] Apply
<[R] Return [Q] Quit Command > [2] 1111:2222:3333:4444:5555:6666:7777:ABCD

```

Figure 32. TCP/IP screen: Read data

Each parameter can be set by entering the number (1 - 3) and a following value.

If you want to enable DHCP, enter 1 1 and press **Enter**. The value of the **DHCP** field is refreshed with the value entered.

To change the IP address, enter 2 and the fixed IP Address you want to set followed by pressing **Enter**. The value of the **IP Address** field is refreshed with the value entered.

For the subnet mask length, enter 3 and the Subnet Mask Length you want to set followed by pressing **Enter**. The value of the **Subnet Mask Length** field is refreshed with the value entered.

```

ITDT- Configure TCP/IP

Host Bus ID LUN Model Serial Fware Device
+-----+-----+-----+-----+-----+-----+
| 1 | 0 | 4 | 0 | ULT3580-TD9 | 1013000376 | R8MF |
+-----+-----+-----+-----+-----+-----+

Port          +-----+
              | 1 |
+-----+
Socket        +-----+
              | 2 |
+-----+
[1] DHCP      +-----+
              | 0 |
+-----+
[2] IP Address +-----+
              | 1111:2222:3333:4444:5555:6666:7777:ABCD |
+-----+
[3] Subnet Mask Length +-----+
              | 0 |
+-----+

[A] Apply
<[R] Return | [Q] Quit | Command >

```

Figure 33. TCP/IP screen: Altered data

Regular field values:

```

[1] DHCP enabled:      0/1 (false/true)
[2] Address IPV4:      Regular IPv4 address
    Address IPV6:      Regular IPv6 address
[3] Subnet Mask Length V4: 0...23
    Subnet Mask Length V6: 0...127

```

The values are applied to the drive by entering A and pressing **Enter**. ITDT-SE configures the drive and the current addresses are shown.

```

IBM Tape Diagnostic Tool Standard Edition - TCP IP

Host Bus ID LUN Model Serial Fware Device
+-----+-----+-----+-----+-----+-----+
| 1 | 0 | 4 | 0 | ULT3580-TD9 | 1013000376 | R8MF |
+-----+-----+-----+-----+-----+-----+

Port Sock DHCP IP Address Mask [#]
+-----+-----+-----+-----+-----+-----+
| 0 | 1 | 1 | 0 | 1.2.3.4 | 23 | [X] |
| 1 | 1 | 2 | 0 | 1111:2222:3333:4444:5555:6666:7777:ABCD | 127 | |
| 2 | | | | | | |
| 3 | | | | | | |
+-----+-----+-----+-----+-----+-----+

[S] Set          [C] Cancel

[H] Help  [M] Main Menu
<[Q] Quit | [P] Previous | [N] Next | Line # | Command >

```

Figure 34. TCP/IP screen: Applied values

If you want to use other ITDT-SE functions, type R, then press **Enter** to return to the device list. Or, press Q and **Enter** to exit the program.

## Return

Return [R] - type R followed by **Enter** to go back to the first device list screen.

## Standard Edition - Tapeutil menu commands

When the user runs the **U** command on the ITDT-SE start screen, the Tapeutil operation screen is displayed.

**Note:** On any screen, to start a command, press the shortkey displayed in brackets [ ], followed by **Enter**.

The following commands are described in this section:

### [1] Open a Device

When you select the **Open a Device** command [1]:

1. ITDT checks if a device is already opened.
2. You are prompted for a device special file name.
3. You are prompted for an open mode (rw, ro, wo, append).
4. ITDT opens the device that you selected.

**Note:** Always use the Read Only mode when you are working with write-protected media.

The combination of open cmd with parameter **-force-generic-dd** is not supported.

### Device file names - device addressing

ITDT supports generic and Device Driver claimed devices. This section shows examples for device names (addressing) of all supported platforms. The used abbreviations stand for:

- host**      Number of the host adapter (SCSI, FC, SAS)
- bus**        Number of the bus from the host adapter
- target**     Target Number of the device
- lun**        Logical Unit Number of the device that is separated with blanks.

**Note:** The correct IDs are reported in the ITDT Control Center after a scan or with the scripting function "scan".

Table 14. Device addressing			
	IBM Tape Device Driver	Generic IDs separated with blanks	Generic (alternative, as a result from ".itdt scan")
<b>IBM AIX</b>	/dev/rmtX.Y /dev/smcX	<host><bus><target><lun>	H<host>-B<bus>-T<target>-L<lun>
<b>Linux</b>	/dev/IBMtapeX /dev/IBMchangerX		
<b>Microsoft Windows</b>	\\.\tape0 \\.\changer0		
<b>Oracle Solaris</b>	/dev/rmtXsmc /dev/smc/Xchng		
<b>HP-UX</b>	/dev/rmt/Xmnb /dev/rmt/Xchng		
<b>Apple Mac</b>	-	<host><bus><target><lun>	H<host>-B<bus>-T<target>-L<lun> tapeX changerX



Table 14. Device addressing (continued)			
	IBM Tape Device Driver	Generic IDs separated with blanks	Generic (alternative, as a result from ".itdt scan")
IBM 'i'	-	-	Device Name; for example, TAP01

Example for Linux (IBM Tape Driver devices):

```
./itdt scan
Scanning SCSI Bus ...
#0 /dev/IBMchanger2 - [3573-TL]-[C.50] S/N:00L4U78D6118_LL0 H3-B0-T5-L1 (IBM-Device)
#1 /dev/IBMtape3 - [ULT3580-TD5]-[BBNE] S/N:1168001104 H3-B0-T5-L0
Changer: 00L4U78D6118_LL0 (IBM-Device)
Exit with code: 0
```

Example for Linux (generic Operating System devices):

```
./itdt -force-generic-dd scan
Scanning SCSI Bus ...
#0 /dev/sg7 - [3573-TL]-[C.50] S/N:00L4U78D6118_LL0 H3-B0-T2-L1 (Generic-Device)
#1 /dev/sg8 - [ULT3580-TD5]-[BBNE] S/N:1168001104 H3-B0-T5-L0
Changer: 00L4U78D6118_LL0 (Generic-Device)
Exit with code: 0
```

To open the tape device by using the IBM Tape Device Driver, use the **/dev/IBMtape3** as device file name. By using **H3-B0-T5-L0** or **3 0 5 0** as the device name, it is opened with the generic Operating System device driver.

## [2] Close a Device

When you select the **Close a Device** command [2]:

1. ITDT checks if the device is already closed.
2. ITDT closes the device.

## [3] Inquiry

When you select the **Inquiry** command [3]:

1. You are prompted for page code.
2. ITDT then displays a decoded format of a hexadecimal dump and prints a hexadecimal dump of the inquiry data.

The equivalent ITDT scripting command is **inquiry**

## [4] Test Unit Ready

When you select the **Test Unit Ready** (TUR) command [4], ITDT issues the Test Unit Ready command to the device.

The equivalent ITDT scripting command is **tur**

## [5] Reserve Device

When you select the **Reserve Device** command [5], ITDT issues a reserve command to the device.

The equivalent ITDT scripting command is **reserve**

## [6] Release Device

When you select the **Release Device** command [6], ITDT issues a release command to the device.

The equivalent ITDT scripting command is **release**

### **[7] Request Sense**

When you select the **Request Sense** command [7]:

1. ITDT issues a **Request Sense** command.
2. ITDT then displays a decoded format of hexadecimal dump sense data and prints hexadecimal dump sense data.

The equivalent ITDT scripting command is [requestsense](#)

### **[8] Log Sense**

When you select the **Log Sense** command [8]:

1. You are prompted for Log Sense Page.
2. ITDT issues a log sense command.
3. ITDT completes a hexadecimal dump page.

The equivalent ITDT scripting command is [logsense](#)

### **[9] Mode Sense**

When you select the **Mode Sense** command [9]:

1. You are prompted for Mode Sense Page.
2. ITDT issues mode sense command.
3. ITDT completes a hexadecimal dump page.

The equivalent ITDT scripting command is [modesense](#)

### **[10] Query Driver Ver. (Version)**

When you select the **Query Driver Version** command [10]:

1. ITDT issues the required command to print IBM Tape Device device driver
2. ITDT prints the driver version.

The equivalent ITDT scripting command is [qryvers](#)

### **[11] Display All Paths**

When you select the **Display All Paths** command [11]:

All configured SCSI paths for this device are queried. Such as logical parent, SCSI IDs, and the status of the SCSI paths for the primary path and all alternative paths that are configured.

The equivalent ITDT scripting command is [qrypath](#)

**Note:** The command is only supported for devices that are using the IBM Tape Device Driver.

### **[12] Query Runtime Info**

When you select the **Query Runtime Info** command [12]:

1. ITDT issues the required command to get the runtime info.
2. ITDT prints out the Dynamic Runtime Attribute Values.

The equivalent ITDT scripting command is [runtimeinfo](#)

**Note:** The command is only supported for tape devices.

### **[20] Rewind**

When you select the **Rewind** command [20], ITDT rewinds a loaded cartridge.

The equivalent ITDT scripting command is [rewind](#)

**Note:** The command is only supported for tape devices.

### **[21] Forward Space File Marks**

When you select the **Forward Space File Marks** command [21]:

1. You are prompted for the number of file marks to space forward.
2. ITDT is spacing the number of specified file marks in the forward direction.

The equivalent ITDT scripting command is **fsf**

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[22] Backward Space File Marks**

When you select the **Backward Space File Marks** command [22]:

1. You are prompted for the number of file marks.
2. ITDT is spacing the number of specified file marks in the backward direction.

The equivalent ITDT scripting command is **bsf**

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[23] Forward Space Records**

When you select the **Forward Space Records** command [23]:

1. You are prompted for the number of records to space forward.
2. ITDT is spacing the number of specified data records in the forward direction.

The equivalent ITDT scripting command is **fsr**

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[24] Backward Space Records**

When you select the **Backward Space Records** command [24]:

1. You are prompted for the number of records to space backward.
2. ITDT is spacing the number of specified data records in the backward direction.

The equivalent ITDT scripting command is **bsr**

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[25] Space to End of Data**

When you select the **Space to End of Data** (EOD) command [25], ITDT is spacing from the current position to the end of data.

The equivalent ITDT scripting command is **seod**

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[26] Read and Write Tests**

When you select the **Read and Write Tests** command [26]:

- You are prompted for block size (If you press **Enter**, the default block size is 10240 bytes).

**Note:** If the block size is zero, variable mode is used. With a fixed block size, a data amount of (block size \* blocks) is transferred with a single operation. This process can get rejected if the total amount exceeds the transfer size the system can handle.

- You are prompted for the number of blocks per read/write (If you press **Enter**, the default number of blocks is 20).

- You are prompted for the number of repetitions (If you press **Enter**, the default number of repetitions is 1).

You can then select one of the following options:

- Read data from tape (to run Read-only test)
- Write data to tape (to run Write-only test)
- Write/Read/Verify (to run Read and Write test)

ITDT runs the selected test. Then, it displays the transfer size and block size that is used for this test, the number of records read/written, and the total bytes transferred.

The equivalent ITDT scripting command is **rwtest**



**CAUTION:** The command is overwriting the content of the loaded cartridge! Any data is lost.

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[27] Read or Write Files**

When you select the **Read or Write Files** command [27]:

- You are prompted to specify the file name of the source file (for Write test) or the destination file (for Read test).
- You are prompted for the number of records to be read.

You can then select one of the following options:

- Read File from Tape: ITDT reads a file from tape and stores data into a file with the specified file name.
- Write File to Tape: ITDT reads data from file with the specified file name and writes data to tape.

ITDT displays the number of records read/written, the transfer size, and the total bytes transferred.

The equivalent ITDT scripting commands are **write** and **read**



**CAUTION:** The command **write** is overwriting the content of the loaded cartridge! Any data is lost.

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[28] Erase**

When you select the **Erase** command [28], ITDT erases the loaded cartridge. All data on cartridge will be overwritten!

For more information, refer to the ITDT Scripting Command at “erase” on page 129.



**CAUTION:** The command is overwriting the content of the loaded cartridge! Any data is lost.

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[29] Load Tape**

When you select the **Load Tape** command [29], ITDT issues the load tape command.

The equivalent ITDT scripting command is **load**

**Note:** The command is only supported for tape devices.

### **[30] Unload Tape**

When you select the **Unload Tape** command [30], ITDT issues the unload tape command.

The equivalent ITDT scripting command is **bsr**

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[31] Write File Marks**

When you select the **Write File Marks** command [31]:

1. You are prompted for the number of file marks to write.
2. ITDT writes the number of file marks to the current position.

The equivalent ITDT scripting command is **weof**



**CAUTION:** The command is overwriting the content of the loaded cartridge! Any data is lost.

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[32] Synchronize Buffers**

When you select the **Synchronize Buffers** command [32], ITDT flushes the buffer to the cartridge.

The equivalent ITDT scripting command is [sync](#)



**CAUTION:** The command is overwriting the content of the loaded cartridge! Any data is lost.

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[33] Query/Set Parameter**

When you select the **Query/Set Parameter** command [33]:

1. ITDT displays non-changeable parameters.

**Note:** The list of non-changeable parameters is operating system specific.

2. ITDT displays changeable parameters.

**Note:** The list of changeable parameters is operating system specific. For a list of changeable parameters, refer to [Table 15 on page 136](#).

3. You are prompted for parameter to change.
4. ITDT requests prompt for parameter value (if required).
5. ITDT requests safety prompt (if required).
6. ITDT issues the *ioctl* command.

The equivalent ITDT scripting command querying the parameters is [getparms](#)

The equivalent ITDT scripting command setting a parameter is [setparm](#)

### **[34] Query/Set Tape Position**

When you select the **Query/Set Tape Position** command [34]:

1. ITDT displays the current position
2. You are prompted for a new position to set.
3. This subcommand issues the SCSI Locate command to the device to set the tape position.

The equivalent ITDT scripting command querying the parameters is [setpos](#)

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[35] Query Encryption Status**

When you select the **Query Encryption Status** command [35]:

ITDT queries the tape drive encryption settings and displays the encryption state.

The equivalent ITDT scripting command is [qryencryption](#)

**Note:** The command is only supported for tape devices.

### **[36] Display Message**

When you select the **Display Message** command [36]:

1. You are prompted for the text of message 0 (8 characters or less).
2. You are prompted for the text of message 1 (8 characters or less).
3. You are prompted for message type (msg0, msg1, flash0, flash1, alt).
4. ITDT displays a message on the display panel.

Not all drives have a display. The 3592 drive is the only one that has display message capability. It is the only one with a display that is more than one character long. Eight is the limit of the characters on a display screen.

The equivalent ITDT scripting command is [display](#)

**Note:** The command is only supported for tape devices.

### **[37] Report Density Supp (Support)**

When you select the **Report Density Support** command [37]:

1. ITDT prints report status text for all supported media.
2. ITDT issues a report density command to retrieve all supported media.
3. ITDT prints all requested reports. Data is printed in a decoded way. Scroll the screen to print each one of the following information:
  - Density name
  - Assigning organization
  - Description
  - Primary density code
  - Secondary density code
  - Write OK
  - Duplicate
  - Default
  - Bits per MM
  - Media Width
  - Tracks
  - Capacity (megabytes).
4. ITDT prints report status text for current media
5. ITDT issues a report density command to retrieve current media
6. ITDT prints report data in a decoded way.

The equivalent ITDT scripting command is [density](#)

**Note:** The command is only supported for tape devices and requires a loaded cartridge.

### **[38] Test Encryp. Path (Test Encryption Key Path/Setup)**

When you select the **Test Encryption Key Path/Setup** command [38]:

**Note:** Not supported for the HP-UX operating system.

1. ITDT prints status message that server configuration and connections are tested
2. ITDT issues the **Encryption Diagnostics** *ioctl* command, **Ping Diag**
3. ITDT prints number of servers available or error message
4. ITDT issues the **Encryption Diagnostics** *ioctl* command, **Basic Encryption Diag**

5. ITDT prints completion code or error message
6. ITDT issues the **Encryption Diagnostics** *ioctl* command, **Full Encryption Diag**
7. ITDT prints completion code or error message.

The equivalent ITDT scripting command is [ekmtest](#)

### **[50] Element Information**

When you select the **Element Information** command [50]:

1. ITDT issues the **mode sense** command
2. ITDT displays the
  - Number of transport elements
  - First transport element address
  - Number of storage slots
  - First storage slot address
  - Number of I/O elements
  - First I/O element address
  - Number of drives
  - First drive address

The equivalent ITDT scripting command is [elementinfo](#)

**Note:** The command is only supported for tape library devices.

### **[51] Position to Element**

When you select the **Position to Element** command [51]:

1. You are prompted for destination address
2. This subcommand issues the SCSI Position to Element command by using the destination address specified.

The equivalent ITDT scripting command is [position](#)

**Note:** The command is only supported for tape library devices.

### **[52] Element Inventory**

When you select the **Element Inventory** command [52]:

This subcommand issues the SCSI Read Element Status command for each element type and displays the element status information

**Note:** ITDT displays decoded element inventory information. Type n followed by **Return** to show the next page of information.

The equivalent ITDT scripting command is [inventory](#)

**Note:** The command is only supported for tape library devices.

### **[53] Exchange Medium**

When you select the **Exchange Medium** command [53]:

1. You are prompted for source address.
2. You are prompted for first destination address.
3. You are prompted for second destination address.
4. ITDT issues the 2 SCSI **MOVE** commands to exchange the cartridges.

The equivalent ITDT scripting command is [exchange](#)

**Note:** The command is only supported for tape library devices.

### **[54] Move Medium**

When you select the **Move Medium** command [54]:

1. You are prompted for source address.
2. You are prompted for destination address.
3. ITDT issues the **MOVE** command to move the cartridge from *source address* to *destination address*.

The equivalent ITDT scripting command is [move](#)

**Note:** The command is only supported for tape library devices.

### **[55] Initialize Element Status**

When you select the **Initialize Element Status** command [55]:

1. ITDT prints the command summary.
2. ITDT issues the SCSI **INITIALIZE ELEMENT STATUS** command.

The equivalent ITDT scripting command is [audit](#)

**Note:** The command is only supported for tape library devices.

### **[56] Prevent/Allow Medium Removal**

When you select the **Prevent/Allow Medium Removal** command [56]:

1. You are prompted to select **(1)** for Prevent Removal, or **(0)** for Allow Removal.

The equivalent ITDT scripting command is [prevent](#) and [allow](#)

**Note:** The command is only supported for tape library devices.

### **[57] Initialize Element Status Range**

When you select the **Initialize Element Status Range** command [57]:

1. You are prompted for the first slot address.
2. You are prompted for the number of slots.
3. TDT issues the SCSI **INITIALIZE ELEMENT STATUS WITH RANGE** command.

The equivalent ITDT scripting command is [audit](#)

**Note:** The command is only supported for tape library devices.

### **[58] Read Device IDs**

When you select the **Read Device IDs** command [58], ITDT retrieves the device ID information for all available drives and displays the information. Type n followed by **Enter** to show the next page.

The equivalent ITDT scripting command is [devid](#)

**Note:** The command is only supported for tape library devices.

### **[59] Read Cartridge Location**

When you select the **Read Cartridge Location** command [59]:

1. You are prompted for the address of the first slot.
2. You are prompted for number of elements.
3. ITDT verifies that the specified address range is valid, otherwise prints error message and exit.
4. ITDT issues the **Read Cartridge Location** *ioctl* command.
5. ITDT issues the **Element Info** *ioctl* command.



6. ITDT verifies that the address range is valid; otherwise, print the error message and exit.
7. If no slots are found in Element Info data, print the error message and exit.
8. ITDT issues the **Read Cartridge Location** *ioctl* command.
9. ITDT prints decoded storage element information, Type n followed by **Enter** to show next page.

The equivalent ITDT scripting command is cartridgelocation

**Note:** The command is only supported for tape library devices.

### **[70] Dump/Force Dump/Dump**

When you select the **Dump/Force Dump/Dump** command [70]:

- ITDT retrieves the dump.
- ITDT issues the **Force Dump** command.
- ITDT retrieves second dump.
- ITDT displays the name of stored dump files and the output directory where they are stored. The dump file names start with the serial number of the device.

The equivalent ITDT scripting command is dump

**Note:** The command is only supported for tape devices.

### **[71] Firmware Update**

**Note:**

The following site is available for the latest firmware files: <https://www.ibm.com/support/fixcentral>.

1. Press the button **Select Product**
2. Product Group: **System Storage**
3. **Tape systems**
4. **Tape drives or Tape auto loaders and libraries**
5. Select your product accordingly
6. Press **Continue**
7. Select the related fix and download the file(s).

When you select the **Firmware Update** command [71]:

1. ITDT displays the firmware update dialog:
  - If the downloaded firmware file is listed in the **Content** field of the **Firmware Update** screen, type the corresponding line number and press **Enter**.
  - If the firmware file is stored in a directory other than **FW Dir**, type F followed by a space and the fully qualified path to the directory that contains the firmware file, then press **Enter**.

For example, enter the following to change the firmware directory (UNIX):

```
f /home/user/firmware
```

- If no files are displayed in the **Content** field, check the **Dir OK** field on the right side of the screen. It indicates true if the directory exists, false otherwise.
- If the content of the displayed **FW Dir** changed, type D and press **Enter** to refresh the directory content.

**Note:** The selected file name is reset to the first item (#0) after the Refresh function is used.

- If the displayed directory contains more files than the files shown, type + and press **Enter** to scroll down the list. For fast down scrolling type + followed by a space and the number of lines to scroll down then press **Enter**. To scroll back, use - instead of +.

Scrollable data is indicated by "VVVVVVVVVVVVVVVVVV".

2. After the firmware file is selected, type C and press **Enter** to continue.
3. Before the firmware update is started, make sure the file that is displayed in the **FW File** field is the correct file.
  - If the correct file is displayed, proceed to the next step.
  - If the correct file is not displayed, type C and press **Enter** to change the selected firmware file. Go to Step 4.

**Note:** The selected file name is reset to the first item in the list when you return to that dialog from the Start Update dialog.

4. If you decide to run the firmware update, type S and press **Enter** to start the firmware update.

During the firmware update, a firmware update progress screen is displayed.



**Attention:** Once started, do not interrupt the firmware update.

The firmware update usually takes 3-5 minutes, but it can take up to 45 minutes for libraries. If you decide not to run the firmware update, type R and press **Enter** to return to the Device List.

**Note:** If ITDT-SE detects FIPS-certified drive firmware, it displays a warning dialog. Before you continue, ensure that you use FIPS-certified firmware to update the drive.

5. After completion, the **Status** field on the lower right side indicates PASSED if the firmware was updated successfully and FAILED otherwise.

Type R and press **Enter** to return to the Device List.

## Add device manually

On the **Start** screen, press the [A] key followed by **Enter** to add a drive and a changer manually. In a dialog box, the tape device and the associated changer device can be defined. Supported formats are: **devicename** and **H B T L**. For a standalone tape drive the changer device is not required.

A default tape device is set and stepping through the device names the values can be altered.

When **Enter** is pressed, those drives which are accessible are opened. The manually added devices are shown in the same way as they are shown after an ITDT **Scan** command.

## Standard Edition - Program options

For problem determination and customization, ITDT is providing the following command line options.

-h -help	Prints help information
-version	Displays the version of ITDT, used configuration files and creation dates.
-force-generic-dd	the usage of the generic Operating System driver (not using the IBM Tape Device driver) will be forced.
-LP logpath	Use 'logpath' as logging path (default: output)
-L logfile	Use 'logfile' for log messages (default: metro.log)
-LL Errors Warnings Information Debug	Set log level (default: Error)
-R resultdir	Use 'resultdir' as result file path (default: output)
-settings	Change default values of ITDT configuration parameters.
-start deviceaddress -end deviceaddress	Set range for device scan function, devicename and H B T L is supported.

## Standard Edition - Tapeutil scripting commands

Scripting is enabled with the 4.0 release of ITDT SE. ITDT-SE provides compatibility with earlier versions for existing tapeutil scripts. While some legacy commands exist, they are not documented in their entirety as they are phased out over time. New scripts must always use the scripts that are listed in this guide, the Common Command set (CS). Also, existing scripts must be modified for forward compatibility with ITDT.

You can find a list of commands, on each command you find the command, a description, parameter list, and which platforms are supported. Some commands have numbers after them. The numbers mean that a corresponding menu command is in [“Standard Edition - Tapeutil menu commands”](#) on page 107.

The following are the generic invocation parameters as in use by the legacy command sets: **-f filename**

**Note:** "filename" is a device special file for the drive/changer, for example:

```
/dev/rmt0 (AIX), /dev/rmt/0st (Solaris), /dev/rmt/0m (HP-UX),  
/dev/IBMtape0 (Linux), \\.\tape0 (Windows)
```

The calling convention for the Common Command set is

```
itdt -f filename [Open Mode] Subcommand [Subcommand ...]
```

**Note:** "filename" is a device special file for the drive/changer or the device address (host bus target lun). For a complete list of the file name or address syntax, refer the section "Special Files" on each platform or go to [“Device file names - device addressing”](#) on page 108.

The calling convention for the Common Command set is

```
itdt -f filename [Open Mode] Subcommand [Subcommand ...]
```

The Open Mode flag is supported on all platforms. If the flag is not set, the device is opened in read/write mode. More parameters that might be required for opening the device are automatically detected and set.

```
-w mode  
Open mode, by default Read/Write.  
Valid modes are:  
1 = Read/Write  
2 = Read Only  
3 = Write Only  
4 = Append
```

**Note:** Always use the Read Only mode when you are working with write-protected media.

The new command set enables legacy commands on every platform, even if that is not previously supported by Tapeutil. The output follows current Tapeutil conventions. But, if different output displays for a single command on various platforms, the output is implemented according to the AIX output as the primary scripting platform.

Tapeutil allows undocumented abbreviations for some of the commands. For example, it was possible to shorten “inquiry” to “inq” or “inqu” or “inqui”. The following command abbreviations are supported by ITDT-SE too: inq(uiery), req(sense), pos(ition), ele(mentinfo), inv(entory), devid(s), cartridge(location). Deprecated commands are listed at [“Deprecated commands”](#) on page 148. Also, there is a list of unsupported commands and known exceptions at [“Standard Edition scripting commands: known limitations and deviations”](#) on page 150.

The following commands are described.

General commands

- [“allow”](#) on page 122
- [“devinfo”](#) on page 122
- [“inquiry | inq | inqj”](#) on page 122
- [“logpage | logpagej”](#) on page 123

- [“loop” on page 123](#)
- [“modepage” on page 124](#)
- [“prevent” on page 124](#)
- [“print” on page 124](#)
- [“qrypath” on page 124](#)
- [“qryversion” on page 125](#)
- [“release” on page 125](#)
- [“reqsense” on page 125](#)
- [“reserve” on page 125](#)
- [“scan|scanj” on page 125](#)
- [“sleep” on page 127](#)
- [“tur” on page 127](#)
- [“vpd” on page 128](#)

#### Tape commands

- [“append” on page 128](#)
- [“bsf” on page 128](#)
- [“bsr” on page 128](#)
- [“channelcalibration” on page 128](#)
- [“chgpart” on page 128](#)
- [“density” on page 129](#)
- [“display” on page 129](#)
- [“erase” on page 129](#)
- [“formattape” on page 129](#)
- [“fdp” on page 130](#)
- [“fdpl” on page 130](#)
- [“fsf” on page 130](#)
- [“fsr” on page 130](#)
- [“getparms” on page 130](#)
- [“idp” on page 130](#)
- [“idpl” on page 131](#)
- [“list” on page 131](#)
- [“load” on page 132](#)
- [“logsense” on page 132](#)
- [“qrypar | qrypart” on page 132](#)
- [“qrylbp” on page 132](#)
- [“qrypos” on page 133](#)
- [“qrytcip” on page 133](#)
- [“read” on page 133](#)
- [“resetdrive” on page 134](#)
- [“rmp” on page 134](#)
- [“runtimeinfo | qryruntimeinfo” on page 134](#)
- [“rewind” on page 134](#)
- [“rtest” on page 134](#)

- [“rwtest” on page 135](#)
- [“sdp” on page 135](#)
- [“sdpl” on page 135](#)
- [“seod” on page 135](#)
- [“setparm” on page 136](#)
- [“setpos” on page 137](#)
- [“settcpip” on page 137](#)
- [“sync” on page 138](#)
- [“unload” on page 138](#)
- [“verlbp” on page 138](#)
- [“weof” on page 138](#)
- [“write” on page 138](#)
- [“wtest” on page 139](#)

#### Medium Changer Subcommands

- [“audit” on page 139](#)
- [“cartridgelocation” on page 139](#)
- [“elementinfo” on page 139](#)
- [“exchange” on page 139](#)
- [“inventory” on page 140](#)
- [“librarymediaoptimization” on page 140](#)
- [“move” on page 141](#)
- [“position” on page 141](#)

#### Service Aid commands

- [“dump” on page 141](#)
- [“ekmtest” on page 141](#)
- [“encryption” on page 141](#)
- [“ucode” on page 141](#)
- [“tapephcp” on page 142](#)
- [“ltfsphcp” on page 142](#)
- [“verify” on page 143](#)
- [“devicestatistics” on page 143](#)
- [“checkltfsreadiness” on page 143](#)
- [“ltfsdefragmentation” on page 143](#)
- [“standardtest” on page 144](#)
- [“fullwrite” on page 144](#)
- [“systemtest” on page 144](#)
- [“tapeusage” on page 145](#)

#### HD-P commands

- [#unique\\_320](#)
- [#unique\\_321](#)
- [#unique\\_322](#)

#### Library RESTful commands

- [“TS4500 REST over SCSI” on page 145](#)
- [“TS4300: RESTful API” on page 146](#)

## ***allow***

(Deprecated: **unlock**, **-o rem**) Allow medium removal for tape or changer devices (unlock door). The counter command for this is prevent.

Parameters:

None

Supported platforms: All

## ***devinfo***

(Deprecated: **-o gdi**) Show device information (device type, sub type and block size)

Parameters:

None

Supported platforms: AIX, Solaris, HP-UX

## ***inquiry / inq / inqj***

**inquiry** or **inq** command identifies the device and returns the output in hexadecimal format. **inqj** command is similar to **inquiry** or **inq** command but generates the output in JSON format.

Parameters:

[Page code in Hex, 00-FF without leading x]

Following is an example of an output from **inquiry** or **inq** command:

```
[root@franc ITDT]# ./itdt -f 4 0 8 0 inq
Issuing std inquiry...
Inquiry Data, Length 96
      0 1  2 3  4 5  6 7  8 9  A B  C D  E F    0123456789ABCDEF
0000 - 0180 0602 3301 1002 4942 4D20 2020 2020 [...]3...IBM      ]
0010 - 3033 3539 3245 3037 2020 2020 2020 2020 [03592E07      ]
0020 - 3343 3834 4A50 3030 3030 3133 4430 3031 [3C84JP000013D001]
0030 - 3233 2031 8100 1181 0113 0008 0000 0000 [23 1.....]
0040 - 0080 0000 0114 0004 0000 0002 0123 0008 [.....#..]
0050 - 0000 0000 0040 0000 0124 0004 0000 0001 [.....@...$......]
Exit with code: 0
```

Following is an example of an output from **inqj** command:

```
[root@franc ITDT]# ./itdt -f 4 0 8 0 inqj
{
  "command": "inquiry" ,
  "parameter": {
    "pageCode": 0 ,
    "evpd": false
  } ,
  "length": 96 ,
  "data": {
    "version": "06" ,
    "vendorIdentification": "IBM" ,
    "productIdentification": "03592E07" ,
    "productRevisionLevel": "3C84"
  }
}
```

Supported platforms: All

## **logpage / logpagej**

(Deprecated: **-o log**) This subcommand issues the **SCSI Log Sense** command to the device for the specified page and displays the log sense data. The generated output is in hexadecimal format. **logpagej** command is similar to **logpage** command but generates the output in JSON format.

Parameters:

```
Page Logpage (in hex without leading x)
[Subpage] Subpage (in hex without leading x)
```

Following is an example of an output from **logpage** command:

```
[root@franc ITDT]# ./itdt -f 4 0 8 0 logpage 2
Issuing log sense for page 0x02...
Log Sense Page 0x02, Length 30
      0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
0000 - 0200 001A 0002 6002 0000 0003 6002 0000 [.....`.....`....]
0010 - 0005 6004 0000 0000 0006 6002 0000 [..`.....`.... ]
Exit with code: 0
```

Following is an example of an output from **logpagej** command:

```
[root@franc ITDT]# ./itdt -f 4 0 8 0 logpagej 2
{
  "command": "log sense" ,
  "parameter": {
    "pageCode": 2 ,
    "subPage": 0
  } ,
  "length": 255 ,
  "data": {
    "subPage": "00" ,
    "pageLength": 26 ,
    "values": [
      {
        "paramCode": "Not supported" ,
        "paramId": "0002" ,
        "paramValue": "0"
      } ,
      {
        "paramCode": "Total Corrected Write Errors" ,
        "paramId": "0003" ,
        "paramValue": "0"
      } ,
      {
        "paramCode": "Total Write Kibibytes Processed" ,
        "paramId": "0005" ,
        "paramValue": "0"
      } ,
      {
        "paramCode": "Total Uncorrected Write Errors" ,
        "paramId": "0006" ,
        "paramValue": "0"
      }
    ]
  }
}
```

Supported platforms: All

## **loop**

This subcommand loops all subsequent subcommands continuously or a number of times if the **Count** parameter is specified. Also refer to the sleep subcommand.

Parameters:

```
loop [Count]
```

Supported platforms: All

## ***modepage***

(Deprecated: **-o mod**) This subcommand issues the SCSI Mode Sense command to the device for the specified page and displays the mode sense data.

Parameters:

```
modepage Page (page in hex without leading x)
```

Supported platforms: All

## ***prevent***

(Deprecated: **-o lck, lock**) Prevent medium removal for tape or changer devices (lock door). The counter command for this command is **allow**.

Parameters:

```
None
```

Supported platforms: All

## ***print***

This subcommand prints the associated text to standard output. It can be used at any time to display the progress of the subcommands.

Parameters:

```
print Text
```

Supported platforms: All

## ***qryhba | qryhbainfo***

The `qryhba` | `qryhbainfo` command issues a set of unique properties like HBA vendor, HBA type, and driver version for the current host bus adapter. The number of properties displayed depends on the operating system and HBA vendor.

Parameters:

```
None
```

**Supported platforms:** All , except IBM System i

## ***qrypath***

This subcommand displays information about the device and SCSI paths, such as logical parent, SCSI IDs, and the status of the SCSI paths for the primary path and all alternate paths that are configured.

Parameters:

```
None
```

Supported platforms: All

**Note:** only supported for devices that are using the IBM Tape Device Driver.

Deprecated: **-o phs, path, checkpath**

Example:



```
$sudo ./itdt -f /dev/IBMtape1n qrypath
Querying SCSI paths...
Total paths configured..... 1

Primary Path Number 1
Logical Device..... IBMtape6
Host Bus Adapter..... qla2xxx
SCSI Host ID..... 1
SCSI Channel..... 0
Target ID..... 5
Logical Unit..... 0
Path Enabled..... Yes
Path Manually Disabled..... No
```

**Note:** ITDT shows the entire path information for all the commands.

## ***qryversion***

(Deprecated: **-o drv**) This subcommand prints out the current version of the IBM Tape Device device driver.

Parameters:

None

Supported platforms: All

## ***release***

(Deprecated: **-o rel**) This subcommand explicitly releases a device and makes it available for other hosts by issuing the SCSI Release command.

Parameters:

None

Supported platforms: All

## ***reqsense***

(Deprecated: **-o req**) This subcommand issues the SCSI Request Sense command to the device and displays the sense data in hex format.

Parameters:

None

Supported platforms: All

## ***reserve***

(Deprecated: **-o res**) This subcommand explicitly reserves a device by issuing the SCSI Reserve command.

Parameters:

None

Supported platforms: All

## ***scan/scanj***

scan command scans the system for connected tape and changer devices, and displays the list of connected devices. For each detected device following information will be reported. scanj command is similar to scan command but generates the output in JSON format.

- Special file name
- Vendor ID
- Firmware version
- Serial-number
- SCSI bus address
- Associated changer
- Device driver name

Parameters:

```
[-force-generic-dd]
[-o formatstring]
[-start=device]
[-end=device]
[-exclude=device]
[-showallpaths]
```

Following is an example of an output from **scan** command:

```
[root@franc ITDT]# ./itdt scan
Scanning SCSI Bus ...
#0 /dev/IBMtape7n - [03592E07]-[3C84] S/N:000013D00123 H4-B0-T8-L0 (IBM-Device)
#1 /dev/IBMtape6n - [03592E07]-[3C84] S/N:000013D00123 H4-B0-T7-L0 (IBM-Device)
#2 /dev/IBMtape5n - [03592E05]-[1F05] S/N:000001365271 H4-B0-T6-L0 (IBM-Device)
Exit with code: 0
```

Following is an example of an output from **scanj** command:

```
[root@franc ITDT]# ./itdt scanj
[
  {
    "index": 0 ,
    "driverName": "/dev/IBMtape7n" ,
    "modelName": "03592E07" ,
    "firmwareVersion": "3C84" ,
    "serialNumber": "000013D00123" ,
    "hostAdapterId": 4 ,
    "busNumber": 0 ,
    "targetId": 8 ,
    "lunId": 0 ,
    "deviceSpecialFile": "IBM"
  } ,
  {
    "index": 1 ,
    "driverName": "/dev/IBMtape6n" ,
    "modelName": "03592E07" ,
    "firmwareVersion": "3C84" ,
    "serialNumber": "000013D00123" ,
    "hostAdapterId": 4 ,
    "busNumber": 0 ,
    "targetId": 7 ,
    "lunId": 0 ,
    "deviceSpecialFile": "IBM"
  } ,
  {
    "index": 2 ,
    "driverName": "/dev/IBMtape5n" ,
    "modelName": "03592E05" ,
    "firmwareVersion": "1F05" ,
    "serialNumber": "000001365271" ,
    "hostAdapterId": 4 ,
    "busNumber": 0 ,
    "targetId": 6 ,
    "lunId": 0 ,
    "deviceSpecialFile": "IBM"
  }
]
```

By using the optional parameter **-force-generic-dd**, the usage of the generic Operating System driver (not with the IBM Tape Device Driver) is forced.

With the parameters **-start**, **-end**, and **-exclude**, the scan can be limited to only a particular HBA or target ID range (partial scan). It starts at **-start=device** to search for supported devices until **-end=device** is reached. Devices that are specified with **-exclude** are skipped during the scan. Multiple excluded devices must be separated by colons. Supported formats are **devicename** and **H B T L**.

The format string controls the output and specifies how the connected devices must be reported. It can include any alphanumeric character. The default format string is `"#%# %D - [%P] - [%F] S/N:%S H%H-B%B-T%T-L%L"`.

The following list contains all the interpreted identifiers:

```
%D device name
%V vendor name
%P product name
%F firmware version
%S serial number
%H host adapter
%B bus number
%T target id
%L logical unit
%h host adapter as hexadecimal value
%b bus number as hexadecimal value
%t target id as hexadecimal value
%l logical unit as hexadecimal value
%I interface type
%# enumeration number
%C serial number of associated changer device
```

Any combination of the identifiers that are listed here are supported.

An integer that is placed between a % sign and the format command acts as a minimum field width specifier. A negative value uses right text alignment.

Following is an example of **scan** command with format string:

```
./itdt scan -o "%-2# %-18S %-12P %F %I"
Scanning SCSI Bus ...
0.      1310115166   ULT3580-TD4  97F2 Fibre
1.      1310034311   ULT3580-TD4  A232 Fibre
2.      1168001104   ULT3580-TD5  B170 Fibre
3.      00L4U78D6118_LL0      3573-TL  9.20 N/A
4.      00L2U78G8705_LL0      3573-TL  9.20 N/A
5.      000001327208      03592E06  26CA Fibre
6.      cigen22164     ULT3580-TD2  73V1 Fibre
7.      000001365271      03592E05  1DD1 Fibre
```

Exit with code: 0

The option **-showallpaths** forces ITDT to display all available data paths that are detected during the device scan. The default device serial number-based filtering is not performed.

Supported platforms: All

## **sleep**

Sleep for the specified number of seconds before running the next subcommand.

Parameters:

```
sleep [Seconds]
```

Supported platforms: All

## **tur**

(Deprecated: **-o tur**) This subcommand issues the SCSI Test Unit Ready command to the device.

Parameters:

None

Supported platforms: All

### ***vpd***

This subcommand displays Vital Product Data (VPD) that are part of the Inquiry command data and outputs Manufacturer, Product Identification and Revision Level.

Parameters:

None

Supported platforms: All

### ***append***

Opens the device in append mode. The file access permission is Write Only.

Parameters:

None

Supported platforms: All, but on Windows this open mode is not supported by the IBM Tape Device Driver. On HP-UX this open mode is remapped to `r/w` by the IBM Tape Device Driver.

### ***bsf***

(Deprecated: **-o bsf**) This subcommand backward spaces Count filemarks. The tape is positioned on the beginning of the last block of the previous file. An optional Count can be specified. The default is 1.

Parameters:

`bsf [Count]`

Supported platforms: All

### ***bsr***

(Deprecated: **-o bsr**) This subcommand backward spaces Count records. An optional count can be specified. The default is 1.

Parameters:

`bsr [Count]`

Supported platforms: All

### ***channelcalibration***

This subcommand issues a self diagnostic command for drive servo channel calibration. The channel calibration returns PASSED or FAILED regarding the result of the command.

Parameters:

None

Supported platforms: All

### ***chgpart***

This subcommand changes the current active tape partition to a new partition specified by Number. Optionally, a Blockid can also be specified. If Blockid is omitted, the tape is positioned at the start of the new partition. Otherwise, the tape is positioned at the Blockid specified.

Parameters:

```
chgpart Number [Blockid]
```

Supported platforms: All

### ***density***

(Deprecated: **-o gdn / -o rds**) This subcommand issues the SCSI Report Density Support command for all supported media and for the current media loaded in the drive, and displays the results. If the drive is not loaded, the current media density is not reported.

Parameters:

```
None
```

**Note:** ITDT-SE outputs detailed information on all platforms.

Supported platforms: All

### ***display***

(Deprecated: **-o msg**) This subcommand displays a message on the display panel of the tape device. Up to 16 characters can be used for the message. If the message is longer than eight characters, the display alternates between the first eight characters and the remainder of the message.

Parameters:

```
display "message1" "message2"
```

Supported platforms: All

### ***erase***

(Deprecated: **-o era**) This subcommand writes EOD at the current position and erases the tape from EOD to the end of current partition.

Parameters:

```
[-short]
```

Supported platforms: All except i5/OS operating system

**Note:** The **erase** command triggers a long erase of the cartridge that sets EOD to the current position. Then, it writes the Data Set Separator (DSS) pattern from the new EOD to the end of the current partition. This process overwrites any data that is on the cartridge after the current logical position. To remove the entire cartridge, the user must remove all partitions (use the **rmpt** command for LTO 5, TS1140, and newer drives). Then, issue the **rewind** command before the **erase** command.

Examples:

For LTO 5 / TS1140 and later

```
itdt -f <device name> load rmp rewind erase
```

For all earlier LTO and Enterprise drive generations

```
itdt -f <device name> load rewind erase
```

### ***formattape***

This command formats the media on the device.

Parameters:

```
[-immed] This parameter immediately returns the status of the device.  
[-verify] This parameter helps the device to run a verification after media is formatted.  
[-mode] Format type
```

Supported platforms: All

### ***fdp***

This subcommand creates fdp (fixed data partitions) wrap wise. The default size for LTO 5 of partition 0 in this case is 1425 GB and the size of partition 1 is 37.5 GB. It also works for TS1140 but the size depends on the used cartridge. Supported by LTO 5, TS1140, and later.

Parameters:

None

Supported platforms: All

### ***fdpl***

This subcommand creates fdp (fixed data partitions) longitudinal. The command is valid only for TS1140 and later drives and creates partitions 0 and 1 on the cartridge. The size depends on the used cartridge.

Parameters:

None

Supported platforms: All

### ***fsf***

(Deprecated: **-o fsf**) This subcommand forward spaces count filemarks. The tape is positioned on the first block of the next file. An optional count can be specified. The default is 1.

Parameters:

fsf [Count]

Supported platforms: All

### ***fsr***

(Deprecated: **-o fsr**) This subcommand forward spaces count records. An optional count can be specified. The default is 1.

Parameters:

fsr [Count]

Supported platforms: All

### ***getparms***

(Deprecated: **-o parms / status / -o gpa**) Get and show drive, media and driver parameters.

Parameters:

None

Supported platforms: All

### ***idp***

This subcommand creates initiator defined partitions (IDP) wrap wise on tape. The parameter pSize0 is used to specify the size of partition 0 and the parameter pSize1 is used to specify the size of partition 1.

One of pSize0 or pSize1 must have a value that is entered in hex matching 37.5 \* n with (1 <= n <= 38) to specify the wanted size of that partition. The other parameter of pSize0 or pSize1 must have the value 0xFFFF to specify that the remaining capacity is used for that partition. If 0xFFFF is not used for one of the parameters, pSize0 or pSize1, the drive might reject the command | unless pSize0 and pSize1 exactly match predefined allowable values.

For TS1140 and later drives (not LTO), the parameters pSize2 and pSize3 are valid and they follow the same rules as described earlier.

For example: If you want a 37.5 GB partition (the minimum size partition) in partition 0 and the remainder in partition 1, then set pSize 0 to 0x26 and pSize1 to 0xFFFF. This action results in a volume with a 37.5 GB sized partition 0 and a 1425 GB sized partition 1.

Parameters:

```
idp pSize0 pSize1
pSize0: size of partition 0
pSize1: size of partition 1
```

Example Call:

```
idp 0x26 0xffff
```

Supported platforms: All

### ***idpl***

This subcommand creates initiator defined partitions (IDP) longitudinal wise on tape. The parameter pSize0 is used to specify the size of partition 0 and the parameter pSize1 is used to specify the size of partition 1. One of pSize0 or pSize1 must have a value that is entered in hex matching 37.5 \* n with (1 <= n <= 38) to specify the wanted size of that partition. The other parameter of pSize0 or pSize1 must have the value 0xFFFF to specify that the remaining capacity is used for that partition. If 0xFFFF is not used for one of the parameters, pSize0 or pSize1, the drive might reject the command | unless pSize0 and pSize1 exactly match predefined allowable values.

For TS1140 and later drives (not LTO), the parameters pSize2 and pSize3 are valid and they follow the same rules as described earlier.

For example: If you want a 37.5 GB partition (the minimum size partition) in partition 0 and the remainder in partition 1, then set pSize 0 to 0x26 and pSize1 to 0xFFFF. This action results in a volume with a 37.5 GB sized partition 0 and a 1425 GB sized partition 1.

Parameters:

```
idp pSize0 pSize1
pSize0: size of partition 0
pSize1: size of partition 1
```

Example Call:

```
idp 0x26 0xffff
```

Supported platforms: All

### ***list***

This subcommand displays the content of a tape. The output lists filemarks and the size of each record found on the tape until the end of data is reached. The output that is generated from this subcommand can be large depending on the amount of data on the tape and must usually be redirected to a file.

Parameters:

```
list [-start blockid] [-count recordcount]
```

Supported platforms: All

## ***load***

(Deprecated: **-o lod**) This subcommand issues a **SCSI Load** command to load a tape. The subcommand loops all subsequent subcommands continuously or a number of times if the **Count** parameter is specified. Also refer to the sleep subcommand.

Parameters:

```
load [-amu 0|1]
```

amu enables or disables archive mode unthread during **Load** command.

Supported platforms: All

## ***logsense***

Retrieves all Log Sense pages and outputs them as hex.

Parameters:

```
None
```

Supported platforms: All

## ***qrypar / qrypart***

Queries and displays tape partitioning information.

Parameters:

```
None
```

Supported platforms: All

## ***qrylbp***

Queries and displays logical block protection information.

Parameters:

```
None
```

Supported platforms: All

## ***qrymon / qrymediaoptimizationneeded***

Issuing the qrymon command prompts ITDT to run a query on the media auxiliary memory (MAM) of the cartridge and check whether media optimization is needed.

Parameters:

```
None
```

Supported platforms: LTO-9 only

## ***qrymov / qrymediaoptimizationversion***

Issuing the qrymov command prompts ITDT to run a query on the media auxiliary memory (MAM) of the cartridge and display the version of media optimization.

Parameters:

```
None
```

Supported platforms: LTO-9 only



## ***qrypos***

(Deprecated: **-o gpo**) This subcommand displays the current tape position.

Parameters:

None

Supported platforms: All

## ***qrytcpip***

This subcommand outputs the current drive TCP/IP configuration. Only supported with LTO 5, TS1140, and later drives. Outputs adapter and TCP/IP address information for IPv4 and IPv6 with address, port and subnet mask. For example:

```
./itdt -f /dev/IBMtape3 qrytcpip
Reading current TCP/IP Configuration...
Number of Port Descriptors 1
Port Descriptor for Port 1
Number of Socket Descriptors 2
Adapter:1 IPV4 9.11.22.111/23 DHCP:0
Adapter:2 IPV6[0002:0000:0000:0000:0007:0008:0000:0000]/24
Active IP Addresses:
IPv4: 169.254.0.3
Exit with code: 0
```

Parameters:

None

Supported platforms: All

## ***qrytemp***

Reads the thermal sensor values of a tape drive and writes the following output:

```
Thermal Value, Max Thermal Value,
Fencing Threshold and Fencing Removal Threshold.
All values are units of °C.
```

Parameters:

None

## ***read***

This subcommand reads a file, or a specified number of records, from the tape to the destination file name specified with the **-d** flag. If the optional count parameter is used, only the number of records that are specified with the **-c** flag are read unless a filemark is encountered before the number of specified records. If the count parameter is not used, all records up to the next filemark on tape are read.

Parameters:

```
read -d Dest [-c Count]
```

Supported platforms: All

## ***readattr***

This subcommand reads a Medium Auxiliary Memory (MAM) attribute from a cartridge to the destination file name that is specified with the **-d** flag. The partition parameter **-p** is a number in the range 0 - 3. The attribute parameter **-a** is the hexadecimal value of the identifier. All three parameters are required.

Parameters:

```
readattr -p[0|1|2|3] -a Identifier -d DestinationPathFile
```

Supported platforms: All

### ***resetdrive***

This subcommand issues a Send Diagnostic command (Reset Drive subcommand) to reset the device.

Parameters:

None

Supported platforms: All

### ***rmp***

Remove partitioning.

Parameters:

None

Supported platforms: All

### ***runtimeinfo / qryruntimeinfo***

This subcommand is used to query the dynamic runtime information. Dynamic runtime information allows an initiator to set dynamic runtime attributes (DRA) about itself into a device server. The device server then associates those attributes to the I\_T\_L nexus and uses the information and associations for enhanced data collection and debugging.

Parameters:

None

Supported platforms: All (supported since LTO 5 and 3592 E07)

### ***rewind***

(Deprecated: **-o rew**) Rewinds the tape.

Parameters:

None

Supported platforms: All

### ***rtest***

(Deprecated: **-o rea**) This subcommand runs a read test by reading a random data pattern from the tape and verifying that it matches the written data. The **rtest** subcommand can be used after the **wtest** subcommand to verify the data. An optional block size, count, and repetition can be specified with the **-b**, **-c**, and **-r** flags, respectively. If the block size is fixed, then the count specifies the number of blocks to read on each repetition. If the block size is zero (variable), then the count specifies the number of bytes to read on each repetition. The default is a block size of 10240, a count of 20 blocks, and a repetition of 1.

Parameters:

```
read [-b Blocksize] [-c Count] [-r Repetition]
```

Supported platforms: All

## **rwtest**

This subcommand runs a read and write test by writing a random data pattern on the tape, reading it, and verifying that it matches the written data. An optional block size, count, and repetition can be specified with the **-b**, **-c**, and **-r** flags, respectively. If the block size is fixed, then the count specifies the number of blocks to write on each repetition. A single transfer transmits (block size \* count) bytes. The operation is rejected if the total amount exceeds the transfer size the system is capable of. If the block size is zero (variable), then the count specifies the number of bytes to write on each repetition. The default is a block size of 10240 bytes, a count of 20 blocks, and a repetition of 1.

In **SCSI Generic mode**, the actual transfer size equals to *blocksize*. In **IBM Device Driver mode**, the transfer size is *blocksize \* blocks per read/write*. In either case, the blocksize must be equal to or less than the maximum Host Bus Adapter supported transfer size. The maximum supported transfer size for **SCSI Generic** is 1048576 Bytes and for **IBM Device Driver** is 500 MB.

Parameters:

```
rwtest [-b Blocksize] [-c Count] [-r Repetition]
```

Supported platforms: All

## **sdp**

Creates SDP (Select Data Partitions) wrap wise on tape.

Parameters:

```
sdp count
For LTO (5 and higher) only the values 0 and 1 are valid.
Using value 0 as parameter leads to partition 0 with 1.5TB
and partition 1 does not exist.
Using value 1 as parameter leads to partition 0 with 750 GB
and partition 1 with 712.5 GB.
For TS1140 and later drives values 0, 1, 2 and 3 are valid.
Using value 0 as parameter will create only one partition,
value 1 creates two and so on.
The sizes of the partitions are depending on the cartridge used in drive.
```

Supported platforms: All

## **sdpl**

Creates SDP (Select Data Partitions) longitudinal on tape.

Parameters:

```
sdpl count
For TS1140 and later drives values 0 and 1 are valid.
Using the value 0 as the parameter creates partition 0 only
and the value 1 as the parameter creates partitions 0 and 1.
The sizes of the partitions depends on the cartridge
used in drive.
```

Supported platforms: All

## **seod**

(Deprecated: **-o eod**) Spaces to end of data on the tape.

Parameters:

```
None
```

Supported platforms: All

## setparm

(Deprecated: **-o spa / valid / compress / nocompress / sili / nosili / autoload / noautoload / retain / noretain**)

ITDT-SE uses the new setparm option that corresponds to the current interactive mode options.

Parameters: The Value is

- 0-65535 for the blocksize parameter
- 0-100 for the capacity parameter (=percentage)
- 1 (SCSI) and 2 (AIX) for recordspacemode
- The Volume Id string for the valid parameter
- NONE|ASSO|PERS|WORM for the **writeprotect** parameter
- 0 for off/no and 1 for on/yes for setparm autoload, autodump, buffering, compression, immediate, readpastfilemark, sili, simmim, trace, trailer, volumelogging

Supported platforms: All, but only a subset of the parameters is supported by the platform's device drivers.

Table 15. Supported platforms					
	Linux	Windows	AIX	Solaris	HP-UX
setparm autoload			X		
setparm autodump	X				
setparm blocksize	X	X	X	X	X
setparm buffering	X		X	X	X
setparm capacity	X	X <sup>3</sup>	X	X	X
setparm compression	X	X	X	X	X
setparm datasafemode <sup>1</sup>	X	X	X	X	X
setparm immediate	X		X	X	X
setparm readpastfilemark	X				
setparm recordspacemode			X		
setparm sili			X	X	X
setparm simmim	X				
setparm skipsync <sup>2</sup>	X	X	X	X	X
setparm sleepmode	X	X	X	X	X
setparm trace	X				
setparm trailer	X		X	X	X
setparm valid			X		
setparm volumelogging	X		X		
setparm writeprotect <sup>2</sup>	X	X <sup>3</sup>	X	X	X
setparm archivemodeunthread	X	X	X	X	X

### Note:

1. The datasafemode can be set to YES or NO when no cartridge is loaded. When a cartridge is loaded, the datasafemode can be set only to YES.

2. Depending on the support of the device.
3. Only supported by 3592.

### **setpos**

(Deprecated: **-o spo / asf**) This subcommand issues the SCSI Locate command to the device to set the tape position. If the optional Blockid parameter is specified, the tape position is set to the Blockid. Otherwise, if the **Blockid** parameter is omitted, the tape position is set to the last position saved by using the **qrypos** subcommand.

Parameters:

```
setpos [Blockid]
```

Supported platforms: All

### **settcpip**

This subcommand sets the drive ethernet port TCP/IP settings for LTO 5, TS1140, and later drives. Either a static IPv4 or IPv6 address can be set or DHCP enabled.

Example DHCP:

```
./itdt -f /dev/IBMtape3 settcpip DHCP
Initializing device...
Setting TCP/IP Configuration...
Reading current TCP/IP Configuration...
Number of Port Descriptors 1
Port Descriptor for Port 1
Number of Socket Descriptors 2
Adapter:1 IPV4 9.11.22.111/23 DHCP:1
Adapter:2 IPV6 [0002:0000:0000:0000:0007:0008:0000:0000]/24
Active IP Addresses:
Exit with code: 0
```

IPv4 or IPv6 addresses are entered in the syntax a.b.c.d/subnet\_mask\_length where a, b, c, and d are values with 1 to 3 digits. If the optional parameter subnet\_mask\_length is not specified, the current setting is kept.

Example static IPv4:

```
./itdt -f /dev/IBMtape3 settcpip 9.155.27.14/23
Initializing device...
Setting TCP/IP Configuration...
Reading current TCP/IP Configuration...
Number of Port Descriptors 1
Port Descriptor for Port 1
Number of Socket Descriptors 2
Adapter:1 IPV4 9.155.27.14/23 DHCP:0
Adapter:2 IPV6 [0002:0000:0000:0000:0007:0008:0000:0000]/24

Active IP Addresses: IPv4:

169.254.0.3 IPv4:

9.155.27.46 IPv4:

9.155.27.14

Exit with code: 0
```

Parameters:

```
dhcp, address[/subnet_mask_length]
```

Supported platforms: All

**Note:** With the current firmware level, the device can be reached (ping, FTP) only within the same subnet. For example, the sample is configured with a static IP address (9.155.27.14) and DHCP enabled. The

drive can be pinged only within the same subnet (9.155.27.xxx). Both IP addresses (9.155.27.14 and the DCP address 9.155.27.46) are active and can be used.

### ***sync***

(Deprecated: **-o syn**) This subcommand synchronizes buffers/flushes the tape buffers to tape.

Parameters:

```
None
```

Supported platforms: All

### ***unload***

(Deprecated: **-o off / offline / rewoffl**) This subcommand rewinds and unloads the tape.

Parameters:

```
unload [-amu 0|1]
```

amu enables or disables archive mode unthread during **Load** command.

Supported platforms: All

### ***verlbp***

This subcommand verifies logical block protection written blocks. The verification length can be set with parameter value filemarks count or with EOD.

Parameters:

```
filemarks (numeric value) | eod
```

Supported platforms: All

### ***weof***

(Deprecated: **-o eof / eof**) These subcommands write count filemarks. An optional count can be specified. The default is 1.

Parameters:

```
weof [Count]
```

**Note:** The weof parameter [count] is optional, if it is not supplied, one filemark is written.

Supported platforms: All

### ***write***

This subcommand writes the source file specified with the **-s** flag on the tape. In case the parameter '**raw**' is specified, the blocksize specified in setparm (setparm blocksize) is used instead of the default blocksize of 64 kB. The parameter count specifies the amount of blocks which should be written.

Parameters:

```
write [-raw] [-count] -s Source
```

Supported platforms: All

### ***writeattr***

This subcommand writes a Medium Auxiliary Memory (MAM) attribute from a cartridge from the source file name that is specified with the **-s** flag. The partition parameter **-p** is a number in the range 0 - 3. The attribute parameter **-a** is the hexadecimal value of the identifier. All three parameters are required.

Parameters:

```
writeattr -p[0|1|2|3] -aIdentifier -sSourcePathFile
```

Supported platforms: All

### **wtest**

(Deprecated: **-o wri**) This subcommand runs a write test by writing a random data pattern on the tape. The **rtest** subcommand can be used after the **wtest** subcommand to verify the data that was written. An optional block size, count, and repetition can be specified with the **-b**, **-c**, and **-r** flags, respectively. If the block size is fixed, the count specifies the number of blocks to write on each repetition. If the block size is zero (variable), the count specifies the number of bytes to write on each repetition. The default is a block size of 10240, a count of 20 blocks, and a repetition of 1.

Parameters:

```
wtest [-b Blocksize] [-c Count] [-r Repetition]
```

Supported platforms: All

### **audit**

(Deprecated: **-o aud / -o ier**) This subcommand with no parameters issues the SCSI Initialize Element Status command to the device. Using the optional parameters Address and Count issues the SCSI Initialize Element Status With Range command to the device. The Address parameter specifies the starting element address and the Count parameter, if used, specifies the number of elements to initialize. If Count is omitted, it defaults to 1.

Parameters:

```
audit [[Address] [Count]]
```

Supported platforms: All

### **cartridgelocation**

This subcommand with no parameters issues the SCSI Read Element Status command to the device to report all slots with the cartridge location information. Using the optional parameters Slot and Count issues the SCSI Read Element Status to the device for a specific starting Slot address and optionally the Count specifies the number of slots to return. If Count is omitted, it defaults to 1.

Parameters:

```
cartridgelocation [Slot [Count]]
```

Supported platforms: All

### **elementinfo**

(Deprecated: **-o ele**) This subcommand displays element information (number and address) of each element type.

Parameters:

```
None
```

Supported platforms: All

### **exchange**

(Deprecated: **-o exh**) This subcommand issues the SCSI Exchange Medium command to the device by using the **Source**, **Dest1**, and **Dest2** addresses specified. This command runs the equivalent function of two **Move Medium** commands. The first moves the cartridge from the element address that is specified

by the **Dest1** parameter to the element address specified by the **Dest2** parameter. The second moves the cartridge from the element address that is specified by the **Source** parameter to the element address specified by the **Dest1** parameter.

Parameters:

```
exchange Source Dest1 Dest2
```

Supported platforms: All

### **inventory**

(Deprecated: **-o inv**) This subcommand with no parameters issues the SCSI Read Element Status command for each element type and displays the element status information. If the optional **-i** parameter is used, then only the import/export element status information is returned. If the optional **-v** parameter is used, then only the element status information for the specified Valid if found is returned.

Parameters:

```
inventory [-i | -v Valid]
```

**Note:** ITDT supports the optional parameters on all platforms.

Supported platforms: All

### **librarymediaoptimization**

Starting with LTO-9 the LTO tape drive will perform a media characterization/optimization process for each initially loaded cartridge.

Regarding the longer lasting optimization time, cartridges should be optimized proactively in the target environment.

The `librarymediaoptimization` subcommand for medium changers offers the option to identify uninitialized cartridges in a logical tape library and load cartridges in the desired tape drive(s) for optimization.

This subcommand with no parameters determines all tape drives and storage slots available in a logical tape library and use those for media optimization.

Already initialized cartridges will be skipped.

With the optional **-start number** parameter, the first storage element address can be specified used for and with optional parameter **-slots numberofslots**, the numbers of slots can be limited.

The optional parameter **-drives numberofdrives** specifies the numbers of drives and with the optional parameter **-drivelist identifiers**, a defined list of drives is used **-identifiers** is a comma separated list containing the drive element address or the drive serial number.

The optional parameter **-volserlist identifiers** allows the specification of a list of volser names where the media characterization/optimization should be performed, **identifiers** is a comma separated list containing the volser names. The characters \* and ? can be used as wildcard characters .

Parameters:

```
librarymediaoptimization [-start number] [-slots numberofslots] [-drives numberofdrives] [-drivelist identifiers] [-volserlist identifiers]
```

**Note:** ITDT supports the optional parameters on all platforms.

Examples:

```
./itdt -f /dev/IBMChanger0 librarymediaoptimization
./itdt -f /dev/IBMChanger0 librarymediaoptimization -slots 10 -drives 2
./itdt -f /dev/sg7 librarymediaoptimization -start 4096 -slots 10
./itdt -f /dev/sg7 librarymediaoptimization -drivelist 01234567890,0123456789A
```



```
./itdt -f /dev/sg7 librarymediaoptimization -drivelist 256,257 -volserlist  
187AAHL9,187AAKL9,187AALL9
```

Supported platforms: All

### ***move***

(Deprecated: **-o mov**) This subcommand issues the SCSI Move Medium command by using the source and destination addresses specified. The element addresses can be obtained with the **elementinfo** subcommand.

Parameters:

```
move Source Dest
```

Supported platforms: All

### ***position***

(Deprecated: **-o pos**) This subcommand issues the SCSI Position to Element command by using the destination specified.

Parameters:

```
position Dest
```

Supported platforms: All

### ***dump***

(Deprecated: **-o sdp**) This subcommand forces a dump of the tape drive and stores the dumps before and after the force dump in the ITDT-SE output folder with the ITDT-SE naming convention (serialnumber.a.gz and serialnumber.b.gz).

Parameters:

```
None
```

Supported platforms: All

### ***ekmtest***

Test encryption key path/setup.

Parameters:

```
None
```

Supported platforms: AIX, Linux, Solaris, Windows

### ***encryption***

Query tape drive encryption settings and display the encryption state.

Parameters:

```
None
```

Supported platforms: All

### ***ucode***

(Deprecated: **-o dmc**) This subcommand downloads microcode to the device. The Filename is a file that contains the ucode.

The TS4300 (3555) tape library does not support code update with SCSI commands. ITDT integrated RESTful API functions to support code update by using the HTTPS connection of the library. The process is similar to the regular code update sequence, but in addition the user must enter credentials during the code update sequence.

Parameters:

```
ucode [-password loginpassword -user username -ipaddress library Address] Filename
```

Supported platforms: All

### ***tapephcp***

Creates a physical copy of a tape cartridge. The created cartridge has the same physical layout and contents as the origin cartridge. The amount of transferred data and the current data transfer rate is displayed every 3-5 minutes. Tapephcp is supported for LTO and 3592 tape drives and can therefore be used for data migration. A **tapephcp** command that is issued to copy data from a 3592 drive to an LTO drive or from an LTO Gen 4 to an LTO Gen 3 works, if the amount of used data from the source device is equal or less than the capacity of the destination device.

Parameters:

```
tapephcp [-cqs Memorysize] source destination
```

Source and destination can either a special device file name, or a tape image file name. The special device file format is identical to the format specified in chapter 3.19 parameter “-f device”.

Examples: Tape to Tape copy with the IBM tape driver

```
./itdt tapephcp /dev/IBMtape0 /dev/IBMtape1  
Tape to Image File using generic interface  
./itdt tapephcp 3 2 1 0 /tmp/MyTapeImage.img  
Image File to Tape using IBM Tape driver  
./itdt tapephcp /tmp/MyTapeImage.img /dev/IBMtape1  
Tape to Tape copy with adjusted memory allocation  
./itdt tapephcp -cqs 1000 /dev/IBMtape0 /dev/IBMtape1
```

To ensure maximum performance for tape to tape copy actions, tapephcp allocates a read buffer of 2500 MB (assuming the maximum system block size is 1 MB). If the system does not provide this buffer, the operation is aborted with MEMORY ALLOCATION FAILED. The value for memory allocation can be changed with an integer value for the parameter **cqs** (Command Queue Size). The **cqs** value must be multiplied with the maximum supported system blocksize to determine the size of memory that is allocated by ITDT. Supported copy operations are from tape to tape, from image file to tape, and from tape to image file. On UNIX operating systems, a warning to check the file size limit is shown before an image file is created, if the maximum file size is below 2 GB.

### ***ltfsphcp***

Creates a physical copy of an LTFS formatted cartridge. Ltfsphcp is based on tapephcp. The LTFS specified parameters **volumeuuid** and **VCI** are adjusted during this copy operation. The created cartridge has the same physical layout as the origin cartridge. Expect the **volumeuuid** to be identical to the contents of the two cartridges. The amount of transferred data and the current data transfer rate is displayed every 3-5 minutes. When ltfsphcp is used with a non-LTFS formatted cartridge, the behavior of ltfsphcp is identical to tapephcp.

Parameters:

```
ltfsphcp [-cqs Memorysize] [-VOL1 volume id] source destination
```

Source and destination can either a special device file name, or a tape image file name. The special device file format is identical to the format specified in chapter 3.19 parameter “-f device”.

Examples:

```
Tape to Tape copy using IBM tape driver
./itdt ltfsphcp /dev/IBMtape0 /dev/IBMtape1
Tape to Image File using generic interface
./itdt ltfsphcp 3 2 1 0 /tmp/MyTapeImage.img
Image File to Tape using IBM Tape driver
./itdt ltfsphcp /tmp/MyTapeImage.img /dev/IBMtape1
Tape to Tape copy with adjusted memory allocation
./itdt ltfsphcp -cqs 1000 /dev/IBMtape0 /dev/IBMtape1
```

To ensure maximum performance for tape to tape copy actions, ltfsphcp allocates a read buffer of 2500 MB (assuming the maximum system block size is 1 MB). If the system does not provide this buffer, the operation is aborted with **MEMORY ALLOCATION FAILED**. The value for memory allocation can be changed with an integer value for the parameter **cqs** (Command Queue Size). The **cqs** value must be multiplied with the maximum supported system blocksize to determine the size of memory that is allocated by ITDT. Supported copy operations are from tape to tape, from image file to tape, and from tape to image file. On UNIX operating systems, a warning to check the file size limit is shown before an image file is created, if the maximum file size is below 2 GB.

For LTFS formatted cartridges, the volume identifier can be changed for a physical target with the parameter **VOL1**. The value must be alphanumeric with a minimum of 1 and a maximum of 6 characters.

## ***verify***

Verifies the physical contents of two cartridges. The physical data layout and the binary data are compared.

Parameters:

```
verify source destination
```

Source and destination can either a special device file name, or a tape image file name. The special device file format is identical to the format specified in chapter 3.19 parameter **-f**.

Examples:

```
./itdt verify /dev/IBMtape0 /dev/IBMtape1
```

## ***devicestatistics***

This command collects relevant device parameters from several log pages and inquiry data and prints out in a table.

## ***checkltfsreadiness***

This subcommand issues the LTFS Readiness Check test.

The LTFS Readiness Check analyzes the Operating System and tape drive environment to ensure that the IBM Linear Tape file system can be installed. This test checks the Operating System version, the tape device driver version, the tape drive firmware, and the LTFS HBA requirements. LTFS Readiness Check requires an empty data cartridge.

Parameters:

```
None
```

## ***ltfsdefragmentation***

On an LTFS formatted cartridge, the physical data records for a single file can be fragmented across the entire media. When such a file is accessed, a long response time might result. The tape drive must locate to different cartridge positions to retrieve the entire contents the file. If the first data records of a file are stored at the end of the tape and the other records are stored at the beginning of the media, the tape drive must run several times intensive seek operations to fulfill the complete file retrieval. This subcommand creates a copy of the cartridge with unfragmented content.

As an initial step, ITDT stores the complete content of the source tape media in a Tape Image file that is on an HDD. Using this Tape Image file and the ITDT image file backend driver for LTFS, LTFS is able to mount the previously created Tape Image file as a read-only volume. As the final step, the data is defragmented by copying the files from the mounted Tape Image file to the mounted destination cartridge. This algorithm avoids any seek operations on the physical tape device. The seek operations are completed on the temporary ITDT image file that is on a hard disk. The source and destination tape device are accessed at maximum media speed. The defragmentation of a cartridge can take up to 6 hours.

Parameters:

```
ltfsdefragmentation source tempdirectory destination [options]
```

Supported platforms: Linux x86\_64 only

Prerequisites: IBM LTFS SDE Version 2.2 (Build 4700 or later) and sufficient free hard disk space for temporary Tape Image file

Example:

```
./itdt ltfsdegragmentation /dev/IBMtape0 /tmp/tapeimages  
/dev/IBMtape1 -verbose -mklftsoption=--force
```

### ***standardtest***

The Test function (Scan Menu Command [T]) checks if the tape device is defective and outputs a pass/fail result. This test requires a loaded cartridge.

Parameters:

```
standardtest [-forcedataoverwrite]
```

Supported platforms: All

### ***fullwrite***

The fullwrite command runs the ITDT full write test (Scan Menu Command [F]). Issuing a fullwrite command writes the entire cartridge with a given block size either with compressible or incompressible data and output performance data. This test requires a loaded cartridge.

Parameters:

```
fullwrite [-b Blocksize] [-compressible|-incompressible|  
1.5compressible|-2.5compressible|-5.0compressible|-100compressible] [-forcedataoverwrite][-  
percent number]
```

By using the optional parameters, such as -compressible, -incompressible, -1.5compressible, -2.5compressible, -5.0compressible and -100compressible, a data pattern with a predefined compressible ratio could be selected.

#### **Remember:**

- The parameters -compressible and -2.5compressible are equivalent.
- The -percent number can be set as a decimal point value ranging from 0.001 to 100 with up to 3 decimal places.

**Supported platforms:** All

### ***systemtest***

Runs the ITDT Systemtest (Scan Menu Command [Y]).

The System Test is a short test that runs the following actions.

- Reveals system performance bottlenecks. Compressible data throughput values can reveal bandwidth limitations that are caused by the system or cabling or HBA.

- Measures performance variations across the different block sizes to find the ideal block size for the system configuration.

This test requires a loaded cartridge.

Parameters:

```
systemtest [-forcedataoverwrite]
```

Supported platforms: All

## ***tapeusage***

Displays the tapeusage information (Scan Menu Command [U]).

This test requires a loaded cartridge.

Parameters:

None

Supported platforms: All

## ***TS4500 REST over SCSI***

The TS4500 supports a SCSI in band method for sending REST API commands and receiving HTTP responses that use SCSI **Write Buffer** and **Read Buffer** commands. The method is called **REST over SCSI** or **RoS**, for short. ITDT handles the RoS calls.

Syntax:

```
ros | rosraw GET | PATCH | POST url-endpoint [options] [-o filename]
```

Parameters:

- Use 'ros' for prettified output, 'rosraw' for raw mode (no indents, line breaks, Unicode filtering)
- HTTP-Method: GET | PATCH | POST
- Url-endpoint is the RoS method '/v1/...'
- Options in brackets {} - string must be masked with single quotation marks ('')
- Optional file name where the JSON is stored.

For a detailed URL endpoint definition, refer to the TS4500 RESTful documentation: <https://www.ibm.com/docs/en/ts4500-tape-library/1.8.0.3?topic=reference-rest-over-scsi-api>.

Example:

```
$.itdt -f /dev/smc30 ros GET /v1/accessors HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 461
[
  {
    "location": "accessor_Aa", "state": "onlineActive", "pivots": "13256",
    "barCodeScans": "109653", "travelX": "9475",
    "travelY": "11640",
    "getsGripper1": "4488",
    "putsGripper1": "4487",
    "getsGripper2": "4486",
    "putsGripper2": "4465"
  },
  {
    "location": "accessor_Ab", "state": "onlineActive", "pivots": "3496",
    "barCodeScans": "61212", "travelX": "2845",
    "travelY": "2139",
    "getsGripper1": "160",
    "putsGripper1": "154",
    "getsGripper2": "163",
    "putsGripper2": "148"
  }
]
```

```
]
Exit with code: 0
```

## Endpoints with file transfers

The URL-endpoints

- GET /v1/logs/[filename]/export
- GET /v1/library/saveConfig

store the content in a file. The file name is printed at the end.

Examples:

```
./itdt -f /dev/smc30 ros POST /v1/tasks '{"type":"inventoryTier0and1","location":"library"}'
./itdt -f /dev/smc30 ros GET /v1/logs/TS4500_LOG_AA004_20191106143648.zip/export
./itdt -f /dev/smc30 ros GET '/v1/diagnosticCartridges/DG 0200L4'
```

### Notes:

1. The IBM Tape Device Driver is sensitive to not ready conditions that can prevent the usage of RoS commands. For this reason, the generic operating system driver is recommended with RoS. For details, see [“Generic Operating System driver with ITDT-SE ” on page 85.](#)
2. The Microsoft Windows command prompt does not support a single quotation mark (') to send a string to an application. Every special character must be escaped. Example:

```
ros POST /v1/tasks '{"type":"inventoryTier0and1","location":"library"}'
```

works on Linux, but must be sent on Windows as:

```
ros POST /v1/tasks {"type\\":\\"inventoryTier0and1\\",\\"location\\":\\"library\\"}
```

## TS4300: RESTful API

The TS4300 REST API is a simple application planning interface (API) to manage the 3U scalable tape libraries remotely over an HTTPS interface. This API is requested and needed for manufacturing and for automated test and monitoring systems.

Syntax:

```
https-address {login-credentials} GET | PATCH | POST url-endpoint [JSON data]
```

Parameters:

- https-address
- Login credentials in JSON format
- HTTP-Method: GET | PATCH | POST
- url-endpoint
- [JSON data]

### https-address

The IP address of the library.

### Login credentials

The library differentiates the following security levels:

- Admin Security
- User Security
- Service Security

The user/password that is used to log in. Additionally, for some product variants in case of service level login the service password (service\_password) and the administrator password (password) must be sent. In case the administrator password is not required the 'service\_password' must be set through the normal password field. Parameter:

```
'{"username": "administrator",  
  "password": "password"  
  ["service_password": ""servicepassword"]}{'
```

## http-method

The method for a dedicated request: GET, PUT, POST

## url-endpoint

Valid URLs in combination with the http method. If the parameter is required, it can be passed with ? and &.

For a detailed URL endpoint definition, refer to the TS4300 RESTful documentation: [https://www.ibm.com/support/knowledgecenter/STAKKZ/ts4300\\_kc/con\\_3U\\_REST\\_overview.html](https://www.ibm.com/support/knowledgecenter/STAKKZ/ts4300_kc/con_3U_REST_overview.html).

## JSON data

Some commands need extra data in JSON format like 'serialnumber change'.

Example:

```
sudo ./itdt https://1.2.3.4 '{"username": "service", "password": "123456"}' GET /library/baseinfo  
{  
  BaseInfo {  
    "SerialNumber": "1234567890123455",  
    "MacAdress_1": "00:0e:11:1c:31:b5",  
    "MacAdress_2": "00:0e:11:1c:31:b7",  
    "Vendor": "IBM",  
    "ProductID": "3573-TL",  
    "BaseFWRevision": "1.3.0.0-D00",  
    "BaseFWBuildDate": "07-02-2019",  
    "ExpansionFWRevision": "0.30",  
    "WWNodeName": "5000000000000000",  
    "RoboticHWRevision": "4",  
    "RoboticFWRevision": "0.13",  
    "RoboticSerialNumber": "12345678901234", "NoOfModules": "2",  
    "LibraryType": "32"  
  },  
  "ModulesInfo": [  
    {  
      "PhysicalNumber": "4",  
      "LogicalNumber": "1",  
      "ReadyStatus": "TRUE",  
      "SerialNumber": "3555L3A1234567"  
    },  
    {  
      "PhysicalNumber": "5",  
      "LogicalNumber": "2",  
      "ReadyStatus": "TRUE",  
      "SerialNumber": "3555E3A0234567"  
    }  
  ]  
}  
Exit with code: 0
```

## Notes:

1. ITDT sends a login request for each command.
2. Microsoft Windows command prompt does not support a single quotation mark (') to send a string to an application. That is, every special character must be escaped. The Microsoft Windows command shell is different and the command must be sent with the following form.

Example for Windows:

```
itdt.exe https://1.2.3.4 "{\"username\":\"service\",\"password\":\"123456\"}" GET /library/  
baseinfo  
itdt.exe https://1.2.3.4 "{\"username\":\"service\",\"password\":\"123456\"}" GET "/library/  
getevents?EventType=Info&MaxNum=2"
```

### ***Deprecated commands***

The following is a list of commands that are currently available in this version of ITDT. However, in a future release the following commands and some alternate calls of the Common Command Scripting set are no longer available and the scripts that contain these commands must be changed. The scripts that use the deprecated commands must be changed for future editions.

General commands

- -o dis / disablepath
- -o ena / enablepath
- fuser
- kill
- passthru
- resetpath

Tape commands

- bsfm
- -o chk
- fsfm
- -o grs
- list
- -o gmi
- qryinquiry
- qrysense
- -o ret
- setblk
- -o gds
- getrecsize
- setrecsize

Medium Changer subcommands

- -o dvc
- mount
- demount

Service Aid commands

- fmrtape
- -o fdp
- reset
- -o qmc



## Alternative mount/demount script - sample for Windows

The script (load.drive.cmd) issues a **mount/unmount** command to the specified library. If the drive is empty, a cartridge is moved from the first available storage slot to the drive. If the drive contains a cartridge, the cartridge is moved to the previous storage location.

Requirements:

- Windows operating system
- IBM tape library with one or more tape drives
- IBM tape device driver or Generic SCSI driver
- Path variable must include ITDT executable

```
Usage: load-drive.cmd drivename|hbt1 [drivenumber]
       drivename is the device driver name assigned by the IBM device driver
       hbt1 is the associated SCSI address h=host, b=bus, T=target id and l=lun
       drivenumber, logical drive number, default is 1
```

Example for using the IBM device driver: load-drive.cmd \\.\Changer0

Example for using the SCSI Generic driver: load-drive.cmd "2 0 3 1"

```
load-drive.cmd:
@ECHO OFF
IF [%1]==[] (
    echo.
    echo Usage load-drive.cmd drivename^|hbt1 [drivenumber]
    echo.
    echo drivename is the drivename assigned by the IBM device driver
    echo example:load-drive.cmd \\.\Changer0
    echo.
    echo hbt1 is the associated SCSI address h=host, b=bus, T=target id and l=lun
    echo example: load-drive.cmd "3 0 2 1"
    echo.
    goto :EOF
)

set driveNumber=1
IF not [%2]==[] set driveNumber=%2
echo Loading/Unload Cartridge Drive:%driveNumber%

itdt -f %~1 inventory >inventory.txt
set /a count=0
set action=load
set sourceAddress=
set destination=
set driveAddress=
set slotAddress=
FOR /F "tokens=1,2,3,4,5" %%G IN (inventory.txt) DO (
    call :checkline %%G %%H %%I %%J %%K
)
GOTO :moveMedium

:checkline
set name=%1 %2
if "%name%"=="Drive Address" (
    set /a count+=1
    set currentTag=%name%
    set driveAddress=%3
)
if "%name%"=="Media Present" (
    rem echo full addr:%driveAddress% slot addr:%slotAddress%
    action:%action% source:%sourceAddress% Tag:%currentTag%

    if "%count%"=="%driveNumber%" if "%currentTag%"=="Drive Address" if "%4"=="Yes" (
        set action=unload
        set sourceAddress=%driveAddress%
    )

    if "%count%"=="%driveNumber%" if "%currentTag%"=="Drive Address"
    if not "%4"=="Yes" (
        set action=load
        set destination=%driveAddress%
    )
)
```

```

    if "%sourceAddress%"==" " if "%currentTag%"=="Slot Address"
    if "%action%"=="load" if "%4"=="Yes" (
        set sourceAddress=%slotAddress%
    )
)

if "%name%"=="Source Element" (
    if "%count%"=="%driveNumber%" if "%currentTag%"=="Drive Address"
    if "%action%"=="unload" (
        set destination=%5
    )
)

if "%name%"=="Slot Address" (
    set currentTag=%name%
    set slotAddress=%4
)

GOTO:EOF
:moveMedium
echo %action% move media from %sourceAddress% to destination %destination%
itdt -f %~1 move %sourceAddress% %destination%
del inventory.txt
GOTO:EOF

```

### ***Standard Edition scripting commands: known limitations and deviations***

The scripting commands **idp**, **sdp**, **fdp**, and **qrypart** are currently only supported by LTO 5, TS1140, and later drives.

When scripting, one must be aware of the following general deviations to the legacy tapeutil scripting command set.

The **Verbose Mode** parameter is ignored for the ITDT-SE implementation; instead always the full information is printed.

For some operations, the sense data length is limited to 32 bytes - this length is required by the ITDT test sequences.

The list command does not work on Linux variants where the reported maximum SCSI transfer size is larger than the size the system can actually transfer.

Because ITDT-SE opens the device read/write by default in scripting mode, the WORM mode cannot be deactivated in scripting mode. Use the Tapeutil interactive mode instead to deactivate the WORM mode.

Scripting mode command deviations to legacy tapeutil (tapeutil is not changed):

1. The **erg** (Erase Gap) command is not supported.
2. The **mtdevice** command is not supported.
3. The **tell** command is not supported.
4. The **seek** command is not supported.
5. The **format** command is not supported.
6. The **-o qsn** command is not supported.
7. path/querypath/path / qrypath / checkpath / - o phs - output: always show all paths.

The command is valid only in combination with an IBM device driver.

8. devinfo - different output (decoded on all platforms)
9. inquiry - different output on Linux (like AIX in hex)
10. vpd - different behavior on Solaris (like AIX)
11. modepage - HP-UX and Solaris output deviations (like AIX)
12. inventory - additional parameters on AIX available on all platforms
13. cartridgelocation - AIX parameter deviation available on all platforms
14. mediainfo - different output --> decode on all platforms

The command is valid only in combination with an IBM device driver.

15. setpos - logical versus physical position, only set logical position as on AIX
16. HPUX: **-o gpo -t 1|2 --> parameter -t1|2** (logical, physical) is not supported.
17. density - output all information on all platforms
18. setparm (new) - work around the inability to set all parameters on all platforms except the undocumented HP-UX release
19. getparms (new) - retrieve all parameters on all platforms, independent of flag set
20. qryinquiry - output the same as on AIX
21. logsense - does not output header with device/serial/valid, example:  

```
05/12/09 15:10:44 Device: ULT3580- S/N: 1300000206 Valid: UNKNOWN
```
22. erase - does not work on i5/OS because of operating system limitation.
23. Using **-w x** parameter without the necessary open leads to a core at least on Solaris.

## IBM Tape Diagnostic Tool - Graphical Edition

### Installing ITDT - Graphical Edition

#### *Installing ITDT-GE on Windows operating systems*

To install ITDT-GE on Windows, download the executable file **install\_ITDT\_GE\_<version>.exe** on a directory of your choice.

Double-click the downloaded file to start the installer application.

ITDT-GE installer automatically uninstalls any previous version before the current one is installed.

For the graphical user interface (GUI), a minimum screen resolution of 1024\*768 pixels is required.

The supported Microsoft Windows operating systems for ITDT-GE are:

- Microsoft Windows Server 2016 / 2012R2 (64-bit x64)

#### *Installing ITDT-GE on Linux operating systems*

To install ITDT-GE on Linux, download the executable file **install\_ITDT\_GE\_<version>.bin** to a directory of your choice.

**install\_ITDT\_GE\_<version>.bin** must be run by a user with root access rights.

The ITDT-GE installer automatically uninstalls any previous version before the current one is installed.

For the graphical user interface (GUI) a minimum screen resolution of 1024\*768 pixels is required. The supported Linux operating systems for ITDT-GE are

- Linux Distributions with Kernel 2.6, glibc 2.2.5 and later (64-bit x64).

### Graphical Edition - known issues and limitations

This section describes the known issues and limitations of ITDT-GE.

#### *Linux operating systems*

It is recommended to not operate Security Enhanced Linux (SELinux) in enforcing mode while ITDT-GE is running.

On Red Hat Enterprise Linux 4 and 5 and SuSE Enterprise Linux 10, starting the online help might cause an error message Problems opening link / Unable to open web browser on {0}. Workarounds are to issue the commands

```
a) ln -s /usr/bin/firefox /usr/local/bin/mozilla
or
b) export MOZILLA_FIVE_HOME=/usr/lib/firefox<version>
```

Replace with the appropriate path to your installed Firefox version before ITDT-GE is started.

## **Gnome desktop**

If you are using the Gnome desktop, be sure to log in to the desktop session as root to use ITDT-GE to prevent device access issues.

## **Linux GTK**

ITDT-GE requires GTK3+ to run since version 9.4.0. On Linux systems with GTK2, ITDT can be installed or updated but does not start afterward.

## ***Windows operating systems***

On Microsoft Windows systems where the maximum transfer size is limited to less than 64 kB, the Dump and Firmware update operations do not work.

## **Performance issues**

If you are using Adaptec SCSI Host Bus Adapters, ensure that you are using the latest Adaptec Host Bus Adapter Drivers instead of the drivers that are shipped with the Windows operating system.

## **Devices disappear after firmware update**

After a firmware update, devices might disappear.

Repeated **Scan** operations might help to rediscover the device.

## ***All supported operating systems***

## **Command timeout**

There is no instant operation termination upon SCSI command timeout. For example, when the SCSI cable is unplugged after POST A is started.

When a command timeout condition occurs, ITDT might still continue to run more operations (like unmounting the cartridge) instead of instantly terminating with a timeout condition.

## ITDT-GE user interface description

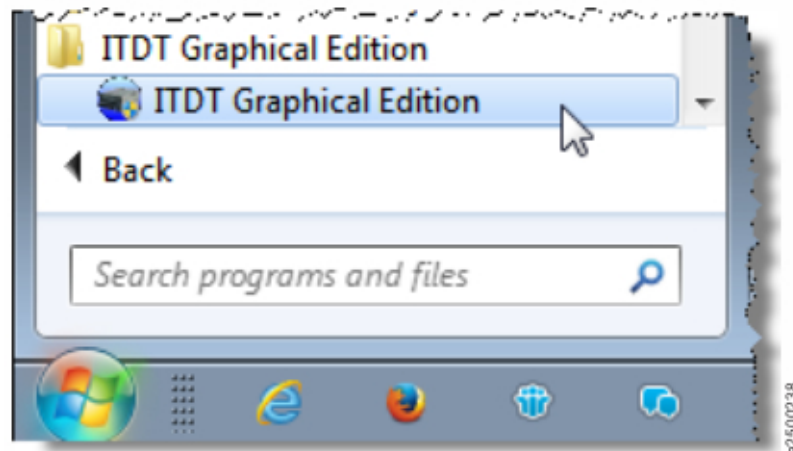


Figure 35. Graphical Edition user interface

To start ITDT-GE on Windows, click the shortcut that is created by the installation process. On Linux, no start menu entry is generated. Start ITDT-GE by opening a Terminal window, then switch to root user.

```
$ su -
```

Finally, start ITDT-GE with

```
$ /opt/ibm/itdt-ge/itdt-ge
```

The **User Settings** dialog box displays the first time that the program is run, allowing the entry of user specifications: User name, company name, output path, and log level.

The Output Path defines the location where the test reports and dumps are saved. The Windows default output path is

```
C:\Documents and Settings\<username>\.itdt-ge\output
```

or

```
C:\Users\<username>\.itdt-ge\output
```

where <username> is the Windows user login name. The Linux default output path is

```
/root/.itdt-ge/output
```

The Log Level must not be changed unless requested to do so by the IBM Support Center. It is recommended that this information be provided to allow for further analysis of test results.

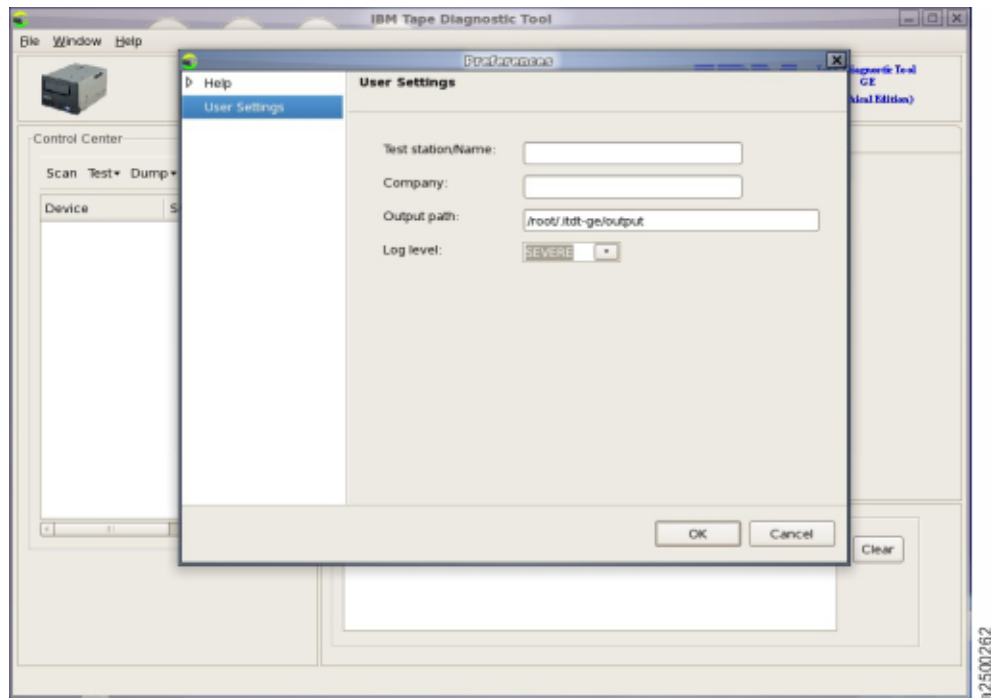


Figure 36. Graphical Edition preferences

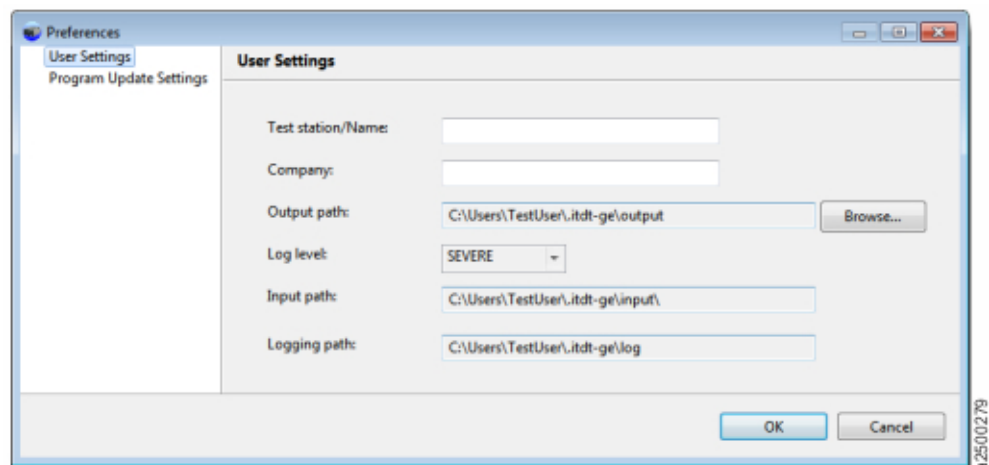


Figure 37. Graphical Edition preferences

The ITDT-GE interface contains the following sections:

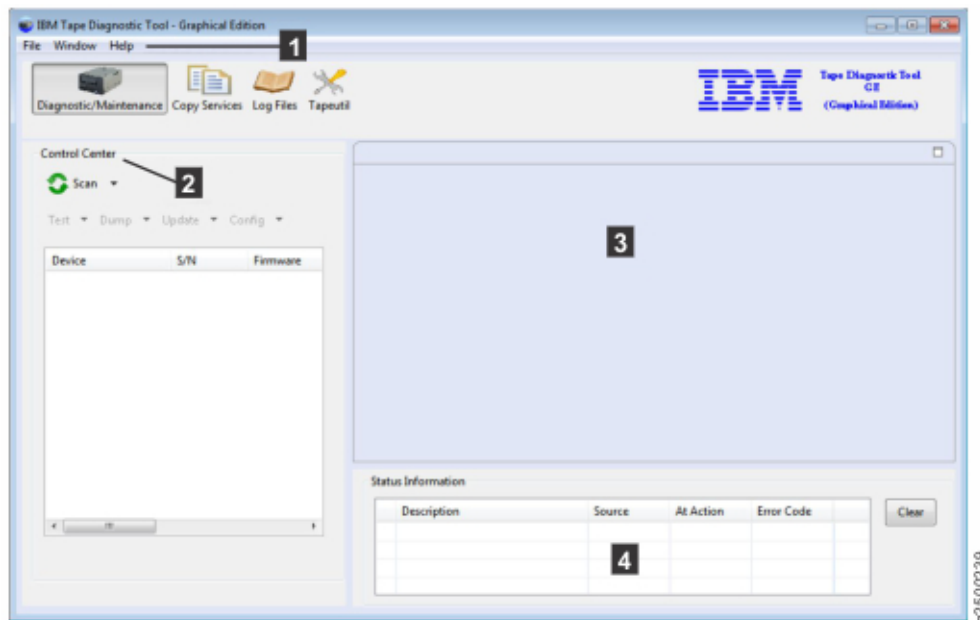


Figure 38. Graphical Edition interface

- Main menu (1 on Figure 38 on page 155)- in upper left (File, Window, Help) The following main program menu items are available:

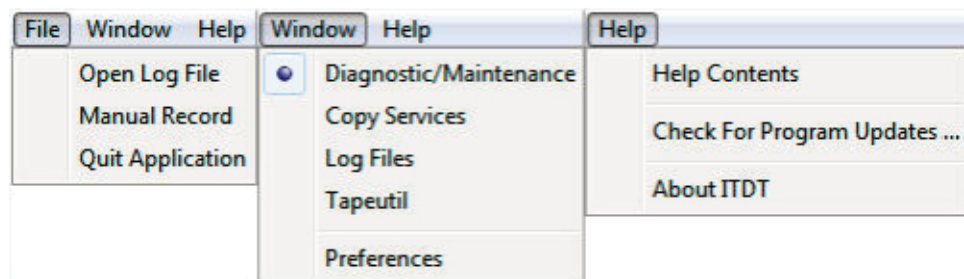


Figure 39. Main program menu items

- Control Center (2 on Figure 38 on page 155) - On left side (Device operations - Scan, Test, Dump, and Update)  
Extra device operations are available by using drop-down arrows.
- Test Lab (3 on Figure 38 on page 155) - Located from center to right side (Contains running and previously run tests)
- Status Information (4 on Figure 38 on page 155) - Located below the Test Lab (Contains summary results of tests)

The Control Center is the primary section the ITDT-GE operations.

The following toolbar buttons for the device operations are available.

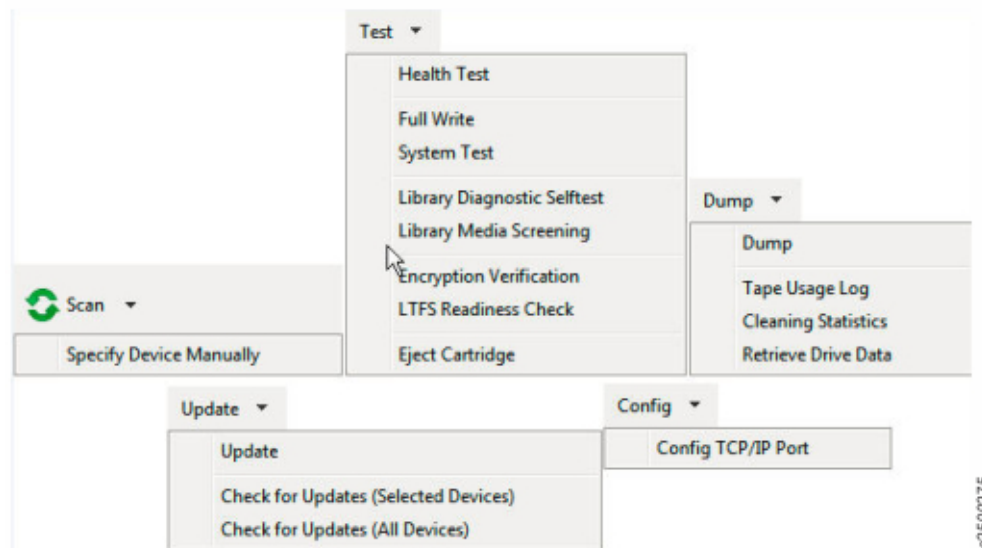


Figure 40. Toolbar buttons

- Click the **Scan** menu item in the Control Center to display the list of tape devices found. When the scan is complete, select one device in the Test Lab by clicking the corresponding check box. Click the arrow next to the **Scan** menu item to add a device manually.
- **Test**  
Click the arrow next to the **Test** menu item to select an extended test.
- **Dump**  
Click the arrow next to the **Dump** menu item to select more log options.
- **Update**  
Click the **Update** menu item to start the firmware update. Click the arrow next to the **Update** menu item to select Online update options.
- **Config**  
Click the **TCP/IP Port** menu item to configure the TCP/IP port.

For each device on which the operation is started, a tab displays on the right panel (Test Lab area). The tab contains the operation information and status. Only one operation can be run at a time with ITDT-GE. The benefit in using tabs even for the single operation mode is that you get a history of operations as for each consecutive operation, a separate tab is opened.

## Graphical Edition - Scan menu commands

The following commands are described in this section:



## Scan

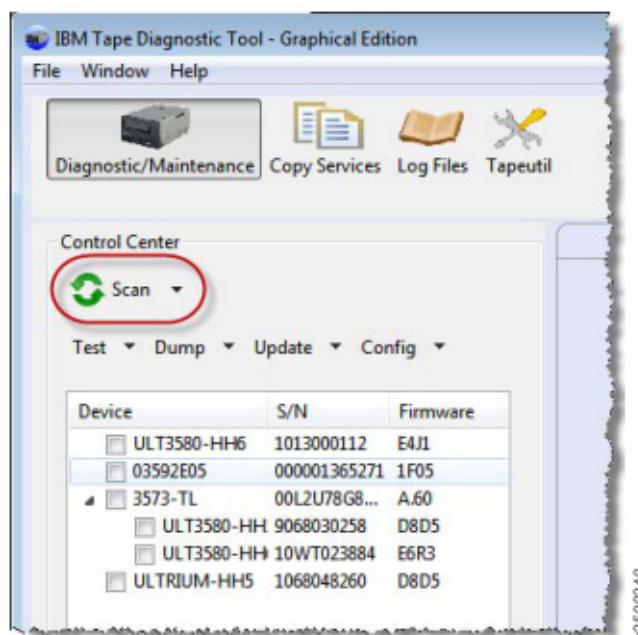


Figure 41. Scan

The scan function is used to discover all supported tape and library devices that are attached to the computer system. Then, they can be selected for the subsequent ITDT-GE operations. The scan function also serves as a connection test that can be used to verify correct attachment of the devices.

**Note:** When **Scan** is pressed for the first time, a dialog box is displayed that warns the user to stop all backup jobs.

When the scan is finished, the device list is displayed in the Control Center area.

A scroll bar is available to show all the devices. When the device you want is displayed, select the device for test. Only one device can be selected.

## Specify Device Manually

The drop-down list next to **Scan** allows you to define a tape device and an associated changer device that must be used for testing. This function is a fast alternative to the **Scan** function where all Host Bus Adapters are discovered for supported tape and changer devices. The function supports the device name and HBTL addressing schema. For stand-alone tape devices the parameter **Changer Device** is not required.

Example device names:

```
Windows: \\.\tape0
Linux: /dev/IBMtape0
```

Example HBTL (all platforms):

```
H1-B0-T3-L0 or 1 0 3 0
```

## Partial Scan

Partial Scan is used instead of **Scan** in cases when only a particular HBA or target ID range is used for device-discovering. The scan starts at "First device to scan" and checks for supported devices until "Last device to scan" is reached. Devices that are specified within "Exclude from scan" are skipped during the scan. If "First device to scan", "Last device to scan", or "Exclude from Scan" is not specified, ITDT performs a regular scan.

Partial Scan supports the device name and HBTL addressing schema.

Example device names:

```
Windows: \\.\tape0  
Linux: /dev/IBMtape0
```

Example HBTL (all platforms):

```
H1-B0-T3-L0 or 1 0 3 0
```

## Health Test

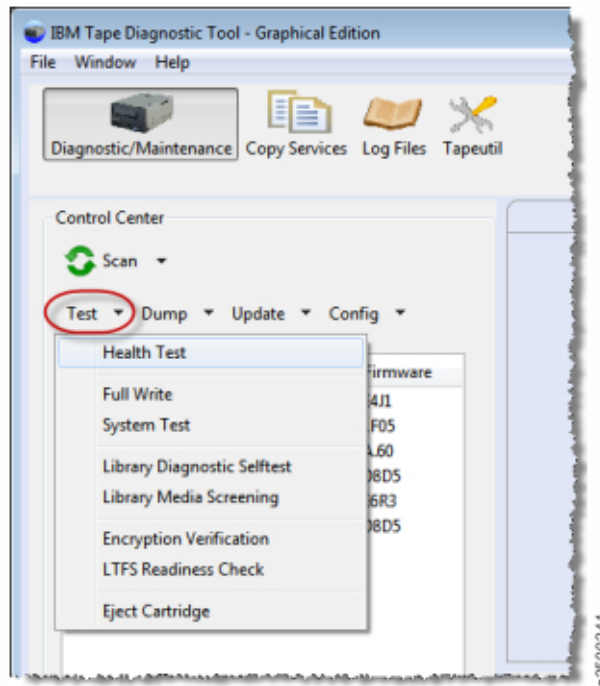


Figure 42. Test

The Health Test function checks if the tape device is defective and outputs a pass/fail result.



### Attention:

1. The test functionality erases user data on the cartridge that is used for the test.
2. The test can take from 15 minutes up to 2 hours.
3. The test runs only on tape drives, not on autoloaders or libraries.

To start the Health Test function, it is recommended that a new or rarely used cartridge is used. Scaled (capacity-reduced) cartridges must not be used to test the device.

To test tape drives within a library, the library must be in online mode.

After the device is selected, start the Health Test function by selecting the **Health Test** menu item.

The Health Test function can be stopped by clicking **Abort**.

**Note:** It can take some time until the Health Test function stops.

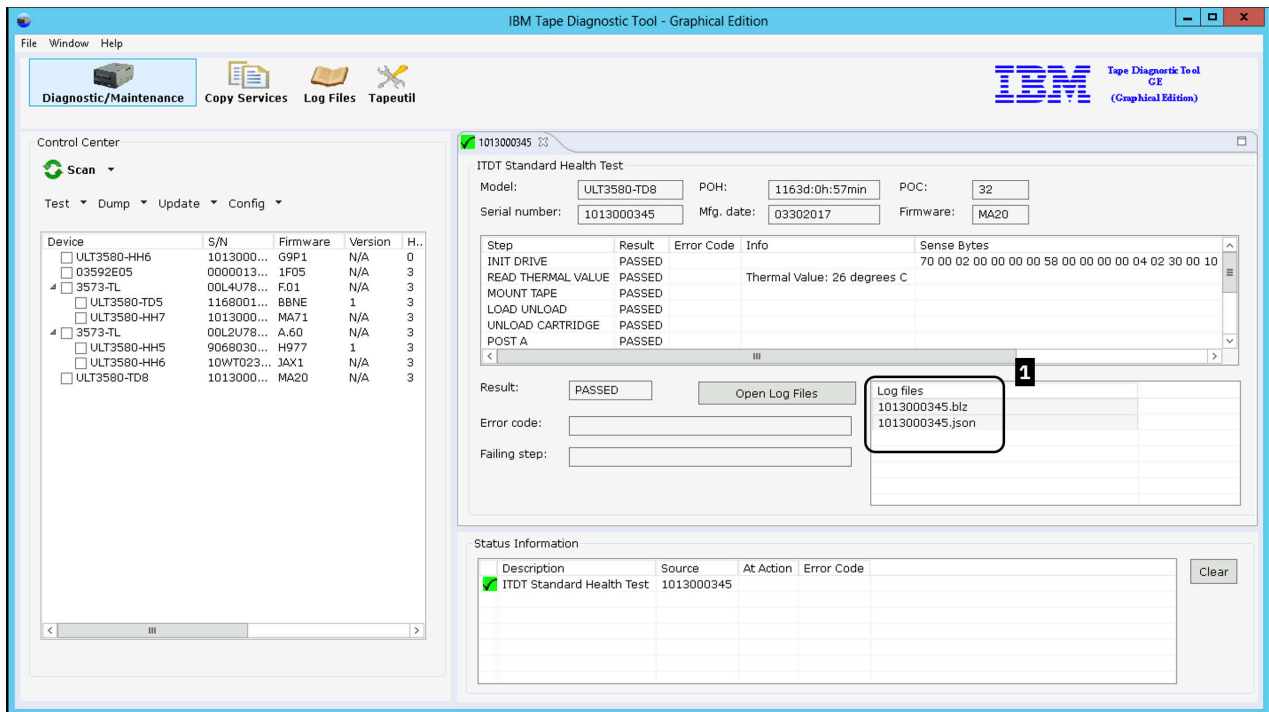


Figure 43. Test results

**Note:** Information can be found in the .json/.blz files. See the **Log files** section (1).

The test sequence contains the following steps:

1. Initialize Drive
2. Read Thermal Sensor
3. Mount Medium
4. [Medium Qualification] – only if previous step indicated requirement
5. Load/Write/Unload/Read/Verify
6. POST A
7. Performance Test (run 2 times if first run failed with performance failure)
8. Unmount Medium
9. Read Thermal Sensor
10. Get FSC
11. Get Logs

### Configure TCP/IP Ports

For LTO 5, TS1140, and later drives, the ethernet port settings can be configured with the **Configure TCP/IP Port** command. The **Configure TCP/IP Port** command displays the current settings:

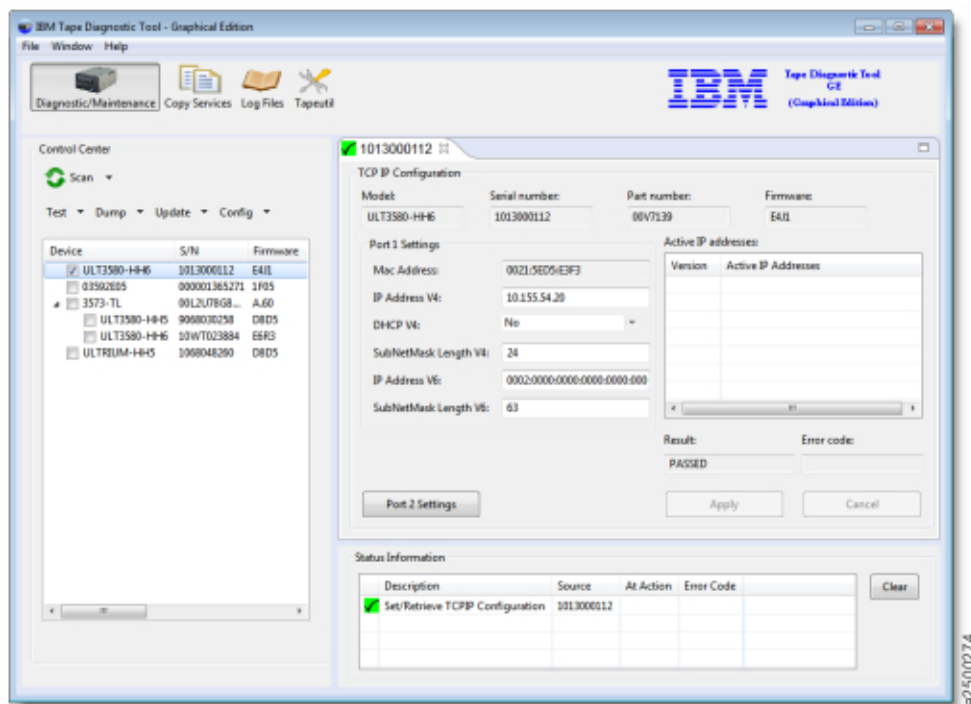


Figure 44. Configure TCP/IP Ports

After you click **Apply**, the new values are set and the updated values display.

**Note:** Because earlier drive generations do not have an ethernet port, the **Configure TCP/IP Ports** command is rejected for these devices with the following message:

TCP/IP configuration is not supported on this product.

## Dump

After the device you want to dump is selected, start the Dump function by selecting **Dump > Dump** from the actions toolbar.

When the dump process is run on a tape library or autoloader other than the 3584/TS3500/TS4500, the Dump function stores 1 log file in the output folder of the program (\*.blz). For the 3584/TS3500/TS4500, a dump file (\*.a) is stored in the output folder. Both files start with the serial number of the device (1).

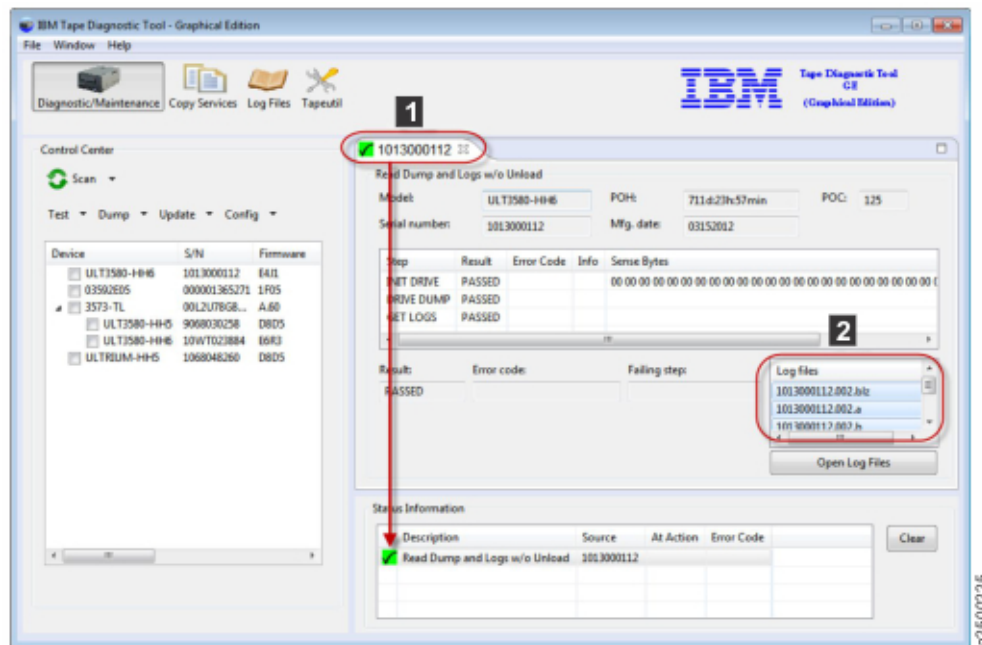


Figure 45. Dump results

**Note:** When the Dump function is run for tape libraries or autoloaders other than the 3584/TS3500/TS4500, the log file contains Log Sense and Mode Sense pages only. A Drive or 3584/TS3500/TS4500 dump contains more diagnostic information. (2)

## Firmware Update

The Firmware Update upgrades the firmware of tape drives and tape libraries.

**Note:** See [supported operating systems](#) for information on how to update the firmware on a TS4500 tape library.

The following site is available for the latest firmware files: <http://www.ibm.com/storage/lto>

To do a Firmware Update, run the following steps:

1. Select the device that you want to update.
2. Select the **Update** menu item.
3. A standard file dialog opens to select the path of where the firmware update is located. The path is either the default input path or the location of the last successfully opened firmware file.
4. Press **OK** on this file dialog to start the update on the selected device.
5. During the firmware update, a firmware update progress screen is displayed, indicating the progress of the update.



**Attention:** Once started, do not interrupt the firmware update.

The firmware update usually takes 3-5 minutes, but it can take up to 45 minutes for libraries.

**Note:** If ITDT-GE detects a FIPS-certified drive firmware, it shows a warning dialog message. Before you continue, ensure that you use a FIPS-certified firmware to update the drive.

## Firmware Update - check for updates

ITDT-GE can **Check for Device Updates** for IBM tape drives and IBM tape libraries. You can choose to select either one device or all devices by selecting the corresponding option from the drop down menu under **Update** in the ITDT Control Center.

The program connects to IBM FixCentral and identify updates for the devices. If the connection cannot be established or another problem occurred, the problem description is shown at the bottom of "FixCentral" Test Tab.

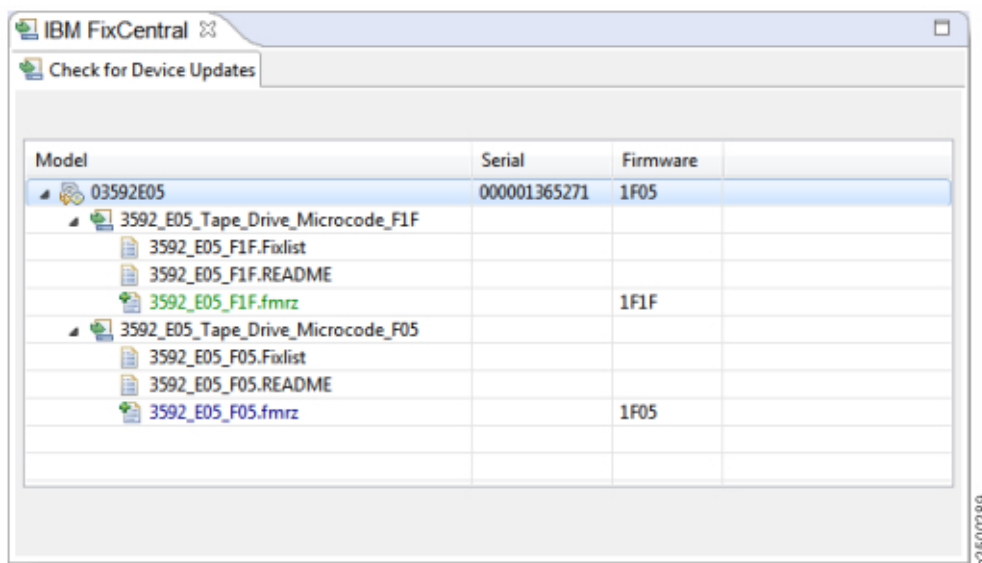


Figure 46. Check for Device Updates - FixCentral components

In the sample above, the drive has two FixCentral components. Each component has several items; such as code files and textual meta information.

The code files (binary files) have a "+" in the icon to distinguish and can be downloaded either by double-clicking or by using the right mouse button. The text files (for example, readme files) can be viewed in the same way. A separate Test Tab is opened and the information shown.

For tape drives an available code is colored:

- GREEN: the code level is newer than the one of the drive.
- RED: the code level is older than the one of the drive.
- BLUE: the code level is the same as the one of the drive.

For automation drives (drives in a library), both devices are used. A code level for an automation drive is directly linked to a code level of a library.

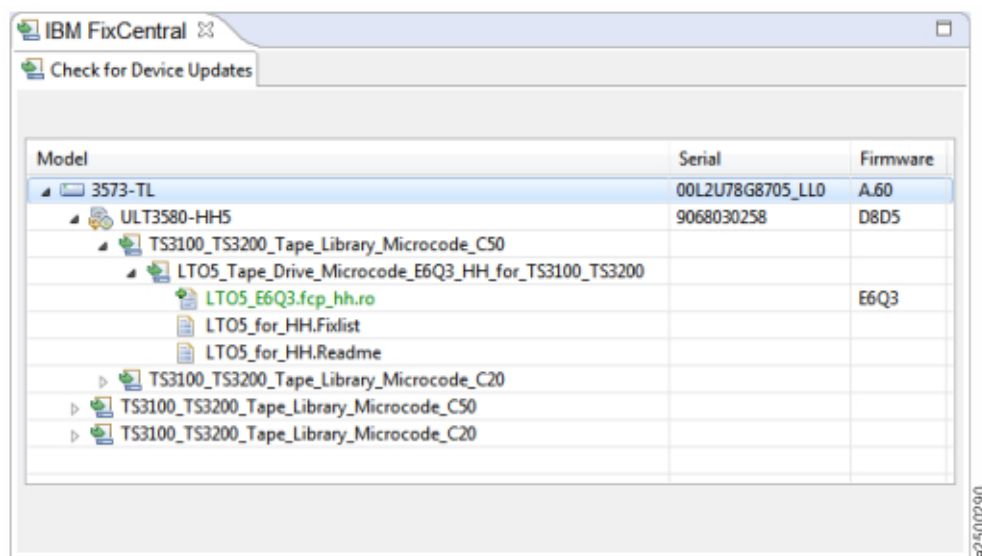


Figure 47. Check for Device Updates - code level

## Encryption

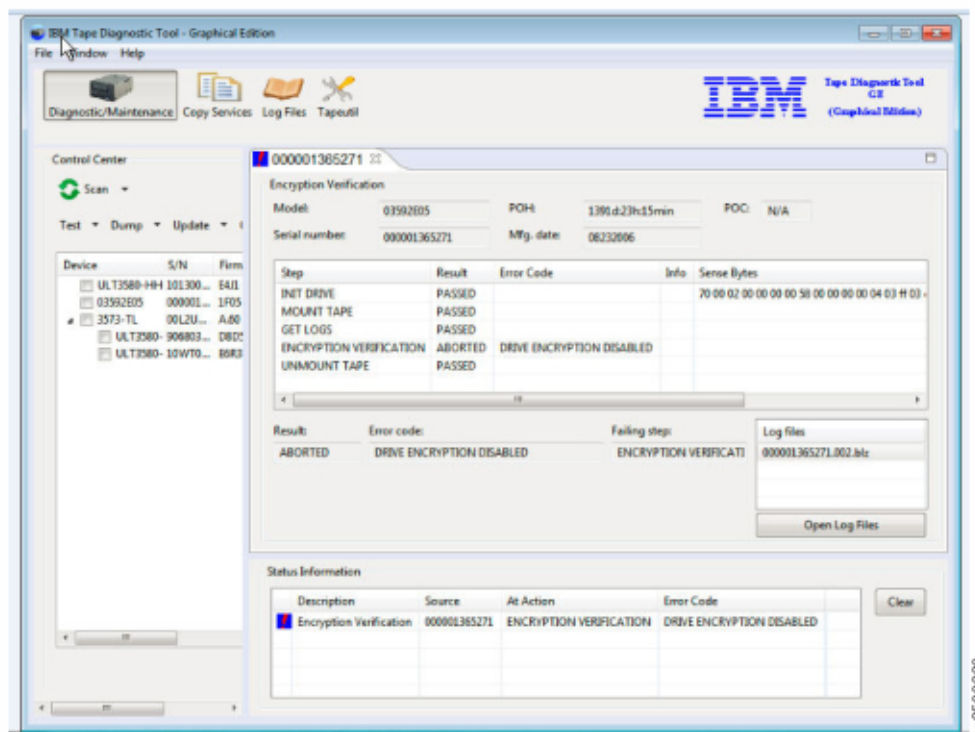


Figure 48. Encryption

**Note:** The test that is shown on the graphic was run on a non-encrypted device and is showing a failure.

The Encryption function is used to verify whether data on the cartridge is written encrypted. It reads both decrypted and raw data from the cartridge into two separate files on disk. The user can then verify that the data differs to ensure that encryption worked.

The Encryption function is only supported on encryption enabled drives and requires an encryption infrastructure, including the Encryption Key Manager (EKM) to be properly set up.

1. After the device you want to test is selected, start the encryption function by selecting **Test > Encryption Verification** from the actions toolbar.
2. ITDT-GE then shows the **Encryption Verification** screen. On this screen, the system requires the entry of the number of the start record and the amount of data (in KB) to be read.
3. Enter the required values and click **OK** to start the encryption.

The Encryption function can be stopped by clicking **Abort**.

**Note:** It can take some time until the Encryption function stops.

Table 13 on page 105 defines the abort codes.

## Full Write

The Full Write function writes the entire cartridge with a specified block size either with compressible or incompressible data and output performance data.



### Attention:

1. The Full Write function erases data on the cartridge that is used for the test.
2. The Full Write function takes approximately 2 hours when incompressible data is written, less time for compressible data.
3. The Full Write function runs only on tape drives, not on autoloaders or libraries.

The Full Write test can be used to

- Demonstrate that the drive can write the full amount of data on a cartridge.
- Reveal system performance bottlenecks.

Full Write runs the test with incompressible data first, then runs the test with compressible data. If no difference in overall performance is observed, a bandwidth limitation might be caused by the system or cabling or HBA.

- Identify system issues with compression.

Compression is always turned on during the full write. When run with compressible data, the output shows the compression rate. If it is higher than 1.0 but your system is not able to compress data on the cartridge, check your device driver and software settings to see whether they disable compression.

To run a full write on tape drives within a library, the library must be in online mode.

1. After the device you want to write to is selected, start the Full Write function by selecting **Test > Full Write** from the actions toolbar.
2. Click **OK** to start the full write.
3. ITDT-GE then shows the **Full Write** screen. If no cartridge is inserted, ITDT-GE prompts you to insert a cartridge. Either insert a cartridge and click **OK** or stop by clicking **Abort**.

**Note:** If ITDT-GE detects data on the cartridge, it shows the following message.

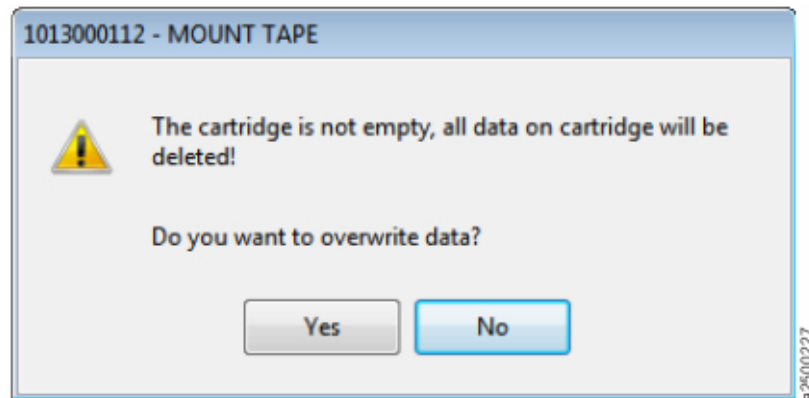


Figure 49. Overwrite data?

Click **Yes** to continue the test if you are sure that data on the cartridge can be overwritten. If you are unsure, click **No** to stop the test.

4. The system prompts for entry of a transfer size between 16 KB and the maximum block size that is supported by the system. Maximum value is 512 KB. Select the appropriate value for your system.



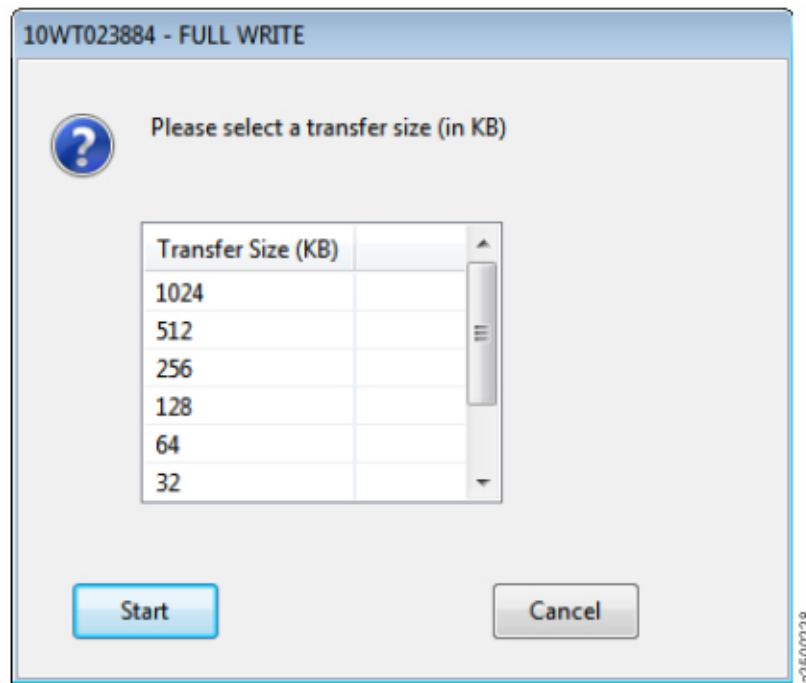


Figure 50. Transfer size

5. Select the type of data to write, either **[C] Compressible** or **[I] Incompressible**.

The full write can be stopped by clicking **Abort**.

**Note:** It can take some time until the full write actually stops.

"Compressible = Yes" means that the data written was just zeros so that the data can be compressed by the drive with a maximum compression ratio. "Compressible = No" means that a data pattern was written that the drive almost cannot compress at all. If the compression ratio is 1, the drive was not able to compress the data (equivalent to 1:1 compression ratio). If the compression ratio is 94.0, the drive was able to do 94:1 compression. It means that 94 bytes in the original data was compressed to 1 byte on the medium. 100.0 means 100 bytes is compressed down to 1 byte on the medium.

If all write operations are finished, ITDT-GE shows the performance statistics for the selected block size that is written on the cartridge, in the **Status Information** area. If an error occurred during the full write, data is only partially written.

## Tape Usage

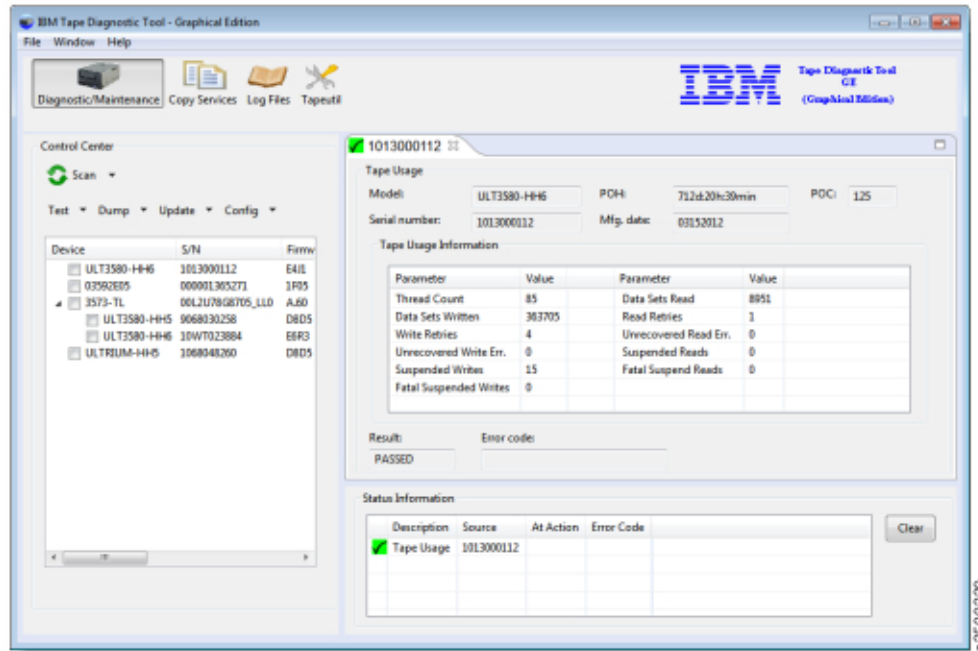


Figure 51. Tape Usage

The Tape Usage function retrieves statistical data and error counters from a cartridge.

1. After the device you want to test is selected, start the Tape Usage function by selecting **Dump > Tape Usage Log** from the actions toolbar.
2. ITDT-GE then shows the tape usage screen. If no cartridge is inserted, ITDT-GE prompts you to insert a cartridge. Either insert a cartridge and click **OK** or stop by clicking **Abort**.

## System Test

The System Test is a short test that runs the following tests:

- Reveals system performance bottlenecks. Compressible data throughput values can reveal bandwidth limitations that are caused by the system or cabling or HBA.
- Measures performance variations across the different block sizes to find the ideal block size for the system configuration.

The System Test runs only on tape drives, not on autoloaders or libraries. To run a System Test on tape drives within a library, the library must be in online mode.

After the device you want to test is selected, start the System Test function by selecting **Test > System Test** from the actions toolbar.

ITDT-GE then shows the **System Test** screen. If no cartridge is inserted, ITDT-GE prompts you to insert a cartridge. Either insert a cartridge and click **OK** or stop by clicking **Abort**.

**Note:** If ITDT-GE detects data on the cartridge, it shows the following message -

Cartridge not empty!  
Overwrite data?

Click **Yes** to continue the test if you are sure that data on the cartridge can be overwritten. If you are unsure, click **No** to stop the test.

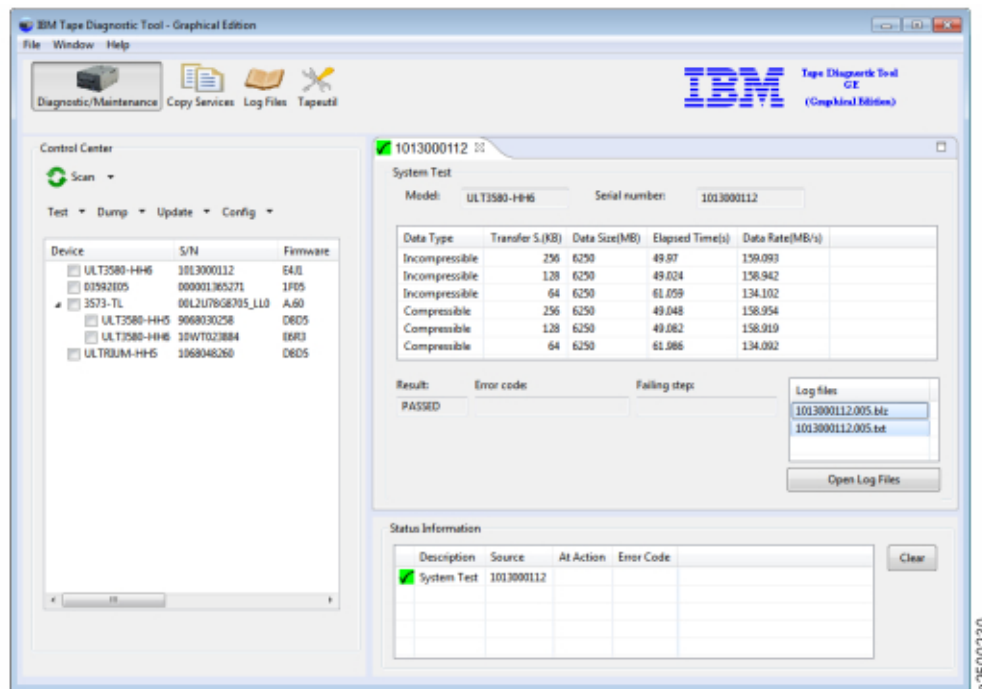


Figure 52. System Test

The System Test is run as follows:

1. System Test determines the amount of data to write for each supported blocksize (a percentage of the cartridge is written for each blocksize).
2. The test determines the maximum supported blocksize of the system.
3. System Test writes the amount of data with all supported block sizes in powers of two down to 64 KB (values of 16 KB and 32 KB are not tested in cases where the capability of a system supports higher block sizes). It begins with the maximum supported blocksize that is detected before, first with incompressible, next with compressible data. Then, it shows performance data and a progress screen.
4. At the end of the test, a summary information is shown.

### Library Diagnostic Self-Test

The Library Self-Test starts and monitors the library-internal self-test. This test runs only on libraries and autoloaders, not on tape drives.

After the device you want to test is selected, start the Library Self-Test function by selecting **Test > Library Diagnostic Self-Test** from the actions toolbar.

At the end of the test, the results are shown in the Status Information area.

### Library Media Screening

The Automated Library Media Screening test generates dumps for each drive and cartridge within a library. It runs only on libraries (except TS3500/TS4500) and auto-loaders, not on tape drives.

First, the test tries to read dump files from each drive that is installed from the library. Then, you can select one drive for loading the cartridges.

All cartridges of the I/O and storage slots are moved - one after the other from their source to the selected drive. A dump is taken and moved back to the source address. In the result screen, the dumps taken and the count of dumps are shown.

## LTFS Readiness Check

The LTFS Readiness Check analyzes the Operating System and tape drive environment to ensure that the IBM Linear Tape File system can be installed. For extended information, refer to [“Check LTFS Readiness” on page 103](#).

## Manual Inspection Record Entry

A manual inspection record can be generated if the device does not show in the device list. This test is intended for devices that are not recognized or have a technical problem that cannot be determined by ITDT-GE.

If a tape drive cannot be identified by using a device scan, the user can manually create a test record for the drive. The system prompts the user to run the SCSI/FC Wrap test for the drive (see the service manual for the drive). The results of the wrap test can be entered along with extra inspection information. The results are saved into binary and text output files that have the same format as the output files generated by the test.

1. From the **Main Program** menu, select **File > Manual Record**.
2. Enter the required information to complete the items in the screen.
  - a. Enter the device serial number.
  - b. Enter the content of the Message Display.
  - c. Optionally, enter any information text.
3. After all information is entered, click **OK**.

The information is stored in a binary file (which can be used for further analysis), and in a human-readable text file. Both files are stored in the ITDT-GE output folder.

## Graphical Edition - Copy Services

With **Copy and Migration Services**, tape content can either be copied or moved

- From a cartridge to another cartridge
- From a cartridge to an image file, or
- From an image file to a cartridge.

On UNIX operating systems, a warning to check the file size limit is shown before an image file is created, if the maximum file size is below 2 GB.

The tool offers different use cases like copying data or migrating data from one generation to another. That is, data can be copied from a Gen4 cartridge to a Gen5 cartridge. Even migration from an LTO to IBM Enterprise Tape Systems is supported. For an LTFS environment, the data can be copied or moved with the LTFS Physical copy that adjusts the LTFS parameter on the target to be unique. Such a copy can be used in the same LTFS environment as the source. To ensure maximum performance for tape to tape copy actions, ITDT allocates a read buffer that calculates to 2500 multiplied by the maximum supported system blocksize. If the system does not provide this buffer, the operation is aborted with **MEMORY ALLOCATION FAILED**. An infinite progress bar indicates the ongoing copy procedure. A dialog box confirms the completion.

## Tape Physical Copy

Creates a physical copy of a tape cartridge either to another cartridge or to an image file. The created cartridge has the same physical layout and contents as the source cartridge. **Tape Physical Copy** supports LTO and 3592 tape drives and can therefore be used for data migration. A copy from a previous generation to a newer generation cartridge and from an LTO to a 3592 works. However, the amount of used data from the source device must be equal or less than the capacity of the destination device. If the source cartridge is partitioned, the target must support the same partitioning sizes and amount. The amount of transferred data and the current data transfer rate is displayed every 3 - 5 minutes.

## LTFS Physical Copy

Creates a physical copy of an LTFS formatted cartridge. The LTFS specified parameters **volumeuuid** and **VCI** are adjusted during this copy operation. The created cartridge has the same physical layout as the origin cartridge. Expect the **volumeuuid** to be identical to the contents of the two cartridges. The amount of transferred data and the current data transfer rate is displayed every 3 - 5 minutes. When 'LTFS Physical Copy' is used with a non-LTFS formatted cartridge, the behavior is identical to **Tape Physical Copy**. For LTFS formatted cartridges, the volume identifier can be changed for a physical target by specifying **New LTFS Volume Label**.

## Verify

Verifies the physical contents of two cartridges. The physical data layout and the binary data are compared.

## Usage

By opening the "Copy Services" perspective, the user can run a 'Scan...' to discover attached devices.

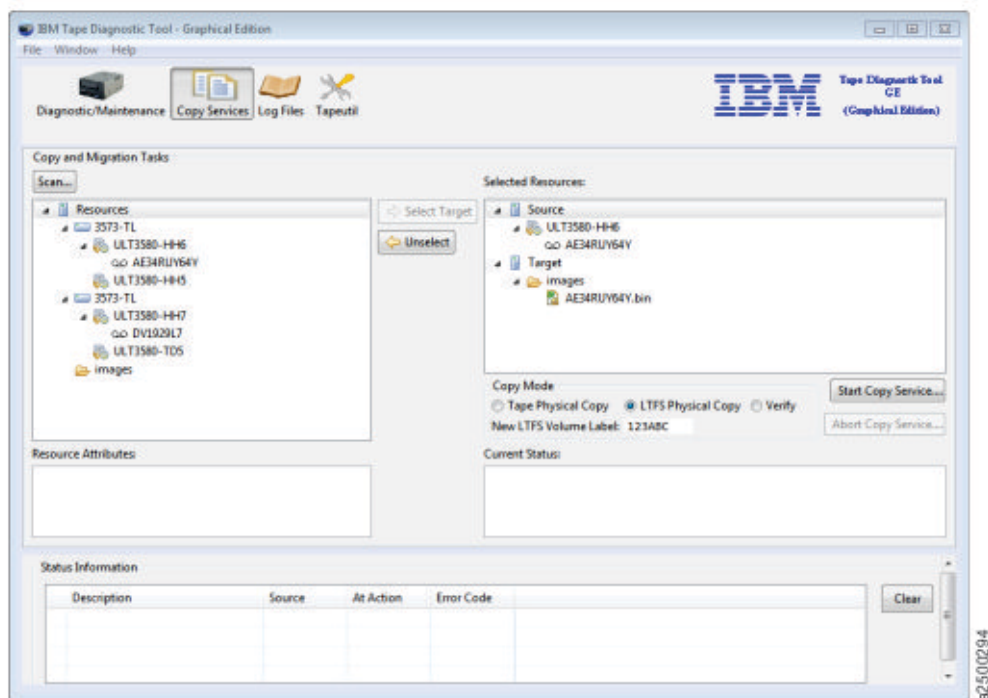


Figure 53. Copy Services

A cartridge can be loaded or unloaded by double-clicking or pressing the right mouse button on the drive that is used for the copy tasks. When a cartridge is loaded, the cartridge Serial Number or the VOLSER in a library environment is shown. If a cartridge or an image file is selected on the left, it can be used as "Source" by pushing **Select Source**. After the "Source" is selected, the target can be chosen. To select the target, either a different cartridge in a different drive or the image file folder can be selected. By pressing **Select Target**, the target is set on the right.

If the source and the target are selected, the Copy Services or a verification can be started regarding the used Mode: "Tape Physical Copy" on page 168, "LTFS Physical Copy" on page 169, or "Verify" on page 169.

## Graphical Edition - visualizing log files

ITDT-GE offers the opportunity to visualize the content of DUMP and ITDT log files (BLOB files).

- Dump files can be retrieved with ITDT or any other tool that supports this function.

- Dump files that are generated by ITDT have the suffix ".a" or ".b".
- BLOB files are generated during the run of an ITDT test sequence, such as "Standard Test".
- BLOB files have the suffix ".blz".
- Both file types can be opened and visualized with ITDT-GE.

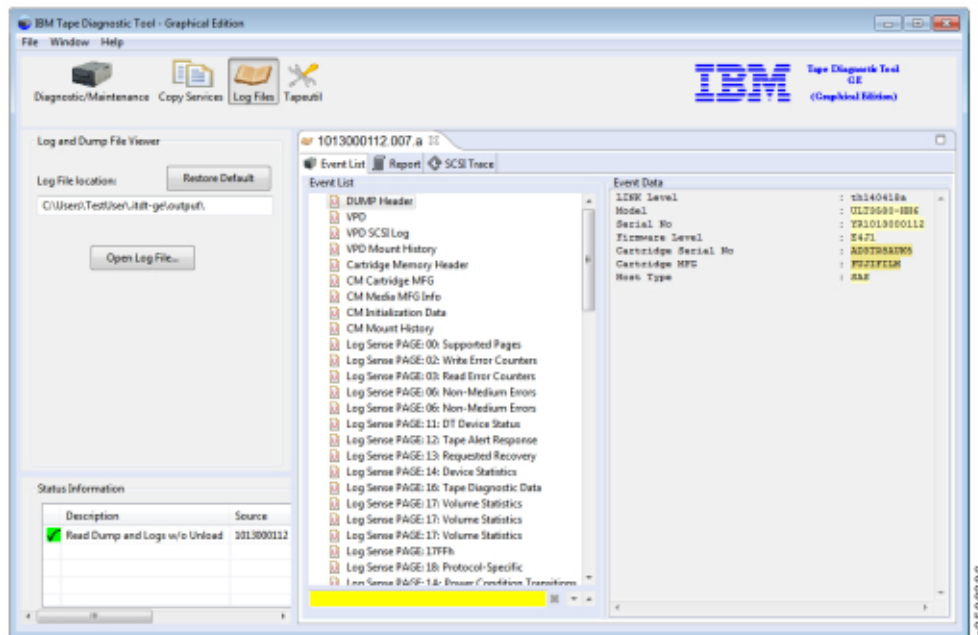


Figure 54. Graphic Edition: log view

## Opening a Dump or BLOB file

A Dump or a BLOB file can be opened either with the menu command **File/Open Log File** or by switching to the **Log Files** perspective and pressing **Open Log File ....** In both cases, a **File Selection** dialog opens where files can be selected for display. After the successful opening of a file, the data is shown.

Two views are available for presenting the data - each at a separate panel.

- The **Event List** shows the events of this file on the left side. An event is a group of information and consists of 1 to n Elements shown on the right. By selecting one event list entry on the left, the corresponding data (event elements) is shown on the right.
- The **Report** panel offers the opportunity to generate a sublist of the available 'Event List' entries.

## Graphical Edition - Tapeutil menu commands

After the initial startup, ITDT-GE shows three figures under the top menu. After the **Tapeutil** option is selected, the following page opens.

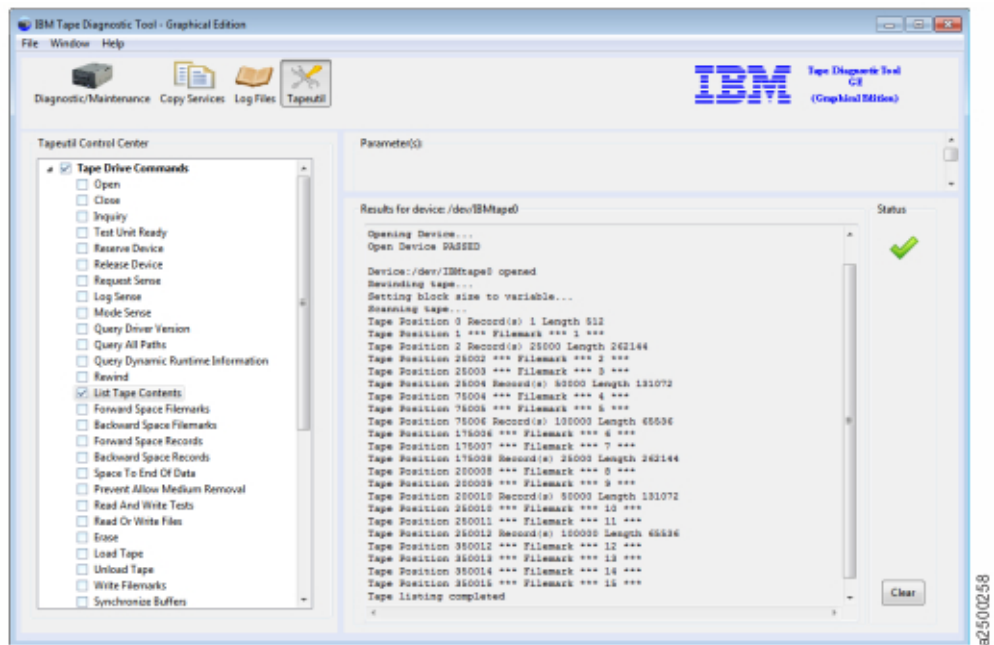


Figure 55. Tapeutil Control Center

On the left, the **Tapeutil Control Center** tree contains all Tapeutil commands for tape drives and tape libraries.

The commands from the **General Commands** and **Service Aid Commands** categories are duplicated into the two sections (to make the GUI navigation easier). After one category is expanded, the related commands open that allows users to select the command.

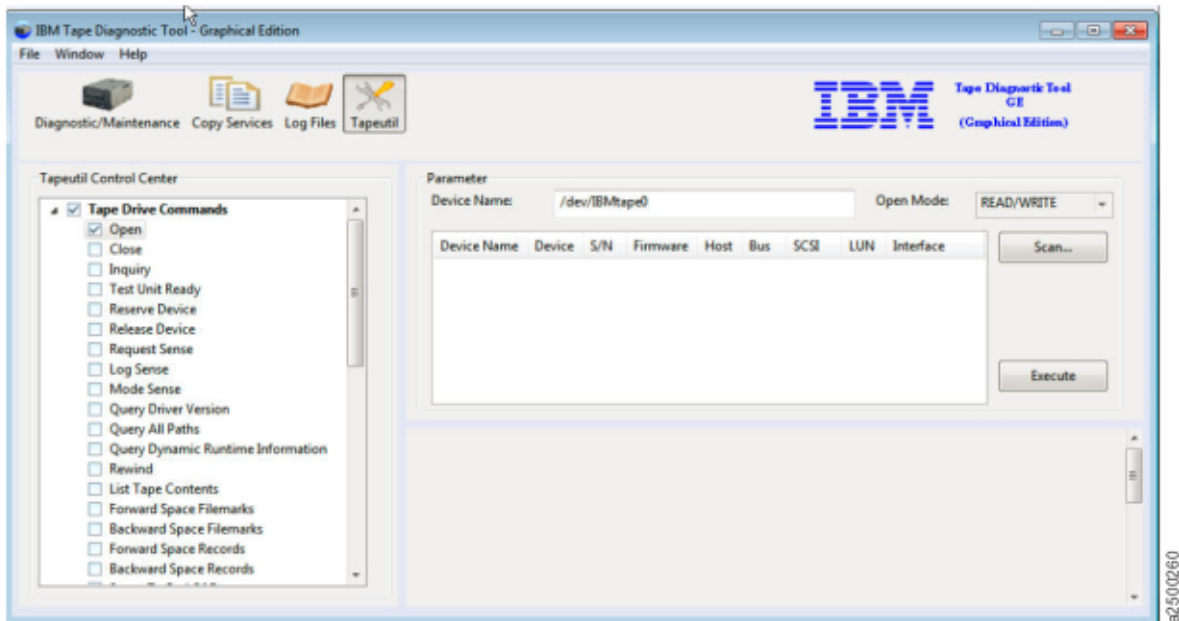


Figure 56. Command parameters

When the user presses **Execute**, the **Results** output is placed below the **Parameter** section:



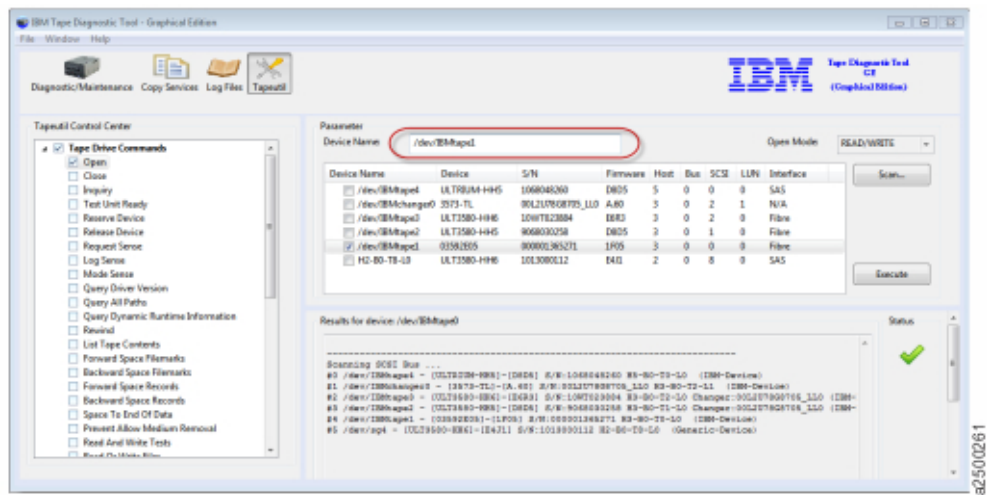


Figure 57. Command parameter results

The **Open** command has a **Scan ...** button in the **Parameter** view. Pressing **Scan...** runs a scan on the host and shows the attached devices in the **Result View** at the bottom. This information is helpful to identify the right Device Name for the open function field **Device Name**.

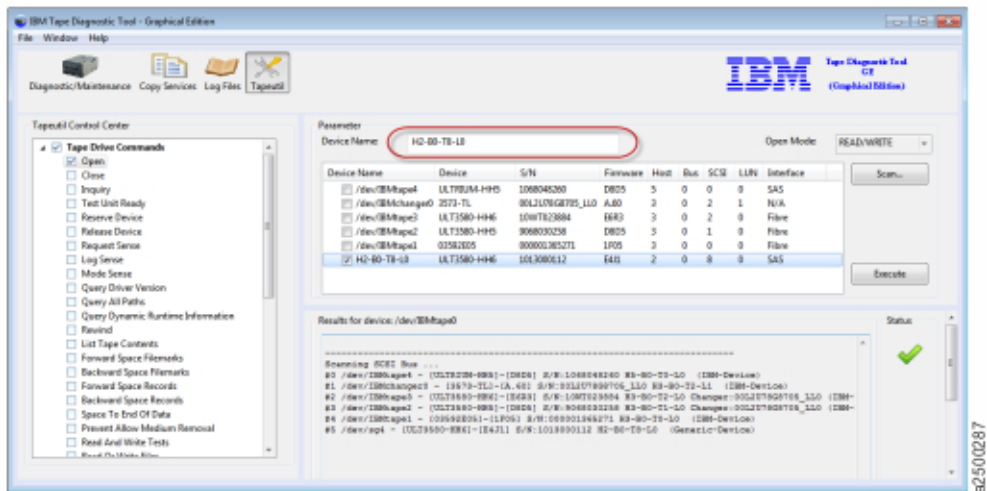


Figure 58. Generic tapeutil scan



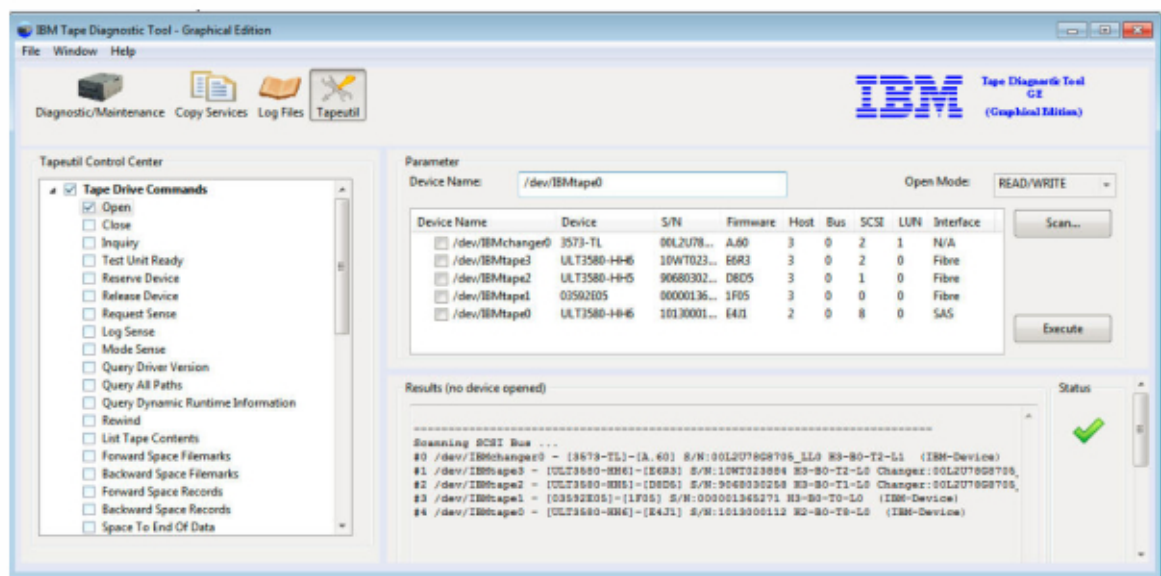


Figure 59. DD tapeutil scan

This screen layout stays within the **Tapeutil** perspective until the program is closed. Outputs of subsequent operations are added to the **Results** field. Commands that fail are indicated with a red cross in the **Status** area. Commands that succeed are indicated with a green check mark. The status area can be cleared by pressing **Clear**.

General commands

- [“Open” on page 174](#)
- [“Close” on page 174](#)
- [“Inquiry” on page 174](#)
- [“Test Unit Ready” on page 175](#)
- [“Reserve Device” on page 175](#)
- [“Release Device” on page 175](#)
- [“Request Sense” on page 175](#)
- [“Log Sense” on page 175](#)
- [“Mode Sense” on page 175](#)
- [“Query Driver Version” on page 175](#)
- [“Display All Paths” on page 175](#)

Tape drive specific commands

- [“Rewind” on page 175](#)
- [“Forward Space Filemarks” on page 175](#)
- [“Backward Space Filemarks” on page 176](#)
- [“Servo Channel Calibration” on page 176](#)
- [“Forward Space Records” on page 176](#)
- [“Backward Space Records” on page 176](#)
- [“Space to End of Data” on page 176](#)
- [“Read and Write Tests” on page 176](#)
- [“Read or Write Files” on page 178](#)
- [“Erase” on page 178](#)

- [“Load Tape” on page 178](#)
- [“Unload Tape” on page 179](#)
- [“Write Filemarks” on page 179](#)
- [“Synchronize Buffers” on page 179](#)
- [“Query/Set Parameter” on page 179](#)
- [“Query/Set Position” on page 179](#)
- [“Query Encryption Status” on page 179](#)
- [“Display Message” on page 180](#)
- [“Display All Paths” on page 175](#)
- [“Report Density Support” on page 180](#)
- [“Test Encryption Path” on page 180](#)

Tape library-specific commands

- [“Element Information” on page 181](#)
- [“Position to Element” on page 181](#)
- [“Element Inventory” on page 181](#)
- [“Exchange Medium” on page 181](#)
- [“Move Medium” on page 181](#)
- [“Initialize Element Status” on page 181](#)
- [“Prevent/Allow Medium Removal” on page 182](#)
- [“Initialize Element Status Range” on page 182](#)
- [“Read Device IDs” on page 182](#)
- [“Read Cartridge Location” on page 182](#)

Service aid commands

- [“Configure TCP/IP Ports” on page 182](#)
- [“Dump/Force Dump/Dump” on page 183](#)
- [“Firmware Update” on page 183](#)

**Note:** When a command is issued in Tapeutil mode for ITDT GE, **Execute** must be pressed before the action takes place.

## ***Open***

When you select the **Open** command:

1. ITDT checks if a device is already opened.
2. Under **Device Name:**, enter the name of the device in the box.
3. In the **Open Mode** menu, select how to open the device (rw, ro, wo, append).
4. ITDT opens the device that you selected.

**Note:** Always use the Read Only mode when you are working with write-protected media.

## ***Close***

When you select the **Close** command

1. ITDT checks if the device is already closed.
2. ITDT closes the device.

## ***Inquiry***

When you select the **Inquiry** command

1. You are prompted for page code.
2. ITDT then displays a decoded format of a hexadecimal dump and prints a hexadecimal dump of the inquiry data.

### ***Test Unit Ready***

When you select the **Test Unit Ready** (TUR) command, ITDT issues the **Test Unit Ready** *ioctl* command.

### ***Reserve Device***

When you select the **Reserve Device** command, ITDT issues a reserve command for the device.

### ***Release Device***

When you select the **Release Device** command, ITDT issues a release command for the device.

### ***Request Sense***

When you select the **Request Sense** command

1. ITDT issues a **Request Sense** command.
2. ITDT then displays a decoded format of hexadecimal dump sense data and prints hexadecimal dump sense data.

### ***Log Sense***

When you select the **Log Sense** command

1. Enter the page number, in hexadecimal, in the **Page-Code** field.
2. ITDT issues a **Log Sense** command and outputs a hexadecimal dump of that page.

### ***Mode Sense***

When you select the **Mode Sense** command

1. Enter the page number, in hexadecimal, in the **Page-Code** field.
2. ITDT issues a **Mode Sense** command and outputs a hexadecimal dump of that page.

### ***Query Driver Version***

When you select the **Query Driver Version** command

1. ITDT issues the required command to get the driver version.
2. ITDT prints the driver version.

### ***Display All Paths***

When you select the **Display All Paths** command

1. ITDT issues an *ioctl* command.
2. ITDT outputs decoded path information for all paths.

### ***Rewind***

When you select the **Rewind** command, ITDT issues the *ioctl* rewind command for the device.

### ***Forward Space Filemarks***

When you select the **Forward Space Filemarks** command

1. Enter the amount of filemarks to forward space, in the **Filemark-Count** box.
2. ITDT issues (extrinsic) *ioctl* command.

3. The tape is positioned on the first block of the next file.

### ***Backward Space Filemarks***

When you select the **Backward Space Filemarks** command

1. Enter the amount of filemarks to backward space, in the **Filemark-Count** box.
2. ITDT issues (extrinsic) *ioctl* command.
3. The tape is positioned on the last block of the previous file.

### ***Servo Channel Calibration***

When you select the **Servo Channel Calibration** command:

1. ITDT issues the self diagnostic command for drive servo channel calibration.
2. The channel calibration returns PASSED or FAILED regarding the result of the command.

### ***Forward Space Records***

When you select the **Forward Space Records** command

1. Enter the amount of records to forward space, in the **Record-Count** box.
2. ITDT issues (extrinsic) *ioctl* command.

### ***Backward Space Records***

When you select the **Backward Space Records** command

1. Enter the amount of records to backward space, in the **Record-Count** box.
2. ITDT issues (extrinsic) *ioctl* command.

### ***Space to End of Data***

When you select the **Space to End of Data** (EOD) command, ITDT issues the (extrinsic) *ioctl* command.

### ***Read and Write Tests***

When you select the **Read and Write Tests** command, ITDT runs the following functions (Read and Write Test, Read Only Test, and Write Only Test). Three parameter fields have default values already in them. Next, a **Test** menu that gives you the option of **Read Data from Tape**, **Write Data to Tape**, and **Read/Write/Verify**.

**Note:** The default is a block size of 10240 bytes, a count of 20 blocks, and a repetition of 1. If the block size is zero, variable mode is used. With a fixed block size, a data amount of (block size \* blocks) is transferred with a single operation. This operation might get rejected if the total amount exceeds the transfer size of the system.

In *SCSI Generic mode*, the actual transfer size equals to **blocksize**. In *IBM Device Driver mode*, the transfer size is **blocksize \* blocks per read/write**. In either case, the blocksize must be equal to or less than the maximum Host Bus Adapter supported transfer size. The maximum supported transfer size for *SCSI Generic* is 1048576 Bytes and for *IBM Device Driver* is 500 MB.

The following steps are run, depending on which test is selected.

- The Read/Write steps:
  1. Issues a Read Position.
  2. Sets block size.
  3. Generates special pattern.
  4. Puts block id in bytes 0-3 of each block.
  5. Prints current block number, number of bytes and blocks.

6. Issues write command.
  7. Prints updated statistics.
  8. If number of bytes written is different from requested bytes to write, stop (go to Step 19).
  9. Writes two filemarks.
  10. Backward spaces two filemarks.
  11. Backward spaces number of records written.
  12. Prints amount of data to read.
  13. Issues read command.
  14. If read error occurred or number of bytes read is different from requested number of bytes to read, go to Step 19.
  15. Compares data that is read with data written, show mismatches and if mismatches exist, stop (go to Step 19).
  16. If compare is OK, print OK message.
  17. Forward space one file mark.
  18. Repeat Steps 10 - 24 until all blocks are written, or go to Step 4 until all blocks are written.
  19. Prints current block id and total number of bytes written.
- The Read Only steps:
    1. Issues a Read Position.
    2. Sets block size.
    3. Generates special pattern.
    4. Print amount of data to read.
    5. Issues read command.
    6. If read error occurred or number of bytes read is different from requested number of bytes to read, stop (go to Step 19).
    7. Compares data that is read with buffer data, show mismatches and if mismatches exist, stop (go to Step 19).
    8. If compare is OK, print OK message.
    9. Repeat Steps 10 - 15 until all blocks are written, or go to Step 4 until all blocks are written.
    10. Prints current block id and total number of bytes read.
    11. Backward spaces number of records written.
    12. Prints amount of data to read.
    13. Issues read command.
    14. If read error occurred or number of bytes read is different from requested number of bytes to read, go to Step 19.
    15. Compares data that is read with data written, show mismatches and if mismatches exist, stop (go to Step 19).
    16. If compare is Ok, print OK message.
    17. Forward space one file mark.
    18. Repeat Steps 10 - 24 until all blocks are written, or go to Step 4 until all blocks are written.
    19. Prints current block id and total number of bytes written.
  - The Write Only steps:
    1. Issues a Read Position.
    2. Sets block size.
    3. Generates special pattern.
    4. Put block id in bytes 0-3 of each block.

5. Prints current block number, number of bytes and blocks.
6. Issues write command.
7. Prints updated statistics.
8. If number of bytes written is different from requested bytes to write, stop (go to Step 10).
9. Repeat Steps 5 - 9 until all blocks are written, or to Step 4 until all blocks are written.
10. Print current block ID and total number of bytes written.

### **Read or Write Files**

When **Read or Write Files** is selected, a box under **File Name:** is where the path and name of the file is entered. Under that is a box named **Number of records to read (0 for entire file)**. The default amount in the box is 100. Next, a menu bar under **Test:** gives you the choice of **Read File from Tape** or **Write File to Tape**. Once the Test is selected, **Browse** appears next to the **File Name** box to allow browsing for the needed file. When you select the **Read or Write Files** command, ITDT runs the following functions:

- Read steps:
  1. Prompts if to read a file from tape
  2. You are prompted for destination file name
  3. You are prompted for number of records to read (If you press **Enter**, the entire file is read.)
  4. Prints the file name to be opened
  5. Opens the file (r/w with large file support, 666 umask)
  6. Issues **Query Parameters** *ioctl* command, if it fails, quit
  7. Sets blocksize to maximum, variable blocksize mode
  8. Calculates the number of blocks to read.
  9. Prints number of records to read.
  10. ITDT read from tape.
  11. Writes to file, stop if data count is not equal to data count requested.
  12. If more data to read, go to Step 10.
  13. Prints statistics.
- Write steps:
  1. Prompts if to write a file to tape.
  2. You are prompted for the source file name.
  3. Prints the file name to be opened.
  4. Opens the file (r/o with large file support).
  5. Issues **Query Parameters** *ioctl* command, if it fails, quits.
  6. Sets blocksize to maximum, variable blocksize mode
  7. Prints write statement.
  8. Reads from file.
  9. Writes to tape, stop if data counts written is not equal to data count requested.
  10. Prints statistics.

### **Erase**

When you select the **Erase** command, ITDT issues the (extrinsic) *ioctl* command.

### **Load Tape**

ITDT issues a SCSI **Load** command to load a tape.

## **Unload Tape**

When you select the **Unload Tape** command

1. ITDT issues the (extrinsic) *ioctl* command.
2. The tape rewinds and then unloads.

## **Write Filemarks**

When you select the **Write Filemarks** command

1. In the **Filemark-Count** box, enter the number of filemarks to write.
2. ITDT issues the (extrinsic) *ioctl* command.

## **Synchronize Buffers**

When you select the **Synchronize Buffers** command, ITDT issues the *ioctl* command.

## **Query/Set Parameter**

When you select the **Query/Set Parameter** command

1. ITDT shows the changeable parameters.  
**Note:** The list of changeable parameters are operating system specific. For a list of changeable parameters, refer to [Table 15 on page 136](#).
2. Select from the list of parameters that can be changed by clicking the choice.
3. ITDT requests prompt for parameter value (if required).
4. ITDT requests safety prompt (if required).
5. ITDT issues the *ioctl* command.

## **Query/Set Position**

When you select the **Query/Set Position** command

1. ITDT prints the current position and requests the new position.  
**Note:** ITDT does not distinguish between logical and physical position. It shows the current position and queries for the one to set, then sets the new position.
2. Enter the block id for where the tape must go. This block id must be entered in decimal. When the tape is set, the block id is printed in decimal with hexadecimal in parentheses.
3. ITDT issues the **Set Position** *ioctl* and returns the pass or fail results.
4. ITDT prints decoded logical position details.
5. ITDT issues **Query Physical Position** *ioctl* command.
6. ITDT prints decoded physical position details.
7. You are prompted for position to set (logical or physical) or to stop.
8. You are prompted for the block id in decimal or hexadecimal.
9. ITDT prints a summary.
10. ITDT issues the **Set Position** *ioctl* command.
11. Repeat steps 2 - 5.

## **Query Encryption Status**

When you select the **Query Encryption Status** command

1. ITDT issues **Get Encryption State** *ioctl* command.
2. ITDT displays encryption settings (Drive EC, Encryption Method, Encryption state).

## ***Display Message***

When you select the **Display Message** command

1. ITDT provides **Parameter** boxes in which you can enter 1 or 2 messages up to 8 characters.

**Note:** **Display Message** works only on drives that have a display pane, the 3590 and 3592 drives.

2. In the **Type:** menu, select which message (0 or 1) you want shown and if you want it to flash. There is also an alternate (alt) selection that alternates between messages.
3. ITDT issues the *ioctl* command.
4. ITDT prints the displayed message.

## ***Display All Paths***

When you select the **Display All Paths** command

1. ITDT issues an *ioctl* command.
2. ITDT outputs decoded path information for all paths.

## ***Report Density Support***

When you select the **Report Density Support** command

1. ITDT prints report status text for all supported media.
2. ITDT issues **Report Density Support** *ioctl* command to retrieve all supported media.
3. ITDT prints all requested reports. Data is printed in a decoded way. Scroll the screen to print each one of the following status texts:
  - Density name
  - Assigning organization
  - Description
  - Primary density code
  - Secondary density code
  - Write OK
  - Duplicate
  - Default
  - Bits per MM
  - Media Width
  - Tracks
  - Capacity (megabytes).
4. ITDT prints report status text for current media.
5. ITDT issues **Report Density Support** *ioctl* command to retrieve current media.
6. ITDT prints report data in a decoded way.

## ***Test Encryption Path***

When you select the **Test Encryption Path** command

**Note:** Not supported for the HP-UX operating system.

1. ITDT prints status message that server configuration and connections are tested.
2. ITDT issues the **Encryption Diagnostics** *ioctl* command, **Ping Diag.**
3. ITDT prints number of servers available or error message.
4. ITDT issues the **Encryption Diagnostics** *ioctl* command, **Basic Encryption Diag.**
5. ITDT prints completion code or error message.



6. ITDT issues the **Encryption Diagnostics** *ioctl* command, **Full Encryption Diag.**
7. ITDT prints completion code or error message.

### ***Element Information***

When you select the **Element Information** command:

1. ITDT issues the *ioctl* command.
2. ITDT shows
  - Number of robots
  - First robot address
  - Number of slots
  - First slot address
  - Number of I/E elements
  - First element address
  - Number of drives
  - First drive address

### ***Position to Element***

When you select the **Position to Element** command:

1. In the **Parameter** boxes, the Transport element address must be entered, in decimal (picker).
2. Insert the Destination element address in decimal.
3. ITDT issues the *ioctl* command.

### ***Element Inventory***

When you select the **Element Inventory** command:

1. ITDT issues the **Element Info** *ioctl* command.
2. ITDT issues the **Element Inventory** *ioctl* command.
3. ITDT displays decoded element inventory information.

### ***Exchange Medium***

When you select the **Exchange Medium** command:

1. Insert source address into the **Source** address box in Decimal.
2. Insert the first destination address in decimal in the **First destination address** box.
3. Insert the second destination address in decimal in the **Second destination address** box.
4. ITDT issues the *ioctl* command.

### ***Move Medium***

When you select the **Move Medium** command:

1. Insert source element address into the **Source** element address box in Decimal.
2. Insert the first destination element address in decimal in the **First destination element address** box.
3. Insert the second destination element address in decimal in the **Second destination element address** box.
4. ITDT issues the *ioctl* command.

### ***Initialize Element Status***

When you select the **Initialize Element Status** command:

1. ITDT issues the *ioctl* command.
2. ITDT prints the command summary.

### ***Prevent/Allow Medium Removal***

When you select the **Prevent/Allow Medium Removal** command:

1. Use the menu to either prevent or allow.
2. ITDT issues the *ioctl* command.

### ***Initialize Element Status Range***

When you select the **Initialize Element Status Range** command:

1. Insert the first slot address in decimal in the provided box.
2. Insert the number of slots in the provided box.
3. ITDT issues the *ioctl* command.

### ***Read Device IDs***

When you select the **Read Device IDs** command:

1. ITDT issues the **Element Info** *ioctl* command.
2. If no drive is present, ITDT prints NO DRIVE PRESENT and exits.
3. ITDT prints information for all drives.

### ***Read Cartridge Location***

When you select the **Read Cartridge Location** command:

1. You are prompted for address of the first element.
2. If address is zero, print the error message and exit.
3. You are prompted for the number of elements.
4. If the number of elements is zero, print the error message and exit.
5. ITDT issues the **Element Info** *ioctl* command.
6. ITDT verifies that the address range is valid. Otherwise, print the error message and exit.
7. If no slots are found in **Element Info** data; print the error message and exit.
8. ITDT issues the **READ\_CARTRIDGE\_LOCATION** *ioctl* command.
9. ITDT prints decoded storage element information.

### ***Configure TCP/IP Ports***

For LTO 5, TS1140, and later drives, the ethernet port settings can be configured with the **Configure TCP/IP Port** command. The **Configure TCP/IP Port** command displays the current settings:

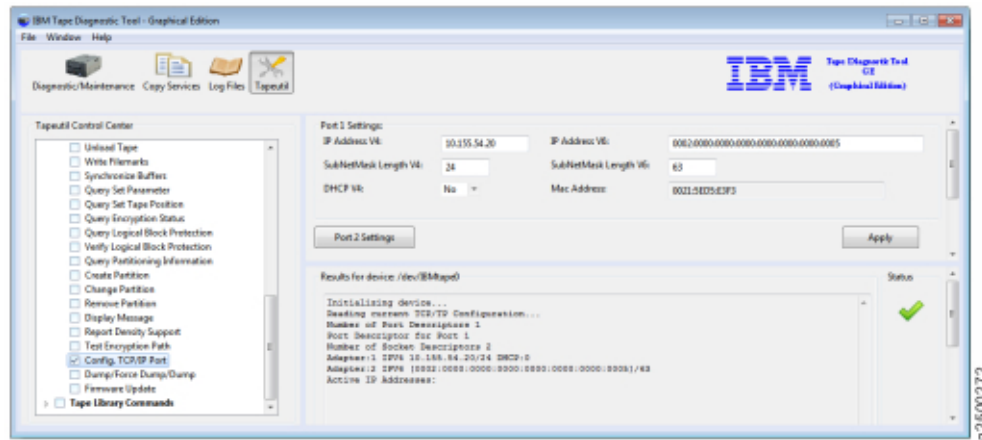


Figure 60. Configure TCP/IP Ports command in the Graphical Edition

After you click **Apply**, the new values are set and the updated values display.

**Note:** Because earlier drive generations do not have an ethernet port, the **Configure TCP/IP Ports** command is rejected for these devices with the following message:

TCP/IP configuration is not supported on this product.

### **Dump/Force Dump/Dump**

When you select the **Dump/Force Dump/Dump** command:

1. ITDT retrieves the dump.
2. ITDT issues the **Force Dump** command.
3. ITDT retrieves the second dump.
4. ITDT displays the name of stored dump files and the output directory where they are stored. The dump filenames start with the serial number of the device.

### **Firmware Update**

When you select the **Firmware Update** command, browse to the microcode file to be used. ITDT runs the firmware update and displays progress status and result.

The following site is available for the latest firmware files: <http://www.ibm.com/support/fixcentral/>. Select **System Storage > Tape Systems > Tape autoloaders and libraries** or **Tape drives**.

## **Verifying correct attachment of your devices**

Before you start to use your devices for production work with your applications, or if you encounter difficulties with your devices, you might want to verify that the hardware, connections, and device drivers are working together properly. Before you can do this verification, you must do the following procedure -

1. Install your hardware as indicated in the appropriate hardware manuals.
2. Power On your hardware and verify that the hardware is functioning properly by running commands according to the product documentation. See "IBM Tape Product Publications" on page xiii.
3. Attach your hardware to the host system as indicated in the appropriate hardware manuals and as indicated in the appropriate chapters from this manual.
4. Start your operating system as indicated in the appropriate chapters from this manual.
5. Log in to the operating system as Administrator.
6. If device drivers are used by your device other than the ones documented in this manual, disable the other device drivers, and install or enable the drivers that are documented in this manual.
7. Start ITDT (for instructions see "Starting ITDT - Standard Edition" on page 82).

8. Scan for devices. Any devices that show up are properly attached.

## Platform-specific help

There is a problem determination section for each platform.

- AIX: [“Problem determination” on page 30](#)
- Linux: [“Problem determination” on page 62](#)
- Windows: [“Problem determination” on page 73](#)

## IBM technical support

If the problem persists after these procedures are followed, it is possible that an unexpected condition occurred in the driver’s environment. In this case, contact your IBM service representative (1-800-IBM-SERV) and provide the following information to help IBM re-create and resolve the problem:

1. Machine type and model of your IBM tape product
2. Specific driver version
3. Description of the problem
4. System configuration
5. Operation that was running at the time the problem was encountered

## Managing the microcode on the IBM tape drive

---

Microcode is computer software that is stored in nonvolatile storage on your tape device or library hardware. It controls the operation of your hardware. When your tape device or library hardware was manufactured, a microcode load was installed and shipped with your device.

If you are having trouble with your hardware, IBM service personnel ask what level of microcode you have on your hardware. If they believe that you need a new level of microcode, they might instruct you to install a newer level of microcode on your hardware. They can provide you with updated microcode.

You can query the current level of microcode by issuing commands on the front panel of your hardware. Consult the appropriate hardware reference manual for specific instructions on querying your microcode level.

If your device is connected to a host system that has device or library support, you can also query the last 4 digits of the current level of microcode with software. Refer to [“IBM Tape Diagnostic Tool \(ITDT\)” on page 76](#). The unit must be powered On, configured properly, and ready. For information, refer to the appropriate chapter in this document (based on the operating system/platform) for details on how to have the device ready.

The following instructions are a guide to install another version of microcode on a tape drive.

1. Ensure that the tape drive is connected to a host system and that the tape device driver is powered-On and configured properly with no tape cartridge in the drive. Follow the instructions in [“Verifying correct attachment of your devices” on page 183](#) to ensure that the drive is configured properly and ready.
2. Open ITDT and follow the instructions for downloading microcode. These instructions are in both the SE and the GE versions. In SE, it is available in all sections; scan menu under Firmware update, the tape utility (71) section, and the scripting (**ucode**) command.

## Notices

---

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries (or regions) in which IBM operates.

Any references to an IBM program or other IBM product in this publication is not intended to state or imply that only IBM’s program or other product may be used. Any functionally equivalent program that

does not infringe any of IBM's intellectual property rights may be used instead of the IBM product. Evaluation and verification of operation in conjunction with other products, except those expressly designed by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You may send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country (or region) where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states (or regions) do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement cannot apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes are incorporated in new editions of the publication. IBM may make improvements and/or changes in the products and/or programs described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

The ITDT-SE and ITDT-GE software uses Henry Spencer's regular expression library that is subject to the following copyright notice:

"Copyright 1992, 1993, 1994, 1997 Henry Spencer. All rights reserved. This software is not subject to any license of the American Telephone and Telegraph Company or of the Regents of the University of California.

Permission is granted to anyone to use this software for any purpose on any computer system, and to alter it and redistribute it, subject to the following restrictions:

1. The author is not responsible for the consequences of use of this software, no matter how awful, even if they arise from flaws in it.
2. The origin of this software must not be misrepresented, either by explicit claim or by omission. Since few users ever read sources, credits must appear in the documentation.
3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software. Since few users ever read sources, credits must appear in the documentation.
4. This notice cannot be removed or altered.

## Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries (or regions), or both:

AIX	IBMLink	RS/6000	System z
AIX 5L	Magstar	S/390	Tivoli
FICON®	Micro Channel	StorageSmart	TotalStorage
HyperFactor	Netfinity	System i	Virtualization Engine
i5/OS	POWER5	System p	xSeries
iSeries	ProtecTIER®	System Storage	z9
IBM	pSeries	System x	zSeries

Adobe and Acrobat are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Itanium, and Pentium are trademarks of Intel Corporation in the United States, other countries (or regions), or both.

Java and all Java-based trademarks are trademarks of Oracle, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and Windows 2000 are trademarks of Microsoft Corporation in the United States, other countries (or regions), or both.

UNIX is a registered trademark of The Open Group in the United States and other countries (or regions).

Other company, product, and service names may be trademarks or service marks of others.

## Terms and conditions for IBM Documentation

---

Permissions for the use of these publications are granted subject to the following terms and conditions.

### IBM Privacy Policy

We intend to protect your personal information and to maintain its integrity. IBM implements reasonable physical, administrative, and technical safeguards to help us protect your personal information from unauthorized access, use, and disclosure. For example, the products only provide IBM data about the asset usage and configuration and do not reflect private use of the asset. When diagnostics are required to be sent to IBM, and a problem is submitted, that data is routed directly to a secured infrastructure. Only individuals with a need to know are given access while working to resolve your problem. When appropriate, we also require that our suppliers protect such information from unauthorized access, use, and disclosure.

Visit the IBM Privacy Policy for additional information on this topic at <https://www.ibm.com/privacy/details/us/en/>.

### Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

### Personal use

You can reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You cannot distribute, display, or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

### Commercial use

You can reproduce, distribute, and display these publications solely within your enterprise provided that all proprietary notices are preserved. You cannot make derivative works of these publications, or

reproduce, distribute, or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

## **Rights**

Except as expressly granted in this permission, no other permissions, licenses, or rights are granted, either express or implied, to the Publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions that are granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or as determined by IBM, the above instructions are not being properly followed.

You cannot download, export, or reexport this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

## **IBM trademarks**

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available at <http://www.ibm.com/legal/copytrade.shtml>.

---

# Index

## A

about data encryption [9](#)  
Accessing [77](#)  
ACTRC utility [36](#)  
Adding or removing devices [68](#)  
AIX (Atape) [4](#), [12–16](#), [21–25](#), [27–32](#), [34–36](#), [38](#), [39](#)  
AIX device parameters [39](#)  
Alternate pathing [16](#)  
Archive mode unthread (AMU) [21](#)  
ATRC utility [36](#)  
autoloading [16](#)  
automatic dump facility for 3590 and Magstar MP tape drives [34](#)  
Automatic failover [5](#)

## B

block size [16](#)  
buffered mode [16](#)  
bulk rekey [11](#)

## C

capacity scaling [21](#)  
changeable parameters [48](#)  
Checklist [10](#)  
common utilities [39](#)  
component tracing [36](#)  
components created during installation [41](#)  
compression [16](#)  
configuration [30](#)  
configuration parameters [46–48](#)  
Configuration parameters [16](#)  
configuring and unconfiguring path failover support [28](#)  
configuring and unconfiguring primary and alternative devices [29](#)  
configuring tape and medium changer devices [14](#)  
Configuring Tape and Medium Changer devices [43](#), [44](#)  
connectivity [30](#)  
Control path failover [25](#)  
conventions used [41](#)  
create an FMR tape [36](#)

## D

Data flow [12](#), [40](#)  
data path [38](#)  
data path failover [4](#)  
Data path failover [27](#)  
Deconfiguring the 3490E, 3590, Magstar MP, or 7332 tape device [15](#)  
Deconfiguring the 3575, 7331, 7334, 7336, or 7337 medium changer device [15](#)  
Detail data [32](#)  
detailed description [36](#)

Device and volume information logging [31](#), [32](#)  
Device driver configuration [29](#)  
Device driver management [68](#)  
Devices not reported [67](#)  
disable procedure [68](#)  
drive dump [30](#)  
Dump device commands [30](#)  
Dump support [30](#)  
Dynamic Runtime Attributes [35](#)

## E

EKM server logs [30](#)  
emulate autoloader [16](#)  
encryption [4](#)  
error labels [32](#)  
error log analysis [36](#)  
error log templates [32](#)  
error logging [30](#), [32](#)

## F

fail degraded media [16](#)  
field support [30](#)  
force microcode dump [36](#)

## G

general information [9](#)  
GKLM [11](#)  
Guardium Key Lifecycle Manager [11](#)

## H

hardware requirements [68](#)  
Hardware requirements [13](#), [40](#), [67](#)

## I

Installation and configuration instructions [13–16](#), [40–45](#), [68](#)  
installation overview [68](#)  
installation procedure [42](#)  
Installation procedure [14](#)  
installation procedures [68](#)  
Introduction and product requirements [4](#), [40](#), [67](#), [76](#)  
iostat utility [39](#)  
ITDT [76](#), [77](#)

## L

legal  
    terms and conditions [186](#)  
library requirements [9](#)  
library-managed encryption  
    planning [11](#)  
Linux (lin\_tape) [40–48](#), [51](#), [67](#)



Linux system-managed encryption [61](#)

Load balancing [4](#)

log file [13](#)

logging [16](#)

logical write protect [21](#)

## M

managing encryption [9](#)

Managing microcode on the tape drive [184](#)

maximum size of the log file [16](#)

media parameters [21](#)

microcode load [36](#)

## N

new logical name [16](#)

nonchangeable parameters [47](#)

## P

path failover [4](#)

Path failover [4](#), [5](#)

Path failover support [28](#), [29](#)

Performance considerations [38](#), [39](#)

Persistent naming support [24](#)

preinstallation considerations [14](#)

primary and alternative paths [28](#)

Problem determination [30](#), [32](#), [34–36](#)

Product requirements [12](#), [13](#), [40](#), [67](#)

Product requirements and compatibility [68](#)

Purpose [4](#), [12](#), [76](#)

## Q

querying drive configuration [29](#)

querying installed package [43](#)

querying primary and alternate path configuration [28](#)

## R

read dump [36](#)

read error recovery time [16](#)

record space mode [16](#)

removal procedure [68](#)

requirements

library [9](#)

tape [9](#)

Reservation conflict logging [32](#)

reservation key [16](#)

reservation type [16](#)

reset drive [36](#)

retain reservation [16](#)

rewind immediate [16](#)

## S

SCSI status busy retry [16](#)

sense data [30](#)

server logs [30](#)

SKLM [11](#)

SMIT panels [29](#)

software requirements [68](#)

Software requirements [13](#), [40](#)

Special files [22](#), [23](#), [51](#)

Special files for 3490E, 3590, Magstar MP, or 7332 tape devices [22](#)

Special files for 3575, 7331, 7334, 7336, or 7337 medium changer devices [23](#)

Special files for medium changer device [51](#)

Special files for tape device [51](#)

supported hardware [4](#)

system encryption [16](#)

system encryption for Write commands [16](#)

System p [44](#)

System z [44](#)

system-managed encryption [61](#)

System-managed encryption [29](#)

## T

Tape drive service aids [36](#)

Tape drive, media, and device driver parameters [16](#), [21](#), [46–48](#)

tape log utility [31](#)

testing data encryption configuration [30](#)

trace facility [35](#)

trailer labels [16](#)

## U

uninstall procedure [16](#), [45](#)

uninstalling the device drivers [68](#)

updating procedure [43](#)

## V

Verifying device attachment [183](#)

volume ID for logging [21](#)

## W

Windows (IBMtape) [68](#)

Windows device driver [67](#), [68](#)

Windows NT [68](#)

