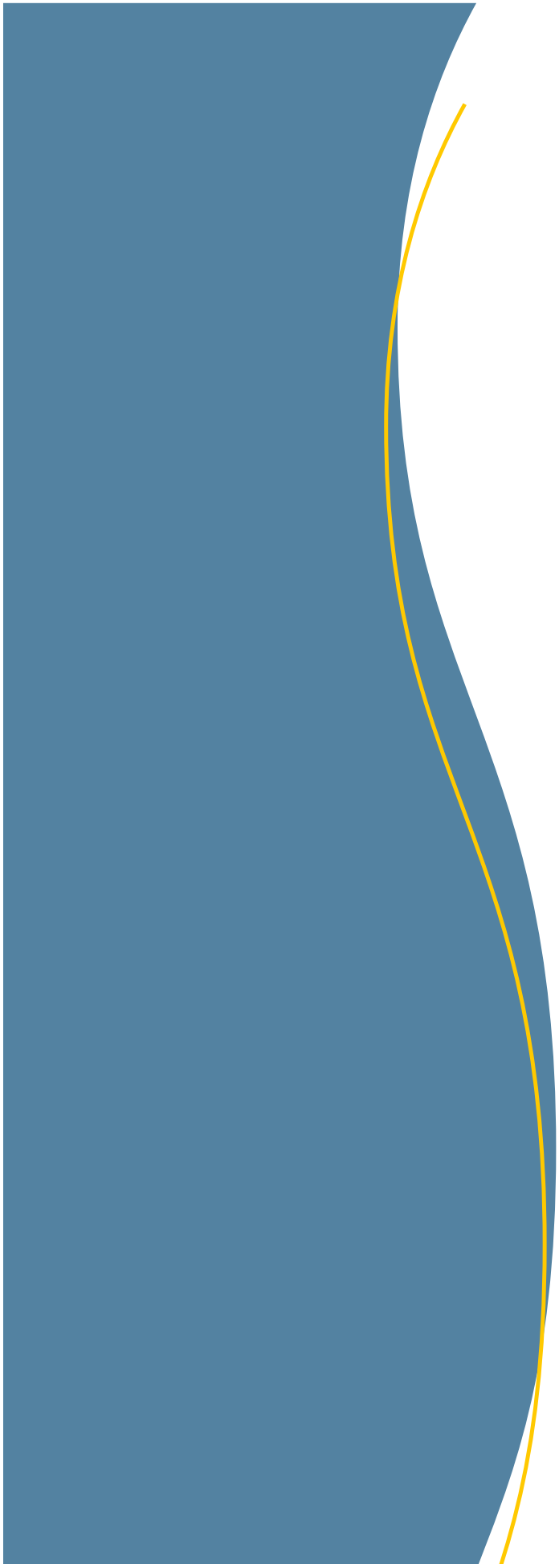


StorageTek Crypto Key Management Solution

Version 2.0

Open Systems Implementation Practices

White Paper
August 2008

We welcome your feedback. Please contact Sun StorageTek Test Engineering at:

greg.drobish@sun.com; martha.sammartano@sun.com; warner.hersey@sun.com

or

Sun StorageTek Open Systems Test Engineering
Sun Microsystems, Inc.
500 Eldorado Blvd.
Broomfield, CO 80021
USA

Contents

Contents **3**
Change History **5**
Audience **5**
Related Publications 5
Introduction **6**
Chapter 1: Product Overview **7**
What is it? 7
Encryption Components 7
How does it work? 8
Benefits of the Sun Encryption Solution 9
Chapter 2: Encryption Implementations **10**
Things to Consider 10
KMA Deployment 10
Mixing Drives 10
Simplified Implementation 11
Why Mixed Encryption? 11
Chapter 3: Integration with Tier 1 Backup Applications **12**
Sample Mixed Encryption Configuration 12
NetBackup Mixed Encryption Implementation 13
Media Management Strategy 14
Operations within a Mixed Encryption Configuration 16
TSM Mixed Encryption Implementation 17
Media Management strategy 19
ACSLS and Native TSM Considerations 20
Gresham EDT in a TSM Mixed Encryption Implementation 22
TSM Solution performance considerations 23
Applying Mixed Encryption Principles to Alternative Environments 24
Primary Components of a Mixed Encryption Solution 24
Chapter 4: Legacy to New Generation Drive Migration **26**
Compatibility Rules 26
Migration Strategies 26
Chapter 5: Optimizing Encryption Solution for Redundancy **28**
Redundant Network 28
Multi-Site Network 30
Chapter 6: T10000 Tape Drive Characterization **31**
Characterization Overview 31
Data Buffer Transfers 31

Lab Performance Data 32
Small Data Buffers 32
Large Data Buffers 33
Characterization Data Charts 34
The Benefit of Multiplexing 36
Using Multiple Write Drives 36
Chapter 7: T9840D Tape Drive Characterization 37
Characterization Overview 37
Data Buffer Transfers 37
Lab Performance Data 37
Default Size Data Buffers 38
Maximum Size Data Buffers 39
Characterization Data Charts 39
The Benefit of Multiplexing 41
Using Multiple Write Drives 41
Chapter 8: HP LTO4 Tape Drive Characterization 43
Characterization Overview 43
Data Buffer Transfers 43
Lab Performance Data 43
Small Data Buffers 44
Large Data Buffers 45
Characterization Data Charts 45
The Benefit of Multiplexing 48
Using Multiple Write Drives 49
Chapter 9: T10000B Tape Drive Characterization 50
Characterization Overview 50
Data Buffer Transfers 50
Lab Performance Data 50
Small Data Buffers 51
Large Data Buffers 52
Characterization Data Charts 52
The Benefit of Multiplexing 54
Using Multiple Write Drives 55

Change History

Document Description	
Document owner	Greg Drobish – Systems Integration Engineer
Organization	Sun StorageTek Test Engineering – Open Systems Customer Emulation Test

Revision	Date	Description
V1.0	03/14/2008	Initial draft
V1.1	03/18/2008	First revision incorporating comments and feedback
V1.2	03/26/2008	Initial release describing open system implementation practices using T10000 encryption drive
V2.0	06/30/2008	Second release incorporating discussion of migration from legacy to new generation tape drives and characterization data for T9840D encryption drive
V2.1	07/03/2008	Third release adding characterization data for HP LTO4 encryption drive
V2.2	08/04/2008	Fourth release updating characterization data for HP LTO4 encryption drive and adding characterization data for T10000B encryption drive

Audience

This documentation is intended for Sun employees, field personnel, partners, and customers who are interested in learning more about the Sun StorageTek encryption solution. Intended audience are those who are already familiar with the information contained within the administration and installation guide.

Related Publications

The following publications provide additional information about specific topics relating to the use of the Key Management System (KMS):

- Key Management System (KMS) 2.0 Installation and Service Manual
- Key Management System (KMS) 2.0 Administration Guide
- Key Management System (KMS) 2.0 Systems Assurance Guide
- Key Management System (KMS) 2.0 Management Practices White Paper

Introduction

The purpose of this document is to provide implementation practices to be considered when installing the StorageTek Crypto Key Management Solution in an open systems environment. Emphasis is on integration with tier 1 backup applications, product performance characterization, implementing a mixed encryption configuration and optimizing the encryption solution for redundancy. This is not intended to be a standalone document. It is designed to be used as a supplement to the systems assurance and installation guides. As such, it is expected that the reader is familiar with these documents and information deemed redundant will not be covered here.

This white paper is not intended to be a step-by-step guide, but rather will serve to highlight the issues and obstacles involved in a typical open systems implementation and present recommended practices for overcoming them.

The information presented in this document is supported by testing initiatives conducted in Sun's Customer Emulation Test Lab.

Chapter 1: Product Overview

What is it?

The StorageTek Crypto Key Management System is a device-based encryption solution. Data is encrypted at rest on the storage medium. Initial offerings will support the enrollment of several tape drive models capable of acting as an encryption agent in this solution.

Encryption Components

Key Management Appliance (KMA) – A secure, dedicated appliance for creating, managing and storing encryption keys. It delivers policy-based lifecycle key management and ensures the security and authenticity of the encryption solution. The KMA is developed on Sun X2100-M2 server hardware embedded with a Sun SCA6000 security card.

KMS Cluster – A Key Management System is comprised of multiple KMAs clustered together. Key management appliances are clustered to provide failover, load balancing and data protection. KMAs in a cluster act in an active/active manner. All KMAs can provide full functionality to any encryption agent and changes made on any one KMA are automatically replicated to all KMAs within the cluster.

KMA Manager Software – The key management appliance is a locked down, security hardened device. There are only very limited options available to a privileged security officer through the console or ELOM (enhanced lights out manager) of the appliance. Other than specific setup operations, administration of the StorageTek Crypto Key Management Solution occurs through a GUI management program that is executed from a customer-provided workstation or server.

Encryption Agent – Encryption agent is a generic term for the storage peripheral device that is used by the KMS to manage encrypted data. At the time of release, the Crypto Key Management 2.0 System supports only the StorageTek T10000 tape drive. In subsequent releases, T10000B, 9840D and LTO4 encryption agents will be enhanced to operate with KMS 2.0. The T10000 tape drive has a native transfer rate of 120 MB/sec and has demonstrated speeds up to 330 MB/sec using compression. The drive utilizes a 4Gb FC interface. Standard T10000 media have a 500GB uncompressed capacity with a shorter “Sport” cartridge at 120GB. WORM VolSafe media is available in both 500GB and 120GB uncompressed capacity format.

How does it work?

Encryption serves to limit access to data by making information unreadable without special knowledge. This is accomplished by applying a cryptographic algorithm called a cipher to data. The result is an encrypted ciphertext that is unreadable until an inverse algorithm is again applied to decrypt the data. This requires access to the key value used to encrypt the data. Data is transported and stored in this unreadable state, thus achieving data security when information is most vulnerable.

Sun's encryption solution utilizes an AES(Advanced Encryption Standard)-256 substitution-permutation network cipher algorithm that is applied by the storage peripheral device, in the example below, the T10000 tape drive. CCM-AES is the mode employed by this solution. This is a FIPS (Federal Information Processing Standard) compliant encryption standard. Key management occurs outside of the data path and the encryption of sensitive data is completely transparent to the backup or archiving application. A cluster of KMAs manage these encrypting devices by an authenticating enrollment process, securing and authenticating the distribution of encryption keys and providing a policy-based lifecycle key management solution. The active/active cluster serves to provide failover, load-balancing and data protection by replicating changes across the cluster in real time. KMA to drive communications occur over an isolated secured network in recommended configurations. Encryption keys are never in the clear, even during delivery over a secured network.

Administration of the encryption solution is performed via the KMS manager GUI that is installed on workstations or management servers. Separation of roles and responsibilities are customized to meet the needs of the organization and a quorum is created to govern critical operations such as adding a new KMA to the cluster. Key policies, key groups and agent assignments are defined through the manager GUI and enable the automated management of encryption keys throughout the lifecycle of the data being encrypted.

When a tape cartridge is mounted, the encryption agent requests the appropriate encryption keys from the KMS cluster. Any KMA in the cluster is capable of providing all necessary functions to any drive enrolled in the cluster. Keys are transferred to the encryption agent and are used for writing and reading of the data. A different write key is issued for each tape. The KMA database keeps track of all keys used on a tape and supplies the keys automatically when the tape is mounted. The expiration period of an encryption key depends on policy-based settings that were defined through the KMS manager GUI. When a tape is loaded in an encrypting drive after the key's encryption period has expired, a new encryption key is generated and issued.

The illustration below depicts a logical sample environment that consists of a two KMA cluster servicing multiple automated tape libraries.

Figure 1: Sun's Encryption Solution

Benefits of the Sun Encryption Solution

Performance: Data is encrypted after tape compression occurs which allows for maximum performance and the most efficient possible usage of tape media. Sun utilizes a very powerful encryption algorithm (AES-256) that is also highly efficient. As such only a 100 byte overhead is required for each block of encrypted data that is recorded. Most backup / restore or archiving applications realize maximum performance when utilizing an average blocksize between 256KB and 1MB. The T10000 tape drive supports blocksizes up to 2MB. Given this, the impact of encryption on performance is negligible. This gives Sun a significant advantage over competitor products where encryption overhead, processing strain and inability to realize maximum compression by encrypting before the data reaches the storage device all contribute to drastic performance degradation.

Ease of Management: The KMS software manager GUI provides a central point of administration for a scalable encryption solution that can grow to manage multiple libraries of encrypted drives in multiple locations. Powerful policy-based lifecycle management options allow for intuitive and automated administration of encryption keys.

Security: AES-256, which Sun utilizes as a block cipher algorithm, is the most powerful commercially available security algorithm and Sun's implementation has been validated by NIST – the US Government National Institute of Standards and Technology. The key management appliance is a locked down, security hardened device. Separation of roles and responsibilities allows for a system of checks and balances to be implemented. Quorum operations are required for changes to the configuration that could pose a security risk.

Chapter 2: Encryption Implementations

Things to Consider

Several things to consider when planning a data encryption implementation are laid out below.

KMA Deployment

One KMS cluster is capable of administering an encryption solution for an entire organization even if that organization contains datacenters in multiple locations around the world. Using one cluster introduces a single point of administration and serves to increase ease of management and simplicity of the solution. To achieve maximum failover, load balancing and redundancy, it is recommended that a minimum of two KMAs be deployed per location. For instance, if an organization employs an encryption solution across datacenters in Dallas and Boston, they would include two KMAs at each location. All four KMAs would be clustered together over the wide area network and would continually replicate any changes across the cluster. The KMA's at each location connect to the drives at that location using the private network within the library. If the Sun Service Delivery Platform is deployed, it also attaches to that private network. An encrypting tape drive in the Dallas site would request encryption keys from either of the two KMAs at that location over the isolated services network.

Mixing Drives

Key management occurs outside of the data path. The solution is application transparent and because of this, backup and archiving applications do not have to be aware that encryption is taking place. Though there is a change in the drive inquiry string that occurs when a drive is enabled for encryption, backup applications currently do not differentiate between an encrypting and non-encrypting tape drive of the same model. This is an important point to consider when planning an encryption implementation for the following reasons:

- Encrypting drives can read cartridges that contain non-encrypted data but they cannot append to them.
- No drive can read encrypted data or append to an encrypted cartridge without the proper read key.

This is an important thing to consider when planning an encryption implementation. Encrypted drives can read cartridges that contain non-encrypted data but they cannot append to them and no drive can read encrypted data or append to an encrypted cartridge without the proper read key.

In light of the considerations outlined in the above section, the most simple and straightforward configuration is to not mix encrypting and non-encrypting tape drives of the same model in the same library. For instance, a configuration including an automated tape library that uses only encrypting T10000 drives would not require any special considerations during implementation. Likewise, an automated library that uses a mix of encrypting T10000 and non-encrypting 9840D would also not require any special consideration during implementation. The illustration below shows a sample environment that only includes encrypting T10000 tape drives.

Simplified Implementation

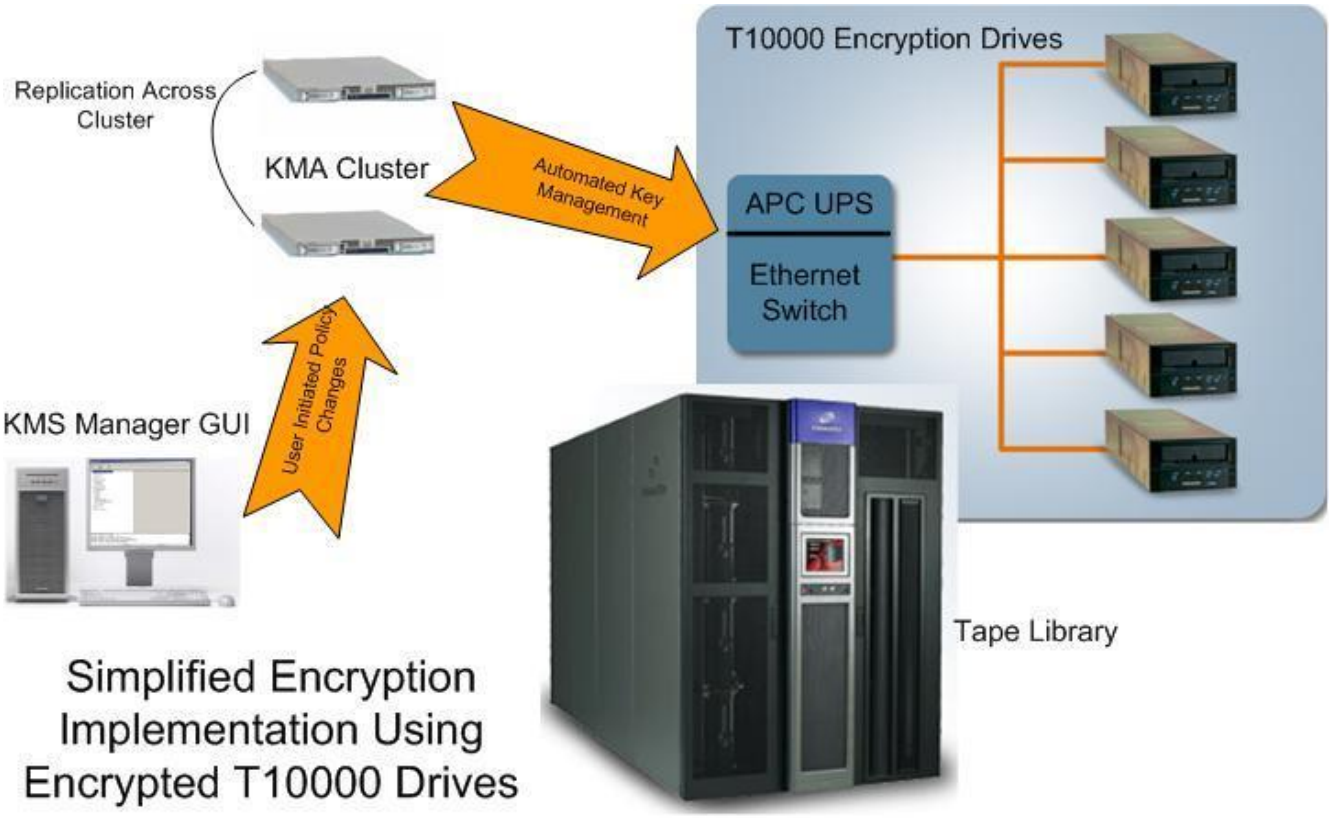


Figure 2: Simplified Encryption Implementation

Why Mixed Encryption?

Simplified encryption implementations will work well for most customers, but we recognize that it will not be the best solution for all customers. For a variety of reasons, some customers have tape environments that cannot be adjusted and they have needs that cannot be met by the configurations laid out above. It is for these reasons that this document addresses mixed encryption strategies to employ when implementing a configuration with tape drives of the same model in encrypting and non-encrypting modes in the same automated tape library.

By mixing encryption in a single library, greater levels of flexibility can be attained and specific customer needs can be met that may not be realized with standard encryption configurations. However, mixed encryption solutions require a more complex configuration and careful management. For those customers where the benefits of mixed encryption are required, the next chapters will highlight strategies and recommended practices to consider when planning a mixed encryption solution.

Chapter 3: Integration with Tier 1 Backup Applications

In a mixed encryption configuration, tape drives of the same drive model reside in the same logical library partition and operate in both encrypting and non-encrypting modes.

This presents a challenge because the encryption management occurs outside of the data path and the encryption process is transparent to applications that utilize the tape drive. Even though there is a change in the drive inquiry string when it is enabled for encryption, applications do not differentiate between a drive operating in encrypting or non-encrypting mode.

Problems can arise when encrypted and non-encrypted drives share the same set of media. In default installations, most applications will indiscriminately use any available tape drive for media operations and failures will occur when attempting to read encrypted data using a non-encrypting drive or when trying to write with a non-encrypting drive to an encrypted cartridge, etc.

These issues can be overcome by customizing storage architecture and media management policies through the backup application. Strategies for architecting an effective mixed encryption environment are outlined in the remainder of this chapter. Symantec Veritas NetBackup, Tivoli Storage Manager and Sun StorageTek Enterprise Backup Software were tested as tier-1 backup applications in our lab.

Sample Mixed Encryption Configuration

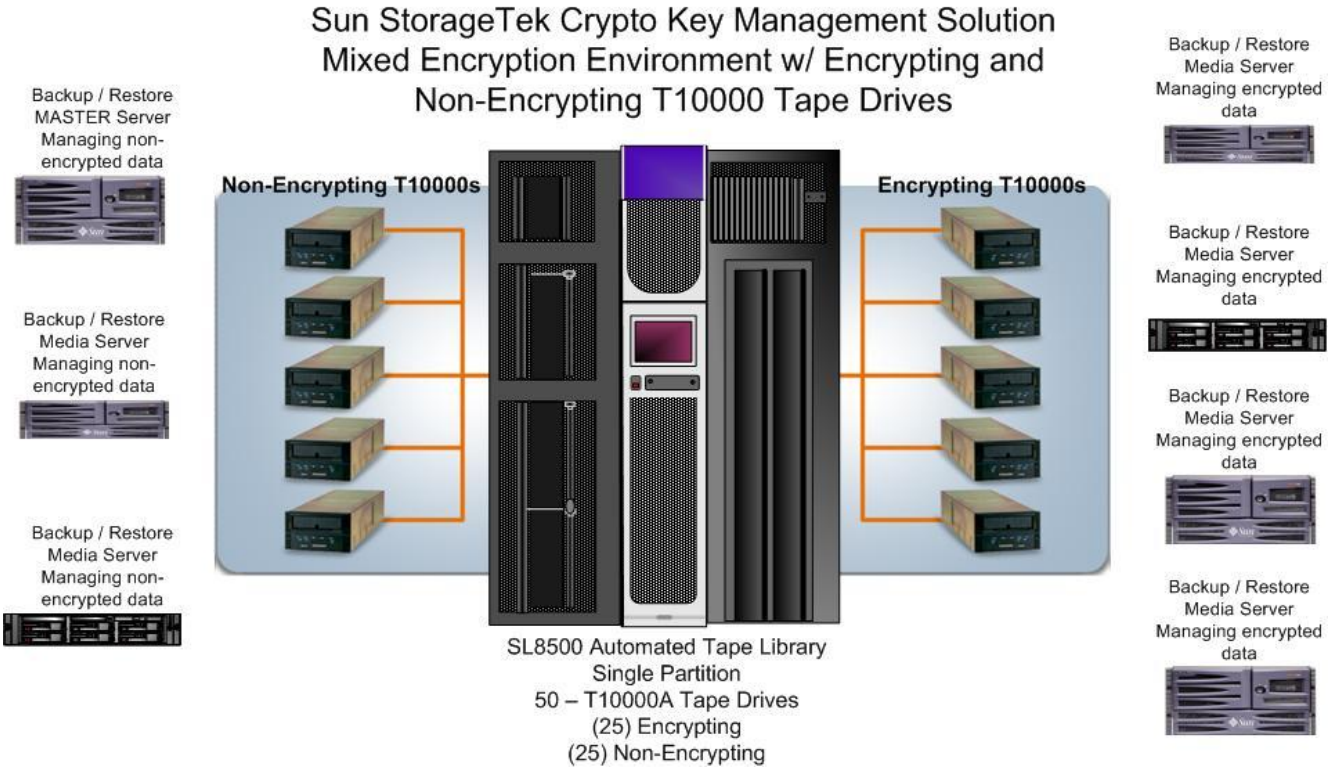


Figure 3: Sample Mixed Encryption

In the above sample mixed encryption diagram, T10000 drives are housed in the same logical library partition and operate in both encrypting and non-encrypting modes. This creates a mixed encryption environment and will result in uncertainty in whether a tape is loaded in an encrypting or a non-encrypting drive. This situation can be managed by placing encrypting T10000s in a separate logical partition on the SL8500 library from the non-encrypting T10000s or by utilizing different drive types such as T10000 for encrypting operations and 9840D for non-encrypting functions.

NetBackup Mixed Encryption Implementation

NetBackup operates by assigning policies (requests to backup a set of files from a particular client) to storage units (storage resources that are used to complete the request). Storage units can be tape or disk. Tape storage units are defined by the media server that is handling the request and the type of tape drives that are available to that particular media server. For instance if a storage unit is created for *Server-1* of type LTO, a request assigned to this storage unit could be fielded by any available LTO tape drive that is accessible from *Server-1*.

It is essential that the NetBackup mixed encryption configuration be architected so that non-encrypting and encrypting drives do not share the same storage unit.

The recommended way of accomplishing this is by segregating the T10000 tape drives among media servers. Tape drives are zoned in the fiber channel fabric to be available to some media servers but not to others. Encrypting drives can be zoned to one group of servers while non-encrypting drives can be presented to another group of servers. Media server segregation is the recommended method of customizing a mixed encryption environment for NetBackup because it creates an extra layer of protection and it reduces the management tasks that would be required with other methods. Utilizing this method requires customization of the backup environment at the time of implementation but ongoing tasks such as drive replacement will not require media type changes and can utilize the default media types that NetBackup assigns to it.

Figure 4 shows our previous mixed encryption example now customized for a Veritas NetBackup environment. Media servers on the right handle all encrypting media operations and media servers on the left handle all non-encrypting functions. The Media / Master server in the top left administers the entire NetBackup configuration. Depending on whether a client needs an encrypted or non-encrypted security level, a policy will be defined for the client utilizing a media server from the appropriate group.

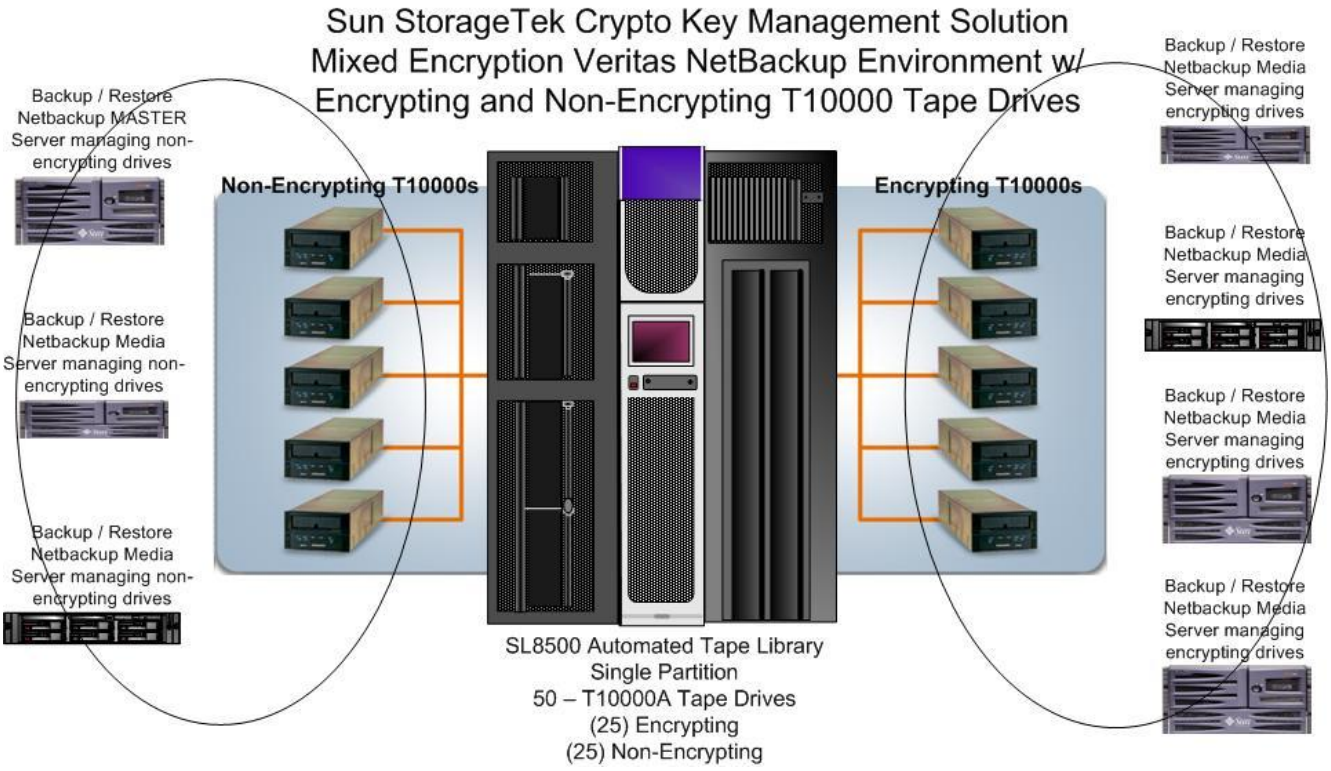
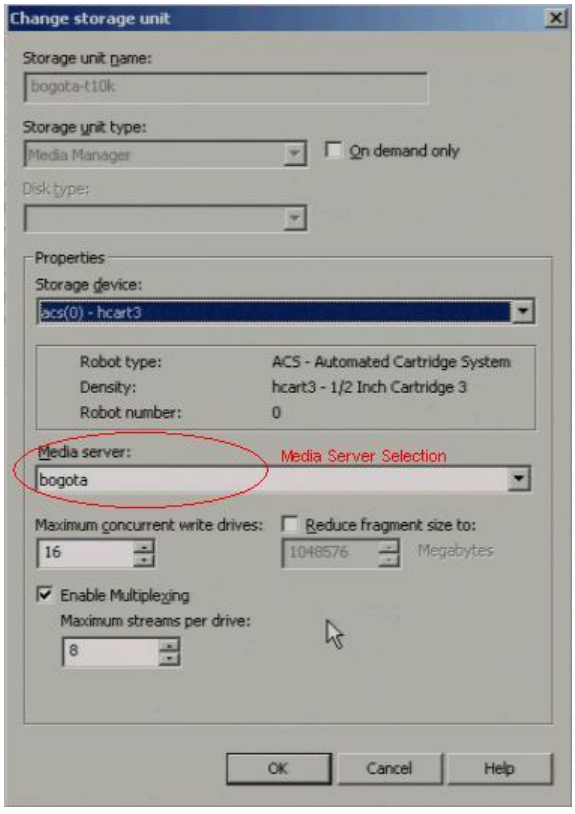


Figure 4: NetBackup Mixed Encryption

A storage unit is created for each drive type and media server combination. In this example, each media server has one storage unit. Here is what a sample storage unit utilizing T10000 tape drives looks like in this mixed encryption configuration. Notice that the storage unit is configured to use ACS robot type in the ACSLS-managed SL8500 tape library.



Media Management Strategy

A good media management strategy is a key piece to a successful NetBackup mixed encryption solution. Encrypting and non-encrypting drives cannot share media from the same pool. A media management strategy is required to support mixed encryption in NetBackup.

Defining Volume Pools

NetBackup configurations default to a single volume pool from which to draw media. A tape backup policy request will look through the default volume pool for an unassigned media cartridge of the same type as the tape drive defined in the storage unit. Once one is found, it will be allocated to the request and mounted in the tape drive.

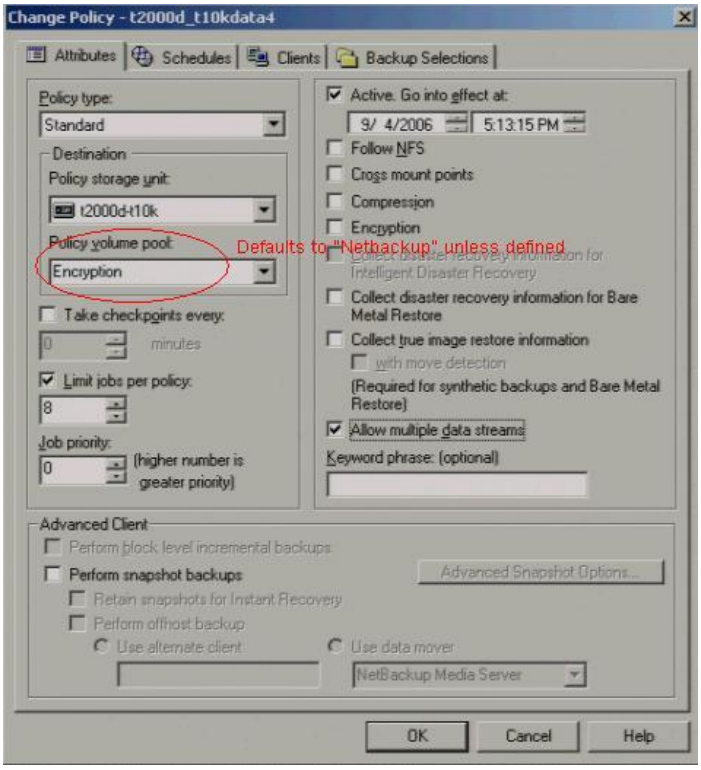
For mixed encryption environments it is necessary that multiple volume pools be created to keep track of media and ensure that encrypted cartridges are not used in non-encrypting drives and vice-versa. For the above example, two volume pools would be required. One for encrypted T10000 media and one for non-encrypted T10000 media.

The volume pools would be:

T10K_non-encrypted → contains T10000 media that is used in non-encrypting drives

T10K_encrypted → contains T10000 media that is used in encrypting drives

Volume pool assignment is determined at a policy level. Backup policies that use non-encrypting drives should be set to draw from the T10K_non-encrypted volume pool and policies that use encrypting drives should draw from the T10K_encrypted volume pool. Here is what a sample policy utilizing encrypting T10000 tape drives looks like in this mixed encryption configuration. The policy calls on a specific T10000 storage unit and a volume pool that is dedicated for encrypting media operations.



By default, NetBackup will place newly inventoried media into the "NetBackup" pool and policies will draw from this pool. After defining the specific volume pools to be used for media management in a mixed encryption solution, it is necessary to populate these pools with the proper media. This is where specific volser ranges for the tape cartridges can come in handy. The volser is a media label on the tape cartridge that is read by the automated tape library. The volser can be up to six characters in any combination of letters or numbers.

A recommended strategy is to lock the first digit of all media to be used by encrypted drives to an "E". Media designated for use in non-encrypting drives could be ordered with an "N" locked into the first digit. The remaining five digits should be numbers to define a unique media cartridge within that pool (i.e. E00001). A labeling scheme such as this can be particularly useful when moving media between primary and disaster recovery sites.

Scratch Pools

If a scratch pool is defined in a NetBackup configuration, all expired media will be automatically transferred into the scratch pool. Once there, it can be drawn and re-assigned to whichever volume pool needs it. If a scratch pool is not defined, expired media will remain in the volume pool to which it

is assigned. This media will be available for re-use within that volume pool but will not be available to other volume pools.

NetBackup does not support multiple scratch pools. Writes from beginning of tape such as re-label or erase operations will cause the KMS cluster to disassociate all old keys from the tape cartridge. Due to the lengthy time of elevator requests, it is inadvisable to use a scratch pool when the drive configuration spans multiple rails on a SL8500 library.

Operations within a Mixed Encryption Configuration

By implementing the recommendations laid out in this chapter, a mixed encryption NetBackup configuration that contains both encrypting and non-encrypting drives of the same model in the same logical library partition can perform the following functions:

- **Concurrent backup or restore of encrypted and non-encrypted data:** The NetBackup master server can field concurrent requests for backup or restore operations on both encrypted and non-encrypted media.
- **Using multiple data streams for tape backup operations:** NetBackup can employ the use of multiple data steams in tape backup operations to significantly reduce backup times. This can occur by using multiplexing to combine several streams from several sources into one tape drive, splitting a single backup of a large file system among several different tape drives or using a combination of these two strategies. When using multiple streams, data from one backup job is either mixed in with data from other backup jobs or striped across multiple tapes. This brings to light the question on how this behavior functions in an encryption solution. The answer is that when using the recommendations outlined in this chapter, multiple data streams function without issue in encryption environments. Multiplexing is recommended if server processing, network speeds or disk throughput are insufficient to stream the high-speed T10000 tape drives. Benefits of multiplexing backups include significantly reducing the backup window as well as alleviating the stop/start wear that is incurred on a tape drive that has insufficient data to stream writes.
- **Using a scratch pool:** A mixed encryption NetBackup environment can support using a scratch pool if it is needed. Be aware that a scratch pool is not advisable in certain library configurations where media would be required to be transferred by elevators, go across rails or through pass-through ports.

Suggestion For Adding Media:

Consider purchasing media of a specific volser range for encrypted operations. This makes it easier to identify encrypted cartridges. Make sure that the volser ranges chosen are unique and distinct from current ranges. For example, you may decide that all cartridge volsers that will be used with encrypting drives start with the character “E”. Please note that this suggestion will not be helpful in an environment that utilizes a single scratch pool for mixed encryption operations.

TSM Mixed Encryption Implementation

In order to implement a manageable mixed encryption TSM solution, it is necessary to define a media management strategy. By customizing how drives are configured in the solution, a TSM administrator will be able to assign scratch media to the appropriate TSM device and eliminate issues of encrypted media being assigned to an unencrypted drive and vice versa. This strategy can be accomplished by isolating encrypted tape devices at a TSM server or device level.

The segregation of encryption tape device types can be implemented in two layers of the overall solution, first at the SAN level with switch hard zoning and second at the application level by creating separate TSM libraries. This second method segregates encrypting and non-encrypting tape drives of the same model at the application level by creating separate TSM libraries on the same server. Encrypting drives will be defined in one library while non-encrypting drives will be defined in another.

For scratch tape pool management there are two options for managing media within a mixed encryption environment. The first solution uses TSM Native ACSLS Library Management with separate scratch pools for each TSM library resource. The second solution uses Gresham Storage EDT Library Management with one large EDT managed scratch pool.

If a multiple TSM server solution is feasible, implementing fiber channel zoning for specific tape devices to a dedicated TSM server is the preferred method of isolating mixed encryption drives. This method is recommended over segregation at the application level in situations where persistent binding of the tape devices is not reliable or extra protection is needed. With this hard zoning method, encrypting drives are allocated as a device to one group of TSM servers; non-encrypting drives are allocated to another group of servers. An individual TSM library is then created on each server and a scratch pool created for each TSM defined library. The encrypted tape drives are assigned to one TSM server library and the non-encrypted tape drives assigned to the second TSM server library. This configuration creates separate server database records, separate scratch pools, separate device classes, and separate resource utilization. Expired tape media returns back into the isolated scratch pool and can be reused by the TSM server without issue.

For better resource utilization, tape drive I/O paths are often shared by additional hosts defined as storage agents. Storage agents should be configured to use the same library and device class that the TSM server uses. The addition of storage agents will not pose a problem to the solution.

It may be necessary to implement a single TSM server solution with mixed drives that are shared with additional storage agent hosts. In this configuration, separate TSM libraries must be created on the TSM server, creating one library for encrypting drives and one library for non-encrypting drives. Tape media from the ACSLS library must be separated into two scratch pools.

Figure 5 shows how our sample mixed encryption environment shown in figure 3 would look when implemented in a TSM configuration with two separate TSM servers. The TSM server on the right manages all encrypting media operations. It has three storage agents that are associated with it to increase drive utilization. The server on the left handles all non-encrypting media functions and has two storage agents associated with it. Fabric zoning segregation is represented by the ovals on each side. Encrypting drives are only visible by hosts on the right and non-encrypting drives are only available to hosts on the left.

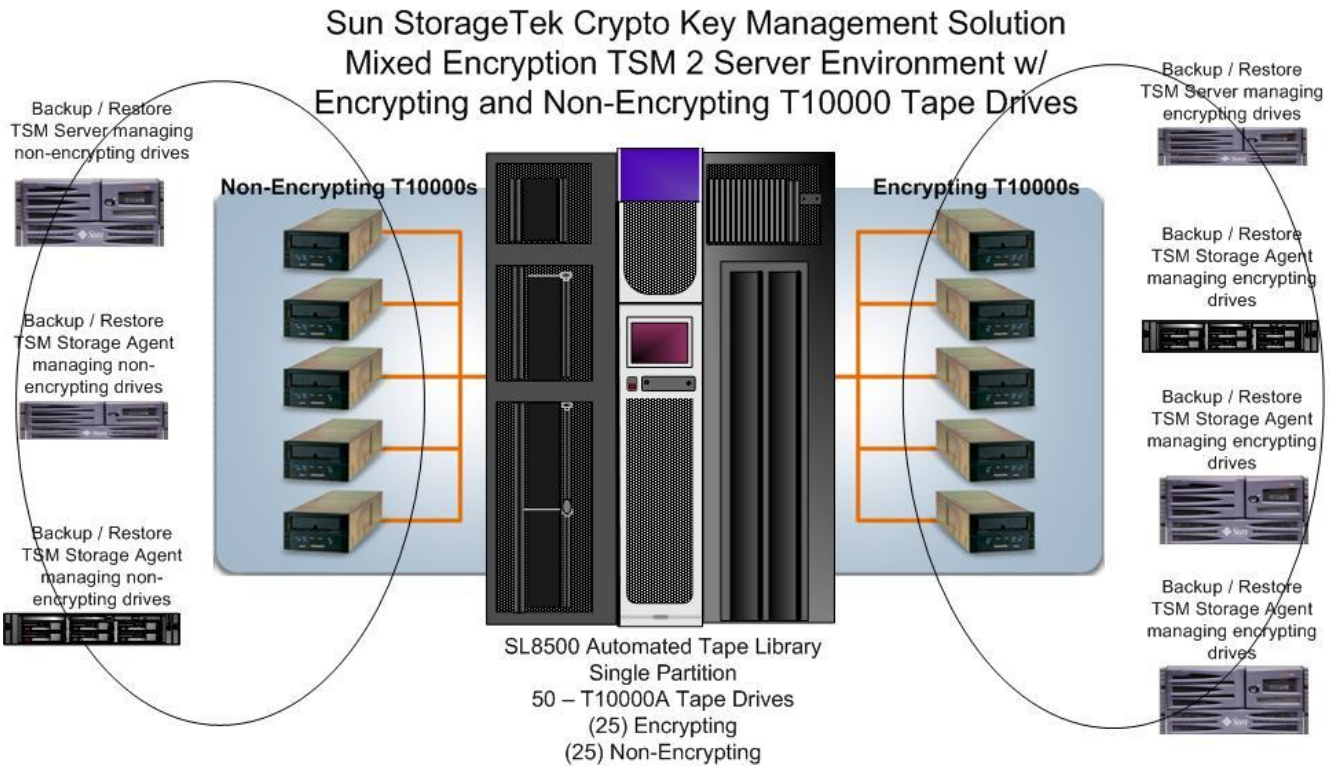


Figure 5: TSM 2 Server Mixed Encryption

The drawback of a two server implementation is that it increases management and administration tasks by essentially creating two separate TSM configurations. Figure 6 shows how this same sample environment would look if it were implemented in a one server TSM mixed encryption configuration. In this example the TSM server is on the top right of the diagram and all other hosts are setup as storage agents for this server. All hosts in this example have the ability to access both encrypting and non-encrypting drives in the library. Segregation is accomplished by defining two libraries in the server configuration and two device classes. Encrypting drives are all added to one library and non-encrypting drives are added to the second library.

A drawback of a single server TSM implementation is that separation of encrypting and non-encrypting operations can be compromised if persistent binding of the tape device targets is not maintained. A change in tape device bindings can result in the drive paths of encrypting drives being inadvertently presented to a non-encrypting library.

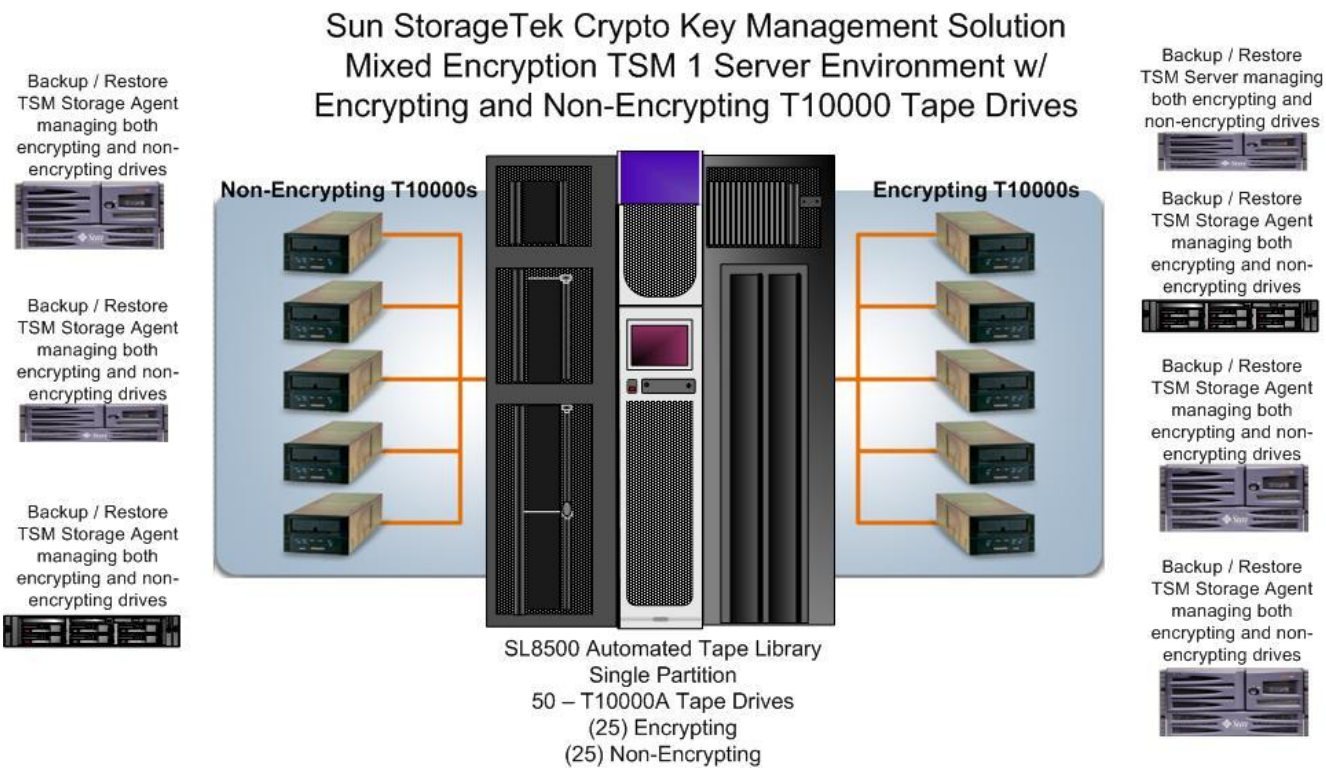


Figure 6: TSM 1 Server Mixed Encryption

Media Management strategy

Once the tape devices are segregated at the server and library level, it is necessary to create scratch pools for each library.

When expired encrypted tape media returns back into the isolated scratch pool, it can be reused by the TSM server without issue as long as the encryption key used to write the tape was not subsequently destroyed. TSM will reassign the expired tape back into the private tape pool and its tape devices will read the encrypted TSM header and write data to the tape. If the encryption key for this tape cartridge was destroyed, the media must be manually exported out of the scratch pool and re-entered back into the tape pool and TSM must be instructed to “Overwrite Existing labels” when checking in the volume. The same steps must be followed if, for media recycling reasons, tapes from an encrypted scratch pool are checked into a non-encrypted scratch pool.

Alternatively, Gresham EDT can be used to create a universal scratch pool that all TSM servers and storage agents can utilize. If Gresham EDT is being used, it is necessary to configure all the EDT libraries to use the same scratch pool ID.

ACSLS and Native TSM Considerations

In a mixed encryption scenario utilizing native TSM, one library is defined for encrypting operations and another for non-encrypting operations. In our mixed encryption sample environment, we have 50 T10000 tape drives. Only 25 of these drives are operating in encrypting mode. Two TSM libraries, with accompanying scratch pools, will be created. One library for the 25 non-encrypting T10000 and one library for the 25 encrypting T10000. This is done to isolate tape media.

TSM Application specific setup information

TSM Library Definitions

Below is an example of the general library definitions used to configure one of the TSM libraries.

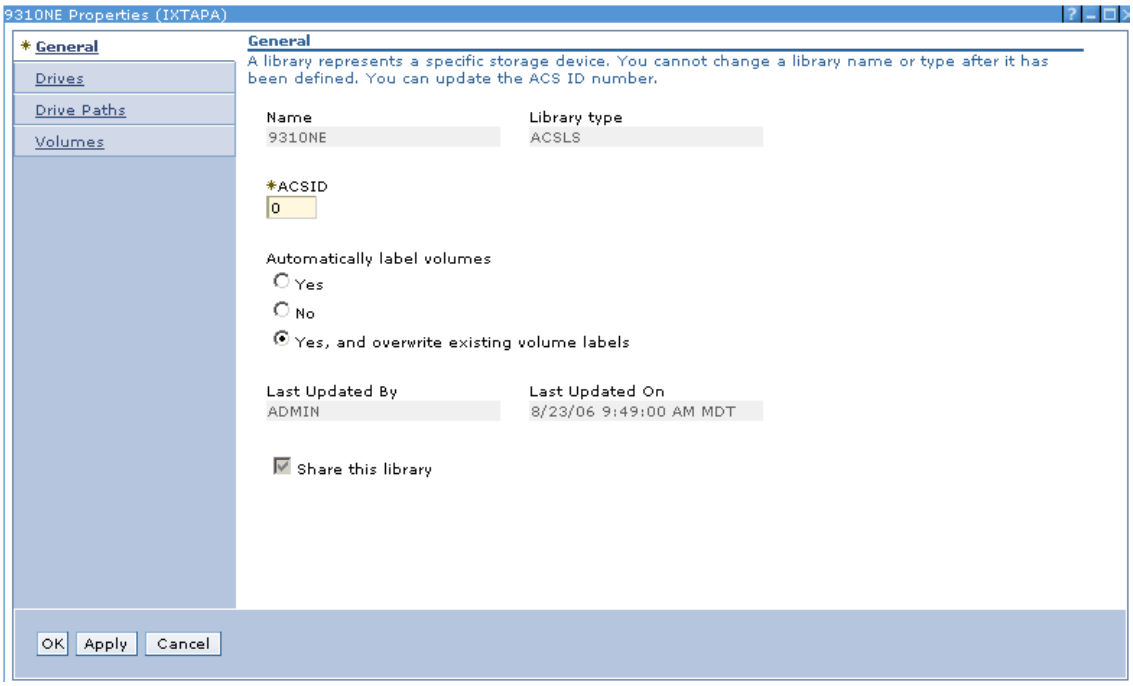


Figure 7: TSM Library Definition

The Library was defined as an ACSLS library with an ACSID of 0.

The *"Yes, and overwrite existing volume labels"* option was selected. This allows TSM to initially label the tape cartridges entered in the TSM defined library's scratch pool. When the cartridges are entered into the scratch pool, each tape is physically loaded in the drive(s) associated with the TSM defined library and labeled with a header.

Continuing with the example from above, if there are currently 1000 tape cartridges available in the ACSLS managed library, 500 tapes could be given to the non-encrypted TSM defined library and 500 could be assigned to the encrypted TSM defined library. Each batch of media would be mounted, labeled and entered into the scratch pool.

If the library is to be shared by additional hosts running the TSM storage agent, select the *"Share the library"* option.

Tape Drive Configuration

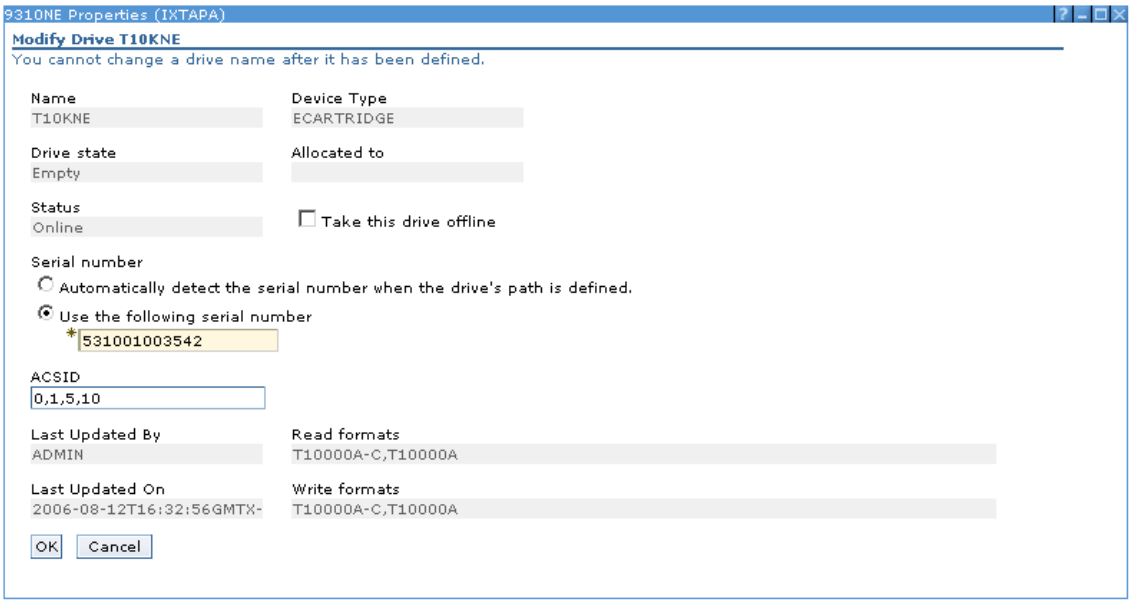
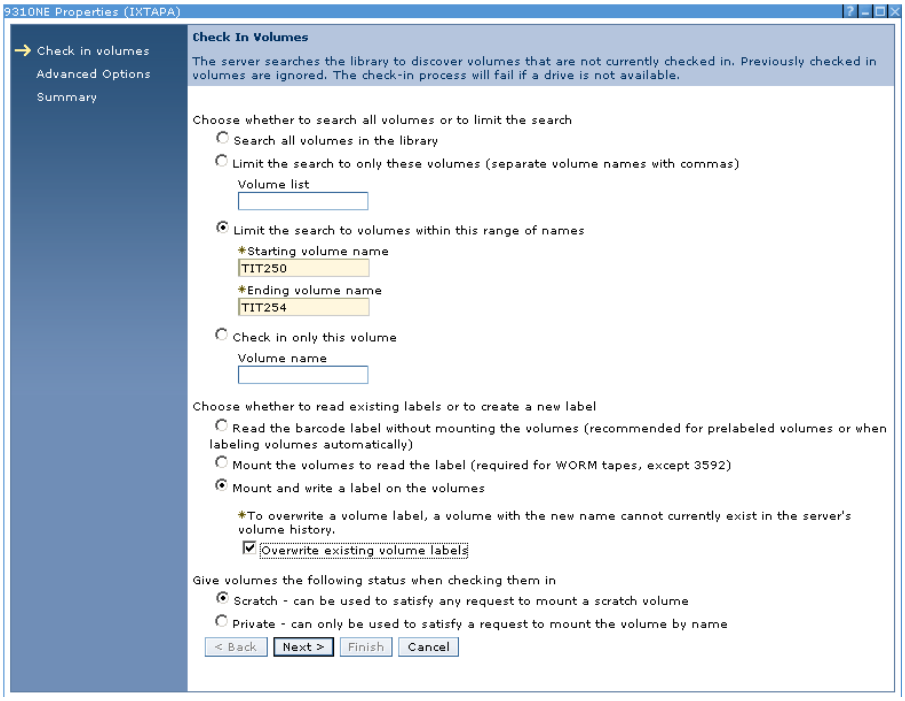


Figure 8: TSM Tape Drive Definition

Tape drives were defined as an "ECARTRIDGE" device type and given the correct ACSLS ID. On shared libraries, the device path is entered for each host running the TSM storage agent.

Add Volume Configuration

The screen capture on the right demonstrates how volumes are added to a TSM library in a mixed encryption configuration.



When adding volumes during the check-in process, *"Mount and write a label"* and *"Overwrite existing volume labels"* options were selected. This forces TSM to overwrite existing labels when a volume is initially entered into the scratch pool and eliminates the chance of newly added tape media with unreadable encryption keys from entering into the scratch pool.

Add Device Class

A device class should be created for each TSM library. It is necessary to create a device class for each library / drive type in order to keep tape media isolated at the device level. The reason for creating a separate device class is to prevent a storage pool from calling tape devices that cannot access the media from the scratch pool due to incompatible encrypting modes. One device class is created for non-encrypting drives and another for encrypting drives. Each device class is associated with a separate TSM library and a separate scratch pool.

The next step is to create separate storage pools. Each storage pool is then assigned to a separate device class. By creating a storage pool for each device class, it ensures that backups will only use one type of tape device and only use media from one specified scratch pool. This ensures backups are written using a specific drives with appropriate media assigned from a scratch pool.

Gresham EDT in a TSM Mixed Encryption Implementation

The addition of Gresham EDT software to a TSM mixed encryption solutions serves to simplify ease of management and increase functionality. Gresham introduces a powerful GUI interface that administers management operations of the TSM environment. Gresham presents two advantages that pertain specifically to a mixed encryption implementation. The first is that it enables the ability to use a single scratch pool for managing both encrypted and non-encrypted media. The second is that it enables a force re-label option when expiring cartridges and releasing them back into the scratch pool. In situations where a key used to write the header on an expired tape cartridge has been destroyed or an encrypted tape cartridge is recycled for non-encrypted use, Gresham will force the re-label of the tape volume even if it does not recognize the volume type. Below is an example of a Gresham library configured to force a re-label operation.

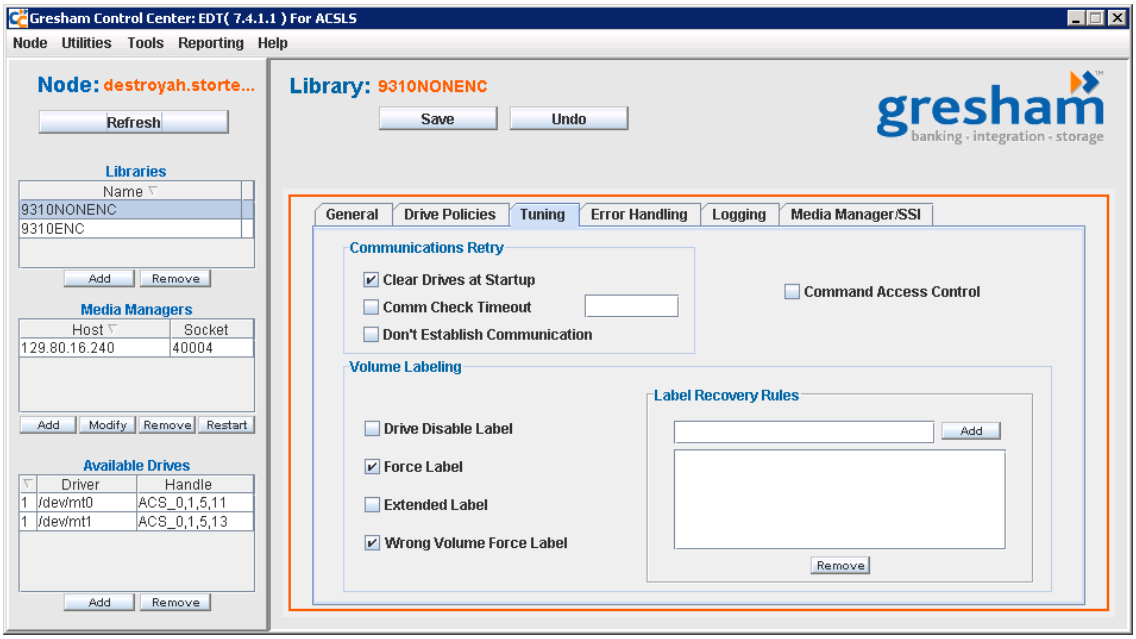


Figure 9: Gresham Library Defined

TSM Solution performance considerations

TSM Application specific client performance tuning:

The following parameters were added to the client system options file (dsm.sys):

TXNBytelimit 2097152

RESOURCEUTILIZATION 8

The *RESOURCEUTILIZATION* option was added to allow multiple threads (streams) to the encrypting tape drive.

The *TXNBytelimit** option specifies the number of kilobytes the client program buffers before it sends a transaction to the server. This option also permits you to control the amount of data sent between the client and server before the server commits the data and changes to the server database, thus changing the speed with which the client performs work. TXNBtyelimit was set to the maximum number, 2097152 (2MB).

TSM Application specific server performance tuning:

The following parameters were added to the server system options file (dsmserv.sys):

TXNGroupmax 256

MOVESizethresh 2048

MOVEBatchsize 1000

The *TXNGroupmax** option specifies the number of objects that are transferred as a group between a client and the server between transaction commit points. The objects transferred are actual files, directories, or both.

While it is possible to affect the performance of client backup, archive, restore, and retrieve operations by using a larger value for this option, in our particular testing scenario, increasing the parameter above 256 appeared to have little to no effect on storing data.

The *MOVESizethresh** option specifies, in megabytes, a threshold for the amount of data moved as a batch, within the same server transaction. When this threshold is reached, no more files are added to the current batch, and a new transaction is started after the current batch is moved. 2048 was set to keep the amount of data moved consistent with the block transfer size of the T10000 tape drives.

The *MOVEBatchsize** option specifies the number of client files that are to be moved and grouped together in a batch, within the same server transaction. This data movement results from storage pool backups and restores, migration, reclamation, and MOVE DATA operations. This option works with the MOVESIZETHRESH option.

If proper system memory resources are available, additional server options such as BUFPOOLSIZE and LOGPOOLSIZE can be used to further enhance the efficiency of TSM.

For example, *BUFPOOLSIZE** specifies the size of the database buffer pool in kilobytes. A large buffer pool means that database pages remain longer in memory cache, and Tivoli Storage Manager requires fewer input/output operations to server storage.

[* Definitions of TSM options are taken, in part from the Tivoli Storage Manager for Solaris Administration guide]

Applying Mixed Encryption Principles to Alternative Environments

While the focus of this mixed encryption chapter and accompanying testing effort is on Veritas NetBackup and Tivoli Storage Manager, many of the principles and recommendations covered in this chapter can be applied to other applications as well. Sun StorageTek Enterprise Backup Software (EBS) was implemented in encryption only environments as part of our product validation coverage. However, we will provide some recommendations in this section for implementing mixed encryption in an EBS configuration.

Primary Components of a Mixed Encryption Solution

Storage Configuration Architecture

The configuration architecture dictates how many media servers exist in the configuration and how they are used. To implement a mixed encryption architecture where drives are segregated by being available to one group of media servers and not another, it is required that fabric zoning be used. Segregating drives by function (i.e. non-encrypting vs. encrypting) is the recommended solution due to the added layer of protection it provides and the reduced level of media administration that would be required compared to other methods.

Some backup applications can differentiate among tape devices by more than just drive type. In these cases, it is possible to implement a mixed encryption architecture in which requests for both encrypted and non-encrypted operations can be handled by the same media server. When implementing mixed encryption on EBS, it is recommended to follow the storage configuration architecture that is defined for NetBackup installation in chapter 3.

EBS can function similar to NetBackup by having a central server that manages a group of storage nodes dedicated to encrypting operations and another group that is dedicated to non-encrypting operations. The flexibility to manage encrypting and non-encrypting media operations from a single configuration is combined with the security and protection of segregating tape drives across separate fabric zones.

Media Management Strategy

Media management is a key piece to a successful mixed encryption solution. Encrypting and non-encrypting drives cannot share media from the same pool. A media management strategy is required to support mixed encryption in any environment. This is typically accomplished by use of volume (or media) pooling at the backup application level. All backup applications provide the ability to pool media into user defined groups and setup rules for certain operations to draw from certain groups.

A recommended media management strategy involves using the backup application to define a volume pool for encrypting and non-encrypting functions. Use of a common scratch pool is possible but only recommended in configurations such as NetBackup or TSM with Gresham EDT where tape headers are relabeled after expiration. A scratch pool can impede library performance in certain configurations where tape drives span elevators or pass-through ports. Some backup applications

such as Legato and TSM support multiple scratch pools. In this case, it is possible to configure a scratch pool for each encryption function (i.e. one for encrypting and non-encrypting).

When implementing mixed encryption in an EBS configuration, it is recommended to define volume pools for each media operation (i.e. encrypting and non-encrypting). Access to these volume pools will be defined at the backup policy level. Data that is to be encrypted will draw from the volume pool dedicated to encrypting operations. It is possible to create multiple scratch pools with EBS. If it is necessary to create multiple volume pools of the same function, it could be beneficial to utilize a scratch pool for encrypting operations and another for non-encrypting operations. An example of this is a configuration that defines separate volume pools for individual clients or client group. The result of this would be multiple volume pools that service encrypting functions and several other volume pools that service non-encrypting operations. To simplify media management operations, a scratch pool could be created to supply all encrypting volume pools and a second scratch pool to supply all non-encrypting volume pools. Media would be recycled to these scratch pools and new media would only be added to the scratch pools. Tapes would transfer to the volume pools as needed.

Chapter 4: Legacy to New Generation Drive Migration

Compatibility Rules

Integrating T10000B or T9840D drives into an environment that contains earlier version drives of the same type (T10000 or T9840A/T9840B/T9840C) requires careful planning. For both drive types, all drives of the same type use the same media. However, the new version of each drive type writes data to tape using a higher density. Consequently, the following rules apply:

- A legacy version of a given drive type cannot read data from or append data to media written by the new version of the same drive type.
- A new version drive can read media written by a legacy drive of the same type but cannot append data to that media.
- A drive can write from beginning of tape (BOT) to media written by another drive of the same type, regardless of generation.

Migration Strategies

The simplest strategy to implement is to replace all legacy drives of a given type with new generation drives of the same type. New generation drives cannot append data to media containing data written by legacy drives of the same type. Therefore, in an automated library environment, media written by legacy drives must be logically separated from media that will be written by the new generation drives. This can be accomplished by creating a new media pool for media to be used exclusively by the new generation drives. In this implementation, backup policies based on storage units using this drive type are reconfigured to use the new media pool to avoid append operations to incompatible media. The original pool of media is accessed only to restore data written by legacy drives.

The use of a scratch pool in this situation is recommended. Since writes from BOT are allowed regardless of how the media was previously used, media that contains only expired data written by legacy drives can be safely assigned to the new media pool. If a scratch pool exists, when all of the data on a tape volume expires, regardless of the pool to which it is assigned, the volume is moved to the scratch pool and reassigned to the new media pool as needed. This ensures that media written by legacy drives can be reused by the new generation drives as soon as the data on it expires. Since media in the original pool is only read and never written, no expired media will ever be reassigned to it. Eventually, all of the media in the original pool may migrate to the new pool to be reused by the new generation drives.

Some customers may wish to implement a phased migration, replacing legacy drives a few at a time. This strategy is much more difficult to implement and requires careful planning and execution. If legacy and new drives of the same type co-exist in the same environment, the challenges are similar to those of a mixed encryption environment.

In a mixed drive environment, the potential for loading incompatible media into a drive is increased because the backup application does not distinguish between legacy and new generation drives of the

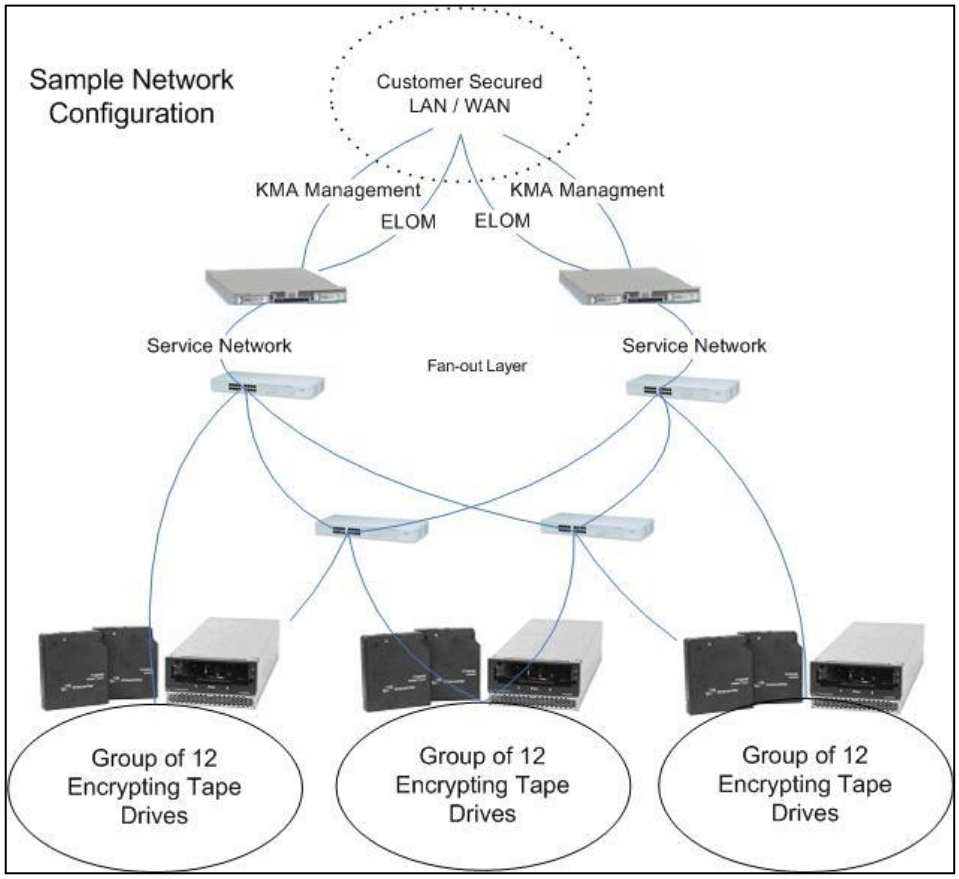
same type. When a backup or restore request is processed, the application may choose any one of the drives of a given type to handle the request without regard to whether the media to be used is compatible with the specific drive chosen. Thus, in this implementation legacy and new generation drives must be logically separated as well as the media written by them. Techniques similar to those recommended in Chapter 3 to manage mixed encryption environments may be used to ensure that media/drive compatibility errors do not occur.

Chapter 5: Optimizing Encryption Solution for Redundancy

Redundant Network

Unlike version 1.0, version 2.0 of the Sun StorageTek Crypto Key Management Solution does not retain encryption keys locally on the drive after unloading a tape. All requests for keys are serviced by a KMA in the KMS cluster and assigned to the drive. Protecting against component failure is a primary concern in maintaining uninterrupted operation of the encryption solution.

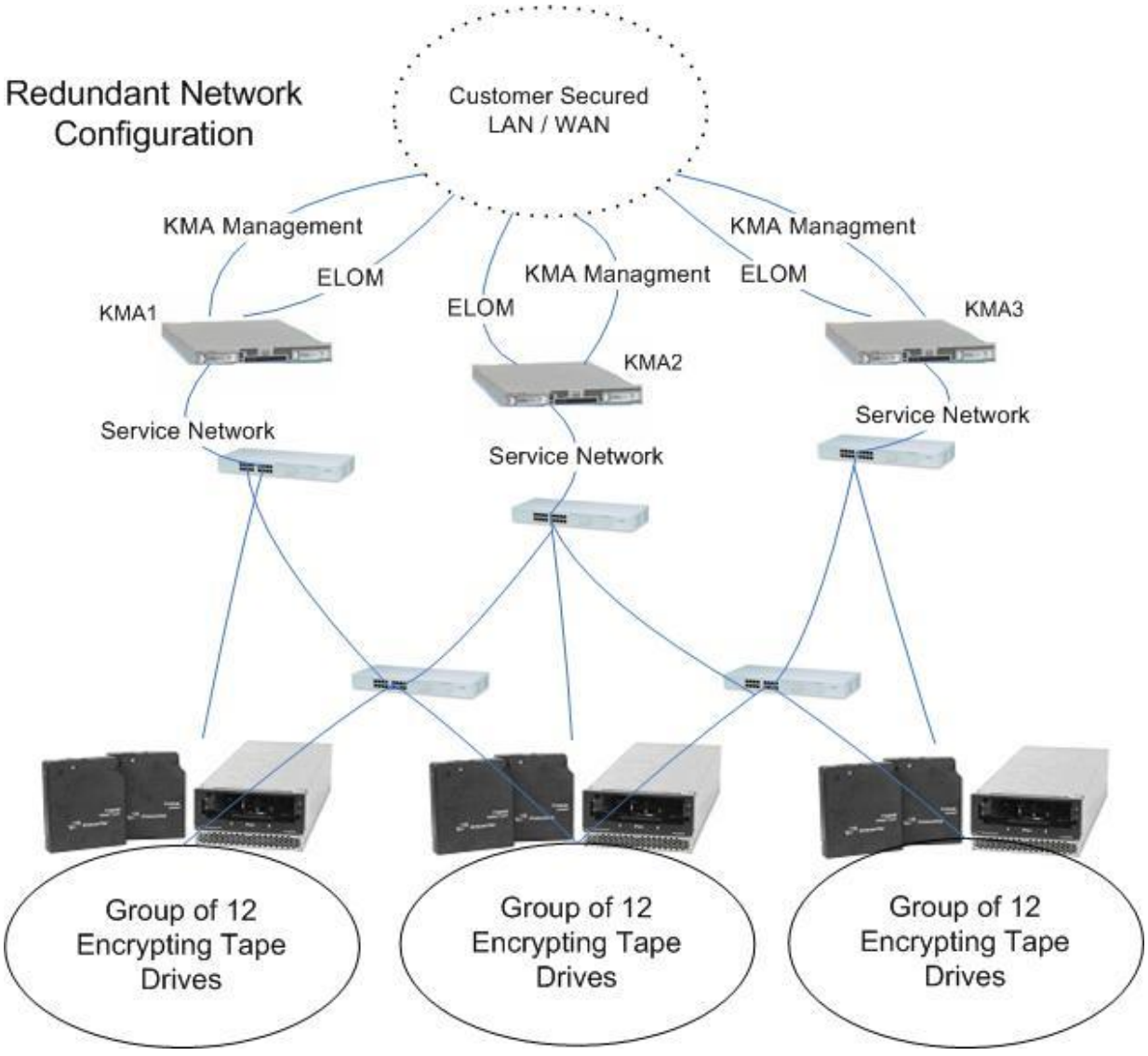
Each Key Management Appliance is configured with three network ports. One port is dedicated to the Enhanced Lights Out Manager (ELOM) that is used to provide remote access to the console. The ELOM is used during initial setup and configuration of the KMA. The KMA is a locked down appliance so very few functions are available through the ELOM but the security officer can utilize the ELOM to perform certain operations such as resetting the KMA. The second configured Ethernet port is dedicated to KMA management. This connects to the customer's secured network and is utilized for the KMS software manager to administer the cluster as well as for replicating changes across the cluster to other KMAs. The final configured Ethernet port is dedicated to the service network on which the encrypting tape drives will reside. It is recommended that the encrypting tape drives be installed on the services network to reduce network traffic and limit external access.



To protect against component failure, it is recommended to install redundant switches on the services network. The example illustrated on the right shows a simple recommended network for an environment with 36 encrypting tape drives. The first step in protecting the network is to connect the KMAs to separate switches on the services network. Any KMA can provide full functionality for any drive enrolled in the cluster. If a component failure is experienced on a KMA or on the switch to which a KMA is connected, the second KMA in the cluster will service available drives and encrypted operations will continue unaffected. Ethernet switches that are directly connected to KMAs are referred to as fan-out switches. The next step in protecting the network is to distribute encrypting tape drives evenly across all available switches, including available ports on fan-out switches. This serves to reduce the impact of a component failure at the switch level by rendering only the drives directly connected to the failed

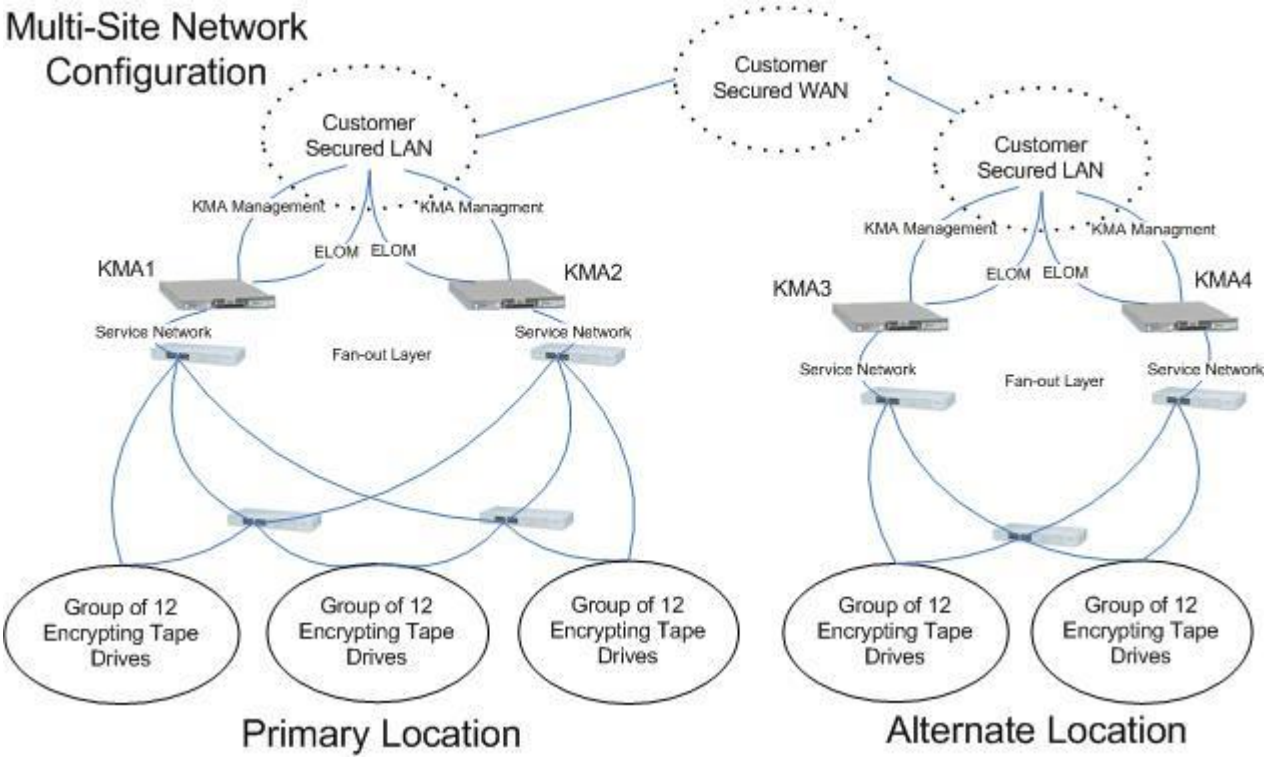
switch temporarily inoperable. The final step is to utilize redundant cabling methods. A good rule of thumb is that any switch that is not directly connected to a KMA should have connections to two switches that are. This ensures that a single failure of a KMA or a switch will not impact the ability of the encryption solution to continue processing data. The configuration pictured above requires the use of spanning tree capable switches. Additionally, uninterruptible power supplies can be used to protect switches and KMA components from loss of power interruptions.

For customers who place a top priority on redundancy, there are some additional configuration changes that will help further mitigate the potential for interrupted operations due to component failure. These include the addition of a third KMA to the local KMS cluster and extra switches on the services network to reduce the number of drives that would be impacted by a switch component failure. An additional KMA in the local cluster adds security in the event that a component failure forces one KMA to be unavailable for an extended period of time. The remaining KMAs can continue to function as a cluster while managing the encryption solutions. Load balancing, replication and protection of new keys are achieved. The example below represents a sample configuration enhanced for maximum redundancy.



Multi-Site Network

A single KMS cluster can administer key management functions for encrypting tape drives spread across many site locations. The diagram below provides an example of a configuration that includes encrypting tape drives at two separate locations. It is important to note that while all four KMAs in this example are clustered together and replicate changes across the cluster, it is still highly recommended to have a local cluster of at least two KMAs at each site. This provides path redundancy and load balancing abilities for the drives at that site.



Chapter 6: T10000 Tape Drive Characterization

Characterization Overview

The purpose of this chapter is to provide some baseline characterization data on how the T10000 encryption capable tape drive performs in an emulated customer environment. This is not intended to be a systems level benchmark and as a result of this, all characterization tests are performed at a file system level. Customer emulated data of varying file sizes are used to most accurately characterize performance within typical real world configurations. The test results in this section were derived from lab testing using Veritas NetBackup on Solaris operating systems but the performance concepts as well as the numbers obtained within can also be applied to other configurations as well.

An emphasis is placed on comparing performance in encrypting and non-encrypting modes. A drawback of many encryption models is a significant performance impact. The Sun StorageTek Encryption Solution has a competitive advantage in this regard.

Data Buffer Transfers

The T10000 tape drive can support block size transfers of up to 2 megabytes. This means that when configuring applications to utilize the drive, data buffers can be set as high as 2MB. It is advantageous to utilize large data transfers to tape as this will enable the high speed T10000 to more easily achieve streaming speeds.

One thing to note when setting data transfer sizes is that other tape drives, including the STK 9x40 series, only support block size transfers up to 256KB. If T10000s are being installed in a mixed drive environment that includes some of these drives, then it is recommended to set the size of the data buffer transfers to 256KB. If the T10000 drives are not in a mixed drive environment or are mixed with other tape drives that also support large block size transfers, then it is recommended to set the size of the data buffer transfers to at least 512KB. Archiving scenarios can benefit from a data buffer size as large as 2MB.

The number of data buffers sent to tape is also configurable in NetBackup and many other backup applications. This can be described as the number of concurrent threads that are sent to the T10000 tape drive. Increasing the number of data buffers can often improve tape performance but will require more system memory.

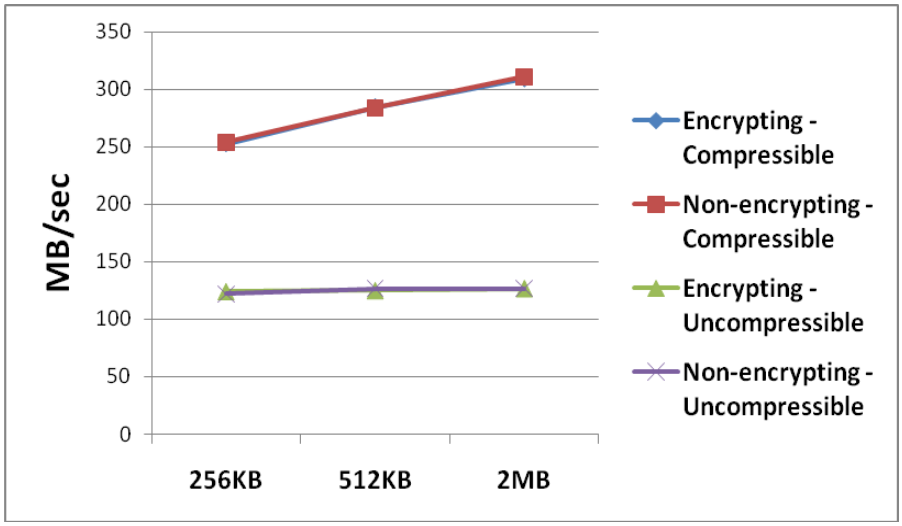
Setting the size and number of data buffers in NetBackup requires creating touch files in the <install path>/netbackup/db/config directory on the local media server. SIZE_DATA_BUFFERS contains the data buffer size measured in bytes and NUMBER_DATA_BUFFERS determines how many threads are used. These are dynamic settings and require no restarting of process daemons to take effect. The screenshot example below shows how these can be modified.

```
t2000u# pwd
/usr/openv/netbackup/db/config
t2000u# cat NUMBER_DATA_BUFFERS
16
t2000u# cat SIZE_DATA_BUFFERS
524288
t2000u#
```

Figure 10: NetBackup Settings

Lab Performance Data

The graph below shows characterization data captured using T10000 tape drives in both encrypting and non-encrypting modes. Data was obtained using single stream backups of verifiable data files. Data sets included both compressible and uncompressible data. As evidenced in the graph, enabling a T10000 tape drive for encryption in the Sun Crypto Key Management Solution has a negligible impact on performance. In fact, the impact on performance is significantly less than the average variance between test runs. As a result of this, some media operations in encrypting mode record a faster time than equivalent operations in non-encrypting mode.



Sun's encryption solution adds a fixed amount of 100 bytes overhead per data buffer transferred. Backup and archiving applications should be configured for a minimum of 256KB transfer size. In this model, encryption would impact performance by only 0.04%. In other configurations, the impact would be even less. The encryption algorithm occurs at the device level after the T10000 has already compressed the data for storage, allowing the maximum benefit of increased speed and capacity due to compression to be realized.

Native data transfer rate for the T10000 is 120MB/sec. This is observed in the results of our tests using uncompressible data sets. When compression is utilized, performance and capacity can be greatly increased.

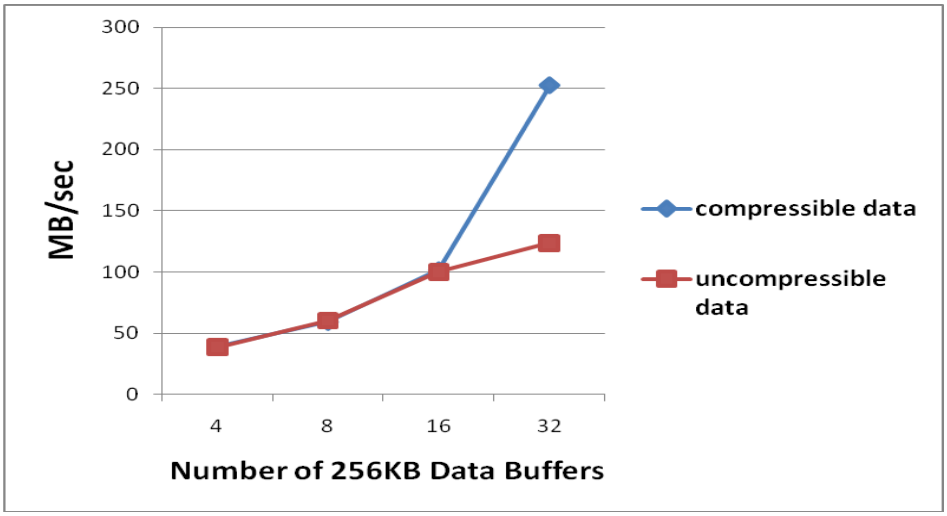
Small Data Buffers

The following graph depicts encrypting T10000 performance when using 256KB data buffer transfers. This example is typical of an environment where T10000 drives are mixed with 9x40 or other drives that do not support large block size transfers and it is necessary to set size data buffers to 256KB. The graph characterizes T10000 encrypted performance across various numbers of data buffers.

Numbers on the left show performance characterization while running 4 data buffers (threads) and the graph scales up to 32 data buffers (threads) on the right.

All data is drawn from backup policies executed within NetBackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and uncompressible data sets are show in the graph. All T10000 drives are operating in encrypted mode.

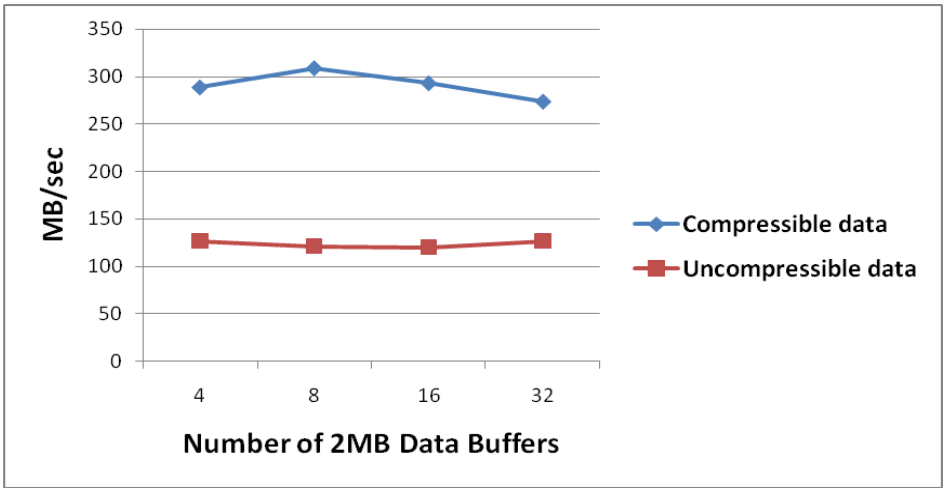
As the graph indicates, when writing data in 256KB blocks, setting the number of data buffers to 32 is advisable. NetBackup does not effectively utilize more than 32 data buffers. If the number of data buffers is configured to something greater than 32, more system memory resources will be allocated and only marginal (if any) performance gain will be realized.



Large Data Buffers

This second graph depicts encrypted T10000 performance when using 2MB data buffer transfers. This example is typical of a customer environment where T10000 drives are not mixed with other tape drives, or they are mixed with other drives that can also support 2MB data buffer transfers. The graph characterizes T10000 encrypted performance across various numbers of data buffers (threads).

All data is drawn from backup speeds executed within NetBackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and uncompressible data is displayed in the graph. All T10000 drives are operating in encrypted mode.



As this graph shows, when writing data in 2MB blocks, 8 data buffers (threads) is sufficient to obtain optimal throughput regardless of the compressibility of the data. Since more data buffers require more memory resources, we recommend using at most 8 threads when using buffers of size 2MB.

Use of 2MB buffers is recommended for archiving solutions or other operations where a customer is writing large data files to tape.

Characterization Data Charts

All data contained in the following charts was obtained using NetBackup speeds with customer emulated file system data. The hosts and file systems used in these tests were tuned for large sequential IO transfers. Small file transfer speeds could be improved with a file system that was tuned specifically for this. All results show speeds of a single stream data set from just one client. This is done to effectively characterize the T10000 drive. In production situations, any backup that fails to achieve tape streaming speeds would be multiplexed together with other backups to achieve native tape speed. More detailed information is provided on multiplexing later in this chapter.

This first table shows average NetBackup speeds when 32 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000 drives.

Using 32 Data Buffers (MB/sec)	**Data Buffer Transfer Size**					
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**
32KB - uncompressible data	4.562	4.588	4.490	4.513	4.604	4.597
32KB - compressible ~2:1	4.502	4.523	4.524	4.575	4.509	4.487
256KB - uncompressible data	15.417	15.872	17.518	17.956	19.994	19.977
256KB - compressible ~2:1	18.221	16.124	18.109	18.222	20.234	20.267
1MB - uncompressible data	53.331	55.944	49.234	45.475	55.482	63.478
1MB - compressible ~2:1	51.583	83.733	52.398	95.885	56.985	101.829
1GB - uncompressible data	122.035	124.043	126.702	124.948	126.716	126.664
1GB - compressible ~2:1	253.758	252.509	308.079	277.879	304.112	274.045
5GB - compressible ~10:1	285.436	255.751	282.902	285.654	267.850	260.717

This table shows average NetBackup speeds when 16 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000 drives.

Using 16 Data Buffers (MB/sec)	**Data Buffer Transfer Size**					
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**
32KB - uncompressible data	4.581	4.568	4.527	4.561	4.593	4.577
32KB - compressible ~2:1	4.631	4.538	5.009	4.528	4.546	4.538
256KB - uncompressible data	15.873	15.868	16.273	17.965	19.035	19.956
256KB - compressible ~2:1	16.120	15.984	18.182	18.173	20.203	20.170
1MB - uncompressible data	62.302	59.460	82.147	84.651	56.374	59.328

1MB - compressible ~2:1	59.666	62.939	54.871	52.296	52.485	60.484
1GB - uncompressible data	100.856	100.095	125.653	124.802	126.692	120.057
1GB - compressible ~2:1	101.638	101.591	283.798	284.244	310.478	293.243
5GB - compressible ~10:1	102.368	101.555	279.271	284.878	306.252	287.913

This table shows average NetBackup speeds when 8 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000 drives.

Using 8 Data Buffers (MB/sec)	**Data Buffer Transfer Size**					
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**
32KB - uncompressible data	4.577	4.550	4.535	4.569	4.579	4.547
32KB - compressible ~2:1	4.516	4.540	4.579	4.555	4.465	4.576
256KB - uncompressible data	14.460	15.744	17.669	17.828	19.984	19.921
256KB - compressible ~2:1	15.105	14.760	18.119	18.075	20.190	20.209
1MB - uncompressible data	44.527	36.176	64.248	73.783	58.518	55.879
1MB - compressible ~2:1	40.316	38.392	87.400	74.290	62.487	85.795
1GB - uncompressible data	68.276	60.154	125.593	116.034	126.690	120.817
1GB - compressible ~2:1	68.178	59.333	136.002	136.370	310.763	308.969
5GB - compressible ~10:1	67.994	60.102	135.540	136.143	299.132	310.248

This table shows average NetBackup speeds when 4 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000 drives.

Using 4 Data Buffers (MB/sec)	**Data Buffer Transfer Size**					
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**
32KB - uncompressible data	4.536	4.581	4.527	4.567	4.583	4.605
32KB - compressible ~2:1	4.530	4.529	4.561	4.552	4.572	4.556
256KB - uncompressible data	15.887	15.682	17.993	17.688	16.324	19.670
256KB - compressible ~2:1	16.076	15.973	18.209	18.053	20.193	19.972
1MB - uncompressible data	24.561	25.622	39.099	46.253	98.458	100.129
1MB - compressible ~2:1	21.815	25.624	43.512	51.187	103.380	102.163
1GB - uncompressible data	40.526	38.264	67.796	67.986	126.607	126.673
1GB - compressible ~2:1	37.200	38.668	68.231	68.194	287.038	288.904
5GB - compressible ~10:1	40.110	37.604	68.241	68.224	291.953	289.518

The Benefit of Multiplexing

Media multiplexing is the process of sending concurrent backups from one or several clients to a single storage device. Multiplexing can serve to reduce backup windows and stream tape drives that could not normally achieve tape speed with a single backup.

The T10000 is a high speed tape drive and in many circumstances client processing power, network throughput or source disk performance will not be great enough to achieve streaming speeds. When this happens, tape drives are forced to start and stop as data cannot be supplied as fast as the drive can write. The T10000 buffer size is chosen so that any start-stop activity does not impact system performance and, unlike mid-range drives, the T10000 mechanism is designed to operate in a start-stop environment with no degradation in reliability. Under these circumstances it is beneficial to use media multiplexing to send several concurrent backup streams to a single T10000. This will serve to reduce backup window times by increasing tape performance and limiting time that backup jobs are waiting for resources. It will also reduce wear on the tape drive and media. When utilizing multiplexing, best performance will be seen when multiplexing data streams from separate clients that use separate source disk.

Multiplexing increases the load on the media server and requires more memory allocation. Because of this, it is recommended to reduce the number of data streams (threads) setting that is used to write to tape. If moderate multiplexing (2-6 concurrent backups) is regularly used, then it is recommended to reduce the number of data streams setting by one half (i.e. 16 to 8). If heavy multiplexing (8+ backups) is regularly used, then it is recommended to reduce the number of data streams setting by three quarters (i.e. 16 to 4). There is no special action that is needed to restore a multiplexed backup but it does negatively impact restore performance as the backup application is required to sort through data from multiple backups.

Since Sun Tape Encryption is managed at a device level, there are no issues with multiplexing several backups in an encrypted environment. For steps on configuring multiplexing, please reference your backup application's system administrator's guide.

Using Multiple Write Drives

Using multiple write drives entails splitting a backup into multiple data streams and concurrently sending these data streams to several tape drives. This is beneficial in situations where a single large database that resides on a powerful server and very fast disk array is required to be backed up.

Due to the high speed of the T10000 drives, situations that call for multiple write drives are rare. To effectively utilize multiple T10000 drives for a single large backup, the source disk, media server and SAN need to be capable of very high throughput. Multiple write drives can shrink the time it takes to do a single lengthy backup and serve to reduce backup windows.

For steps on configuring multiple write drives, please reference your backup application's system administrator's guide.

Chapter 7: T9840D Tape Drive Characterization

Characterization Overview

The purpose of this chapter is to provide characterization data specific to the T9840D encryption capable tape drive. As with the T10000 drive, this is not intended to be a systems level benchmark and as a result of this, all characterization tests are performed at a file system level. Customer emulated data of varying file sizes are used to most accurately characterize performance within typical real world configurations. The test results in this section were derived from lab testing using Veritas NetBackup on Solaris operating systems but the performance concepts as well as the numbers obtained within can also be applied to other configurations as well.

An emphasis is placed on comparing performance in encrypting and non-encrypting modes. A drawback of many encryption models is a significant performance impact. The Sun StorageTek Encryption Solution has a competitive advantage in this regard.

Data Buffer Transfers

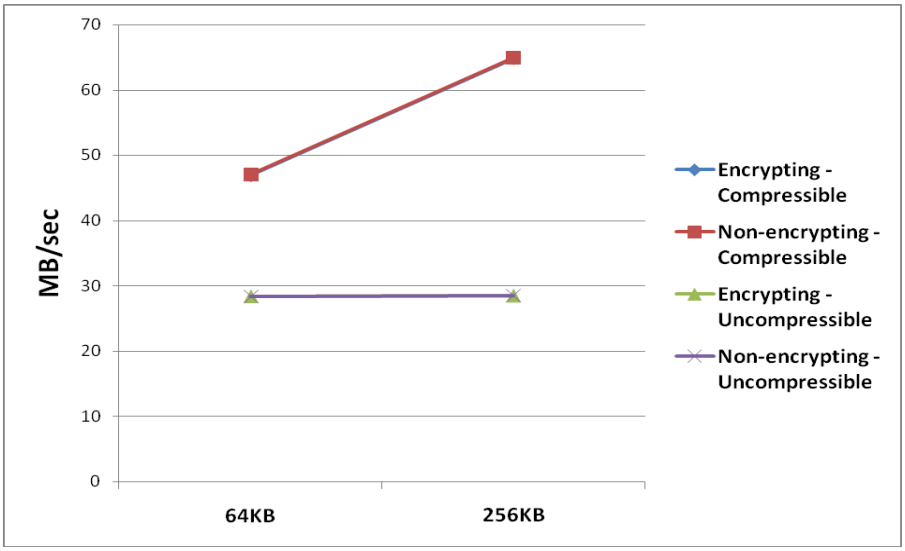
Designed for fast data access, the T9840D drive accesses data in 16.5 seconds, has a native transfer rate of 30 MB/sec and stores up to 75 GB of uncompressible data per cartridge. The T9840D tape drive can support block size transfers of up to 256KB. This means that when configuring applications to utilize the drive, the data buffer size can be set as high as 256KB. NetBackup uses a default data buffer size of 64KB. Increasing this size to 256KB allows the T9840D drive to stream data to tape more efficiently.

The number of data buffers sent to tape is also configurable in NetBackup and many other backup applications. This can be described as the number of concurrent threads that are sent to the tape drive. Increasing the number of data buffers can often improve tape performance but will require more system memory.

Lab Performance Data

The graph on the next page shows characterization data captured using T9840D tape drives in both encrypting and non-encrypting modes. Data was obtained using single stream backups of verifiable data files. Data sets included both compressible and uncompressible data. This graph clearly demonstrates that enabling a T9840D tape drive for encryption in the Sun Crypto Key Management Solution has no observable impact on performance; the graphs representing data of the same type written with and without encryption are indistinguishable.

Sun's encryption solution adds a fixed amount of 100 bytes overhead per data buffer transferred. If backup and archiving applications are configured for a 256KB transfer size, encryption impacts performance by only 0.04%. The encryption algorithm occurs at the device level after the T9840D has already compressed the data for storage, yielding the maximum benefit of increased speed and capacity due to compression.

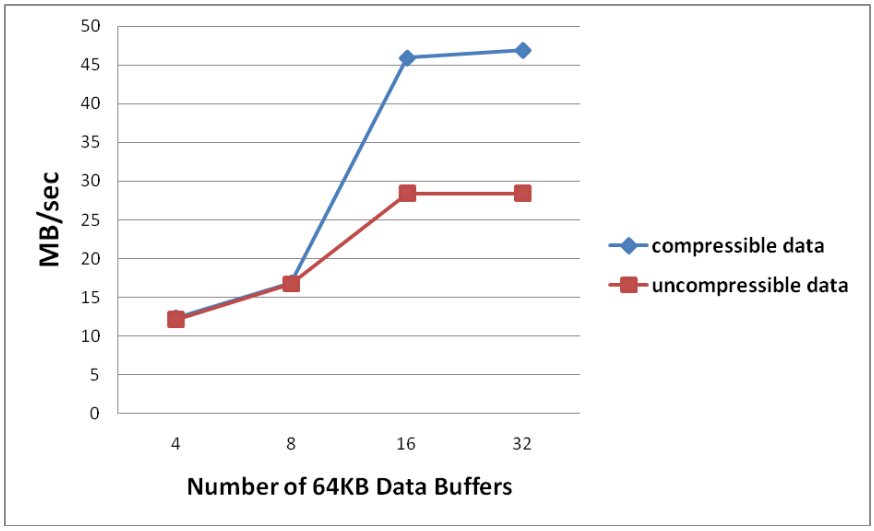


Native data transfer rate for the T9840D is 30MB/sec. Rates very close to this were observed in lab testing using Veritas NetBackup to back up uncompressible data sets with file sizes of 1MB or higher. When compression is utilized, performance and capacity can be greatly increased.

Default Size Data Buffers

Veritas NetBackup uses a default data buffer size of 64KB. The following graph depicts encrypting T9840D performance when using 64KB data buffer transfers. The graph characterizes T9840D encrypted performance across various numbers of data buffers. Numbers on the left show performance characterization while running 4 data buffers (threads) and the graph scales up to 32 data buffers (threads) on the right.

All data is drawn from backup policies executed within NetBackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and uncompressible data sets are shown in the graph. All T9840D drives are operating in encrypted mode.



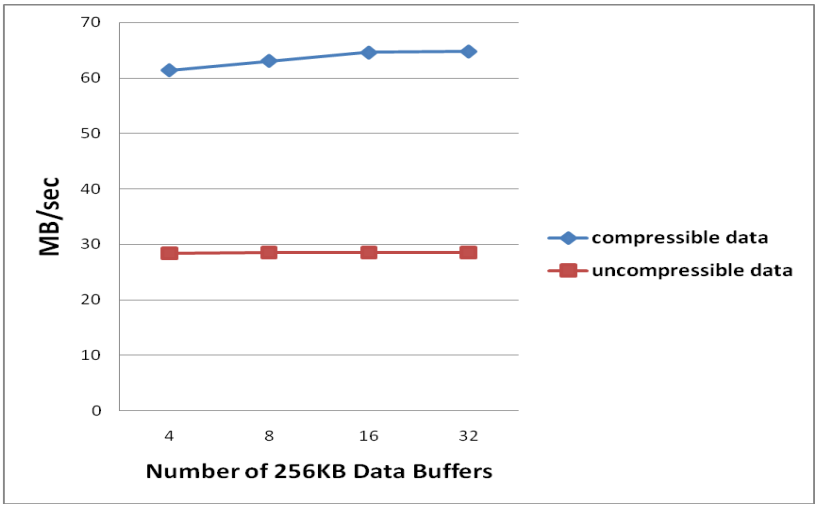
As the graph indicates, when running data buffer transfers in 64KB blocks, it is advisable to set the number of data buffers to 16. However, the poor performance that results from using this buffer size indicates that this default buffer size is insufficient and should always be increased when using Veritas NetBackup as the backup application.

Maximum Size Data Buffers

This second graph depicts encrypted T9840D performance when using 256KB data buffer transfers. The graph characterizes T9840D encrypted performance across various numbers of data buffers (threads).

All data is drawn from backup policies executed within NetBackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and uncompressible data is displayed in the graph. All T9840D drives are operating in encrypted mode.

Comparing this graph to the previous one demonstrates clearly that the preferred buffer size for the T9840D is 256KB. The performance improvement generated by the larger buffer size is more dramatic with fewer streams (threads) but is especially impressive for compressible data regardless of thread count. In fact, optimal throughput for both compressible and uncompressible data is obtained with very low thread count when transferring data in blocks of size 256KB.



Characterization Data Charts

All data contained in the following charts was obtained using NetBackup policies with customer emulated file system data. The hosts and file systems used in these tests were tuned for large sequential IO transfers. Small file transfer speeds could be improved with a file system that was tuned specifically for this. All results show speeds of a single stream data set from just one client. This is done to effectively characterize the T9840D drive. In production situations, any backup that fails to achieve tape streaming speeds would be multiplexed together with other backups to achieve native tape speed.

The first table shows average NetBackup speeds when 32 data buffers of 64KB and 256KB sizes are executed to both encryption enabled and non-encrypting T9840D drives.

Using 32 Data Buffers (MB/sec)	Data Buffer Transfer Size			
File Size of Data Set	64KB (No Enc)	64KB Encrypted	256KB (No Enc)	256KB Encrypted
32KB - non compressible data	5.762	5.762	5.772	5.651
32KB - compressible ~2:1	5.805	5.795	5.793	5.811
256KB - uncompressible data	14.961	14.946	18.129	18.095

256KB - compressible ~2:1	**15.210**	**15.237**	**18.484**	**18.508**
1MB - uncompressible data	**28.417**	**28.403**	**28.442**	**28.492**
1MB - compressible ~2:1	**40.032**	**40.007**	**46.135**	**46.109**
1GB - uncompressible data	**28.433**	**28.399**	**28.420**	**28.394**
1GB - compressible ~2:1	**46.950**	**46.881**	**64.906**	**64.843**
10GB - compressible ~10:1	**47.291**	**47.204**	**64.611**	**64.464**

This table shows average NetBackup speeds when 16 data buffers of 64KB and 256KB sizes are executed to both encryption enabled and non-encrypting T9840D drives.

Using 16 Data Buffers (MB/sec)	**Data Buffer Transfer Size**			
File Size of Data Set	**64KB (No Enc)**	**64KB Encrypted**	**256KB (No Enc)**	**256KB Encrypted**
32KB - uncompressible data	**5.765**	**5.757**	**5.764**	**5.764**
32KB - compressible ~2:1	**5.801**	**5.812**	**5.795**	**5.813**
256KB - uncompressible data	**14.992**	**14.941**	**18.168**	**18.129**
256KB - compressible ~2:1	**15.203**	**15.190**	**18.525**	**18.493**
1MB - uncompressible data	**28.405**	**28.427**	**28.419**	**28.395**
1MB - compressible ~2:1	**35.274**	**35.337**	**46.199**	**46.174**
1GB - uncompressible data	**28.381**	**28.416**	**28.423**	**28.456**
1GB - compressible ~2:1	**45.954**	**45.880**	**64.578**	**64.584**
10GB - compressible ~10:1	**47.145**	**47.014**	**64.587**	**64.498**

This table shows average NetBackup speeds when 8 data buffers of 64KB and 256KB sizes are executed to both encryption enabled and non-encrypting T9840D drives.

Using 8 Data Buffers (MB/sec)	**Data Buffer Transfer Size**			
File Size of Data Set	**64KB (No Enc)**	**64KB Encrypted**	**256KB (No Enc)**	**256KB Encrypted**
32KB - uncompressible data	**5.765**	**5.752**	**5.763**	**5.761**
32KB - compressible ~2:1	**5.805**	**5.811**	**5.806**	**5.806**
256KB - uncompressible data	**14.396**	**14.296**	**18.042**	**18.122**

256KB - compressible ~2:1	**14.565**	**14.535**	**18.523**	**18.474**
1MB - uncompressible data	**17.032**	**17.019**	**28.349**	**28.450**
1MB - compressible ~2:1	**16.835**	**16.840**	**45.641**	**45.869**
1GB - uncompressible data	**16.653**	**16.646**	**28.496**	**28.386**
1GB - compressible ~2:1	**16.596**	**16.681**	**63.239**	**63.142**
10GB - compressible ~10:1	**16.720**	**16.794**	**64.128**	**64.142**

This table shows average NetBackup speeds when 4 data buffers of 64KB and 256KB sizes are executed to both encryption enabled and non-encrypting T9840D drives.

Using 4 Data Buffers (MB/sec)	**Data Buffer Transfer Size**			
File Size of Data Set	**64KB (No Enc)**	**64KB Encrypted**	**256KB (No Enc)**	**256KB Encrypted**
32KB - uncompressible data	**5.746**	**5.724**	**5.770**	**5.758**
32KB - compressible ~2:1	**5.800**	**5.798**	**5.790**	**5.790**
256KB - uncompressible data	**8.946**	**8.939**	**18.092**	**18.121**
256KB - compressible ~2:1	**9.084**	**9.059**	**18.510**	**18.454**
1MB - uncompressible data	**11.466**	**11.518**	**28.470**	**28.313**
1MB - compressible ~2:1	**11.713**	**11.699**	**33.012**	**32.985**
1GB - uncompressible data	**12.147**	**12.140**	**28.411**	**28.422**
1GB - compressible ~2:1	**12.273**	**12.261**	**61.440**	**61.389**
10GB - compressible ~10:1	**12.378**	**12.377**	**63.780**	**63.683**

The Benefit of Multiplexing

The T9840D tape drive was designed for quick access to data rather than rapid transfer of data. Therefore, most servers will be able to push data to it fast enough to attain streaming speed, avoiding the performance degradation caused by stopping and starting the drive. The key to streaming data efficiently is using the preferred block size of 256KB for data transfers to the drive.

Using Multiple Write Drives

Using multiple write drives entails splitting a backup into multiple data streams and concurrently sending these data streams to several tape drives. For lower speed drives like the T9840D, this technique can be useful to reduce the time required to do a single large backup.

Applications that generate nearline data (i.e., data to which users require quick access), often produce such data in large quantities. For example, medical or check imaging software may generate huge quantites of data that must be securely stored for long periods of time. This data is retrieved infrequently, or possibly never. Economics requires that this data be stored on less expensive storage. However, users must have relatively quick access to the data when retrieval is required. The T9840D is designed to provide secure, economical long-term storage and rapid retrieval. Using multiple write drives to speed backups of these large data sets can significantly reduce backup windows and offset the drive's lower transfer rate.

For steps on configuring multiple write drives, please reference your backup application's system administrator's guide.

Chapter 8: HP LTO4 Tape Drive Characterization

Characterization Overview

The purpose of this chapter is to provide some baseline characterization data on how the HP LTO4 encryption capable tape drive performs in an emulated customer environment. This is not intended to be a systems level benchmark and as a result of this, all characterization tests are performed at a file system level. Customer emulated data of varying file sizes are used to most accurately characterize performance within typical real world configurations. The test results in this section were derived from lab testing using Veritas Netbackup on Solaris operating systems but the performance concepts as well as the numbers obtained within can also be applied to other configurations as well.

Data Buffer Transfers

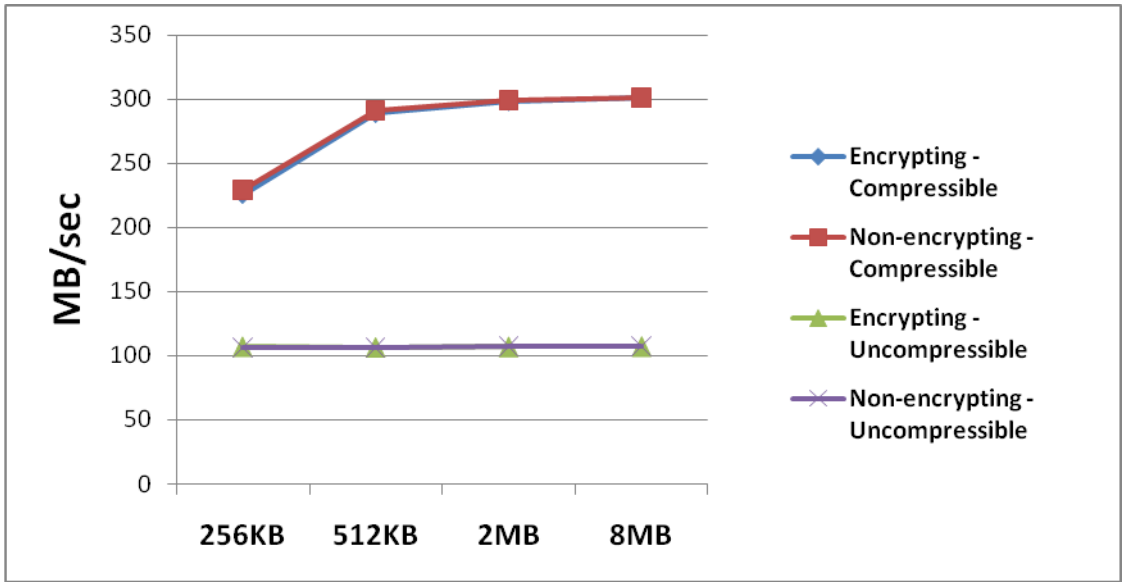
The HP LTO4 drive has an advertised native transfer rate of 120 MB/s and stores up to 800 GB of non-compressible data per cartridge. The HP LTO4 tape drive can support block size transfers of up to 8 megabytes. This means that when configuring applications to utilize the drive, data buffers can be set as high as 8MB. It is advantageous to utilize large data transfers to tape as this will enable the HP LTO4 drive to more easily achieve streaming speeds.

One thing to note when setting data transfer sizes is that other tape drives, including the STK 9x40 series, support block size transfers only up to 256KB. If HP LTO4s are being installed in a mixed drive environment that includes some of these drives, then it is recommended to set the size of the data buffer transfers to 256KB. If the HP LTO4 drives are not in a mixed drive environment or are mixed with other tape drives that also support large block size transfers, then it is recommended to set the size of the data buffer transfers to at least 2MB. Archiving scenarios can benefit from a data buffer size as large as 2MB.

The number of data buffers sent to tape is also configurable in Netbackup and many other backup applications. This can be described as the number of concurrent threads that are sent to the HP LTO4 tape drive. Increasing the number of data buffers can often improve tape performance but will require more system memory.

Lab Performance Data

The graph on the next page shows characterization data captured using HP LTO4 tape drives in both encrypting and non-encrypting modes. Data was obtained using single stream backups of verifiable data files. Data sets included both compressible and non-compressible data. As evidenced in the graph, enabling a HP LTO4 tape drive for encryption in the Sun Crypto Key Management Solution has a negligible impact on performance. In fact, the impact on performance is significantly less than the average variance between test runs. As a result of this, some media operations in encrypting mode record a faster time than equivalent operations in non-encrypting mode.

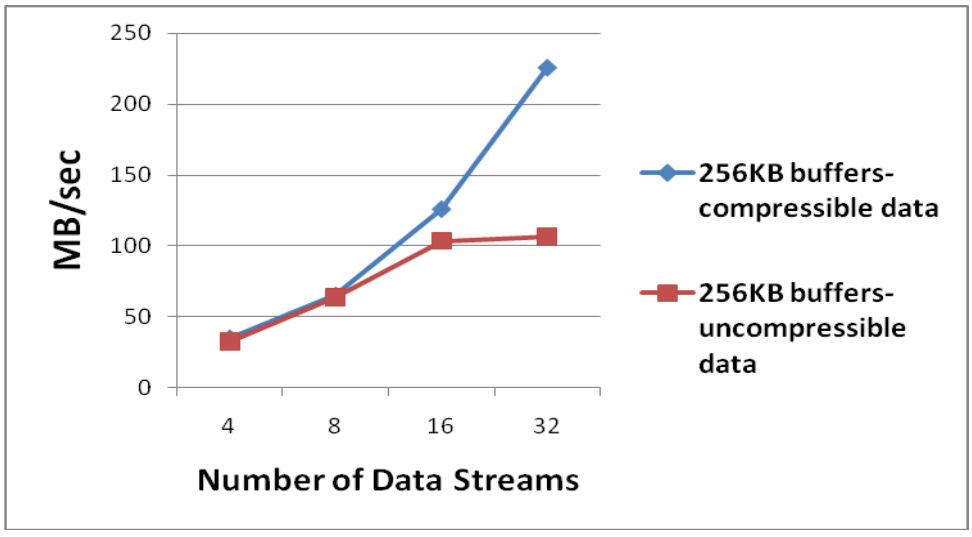


In addition, this graph indicates that the optimal throughput may be obtained with a transfer block size (buffer size) from 512KB to 8MB. The key is the number of streams (threads) running concurrently.

Small Data Buffers

The following graph depicts encrypting HP LTO4 performance when using 256KB data buffer transfers. This example is typical of an environment where HP LTO4 drives are mixed with 9x40 or other drives that do not support large block size transfers and it is necessary to set size data buffers to 256KB. The graph characterizes HP LTO4 encrypted performance across various numbers of data buffers. Numbers on the left show performance characterization while running 4 data buffers (threads) and the graph scales up to 32 data buffers (threads) on the right.

All data is drawn from backup policies executed within Netbackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and non-compressible data sets are show in the graph. All HP LTO4 drives are operating in encrypted mode.

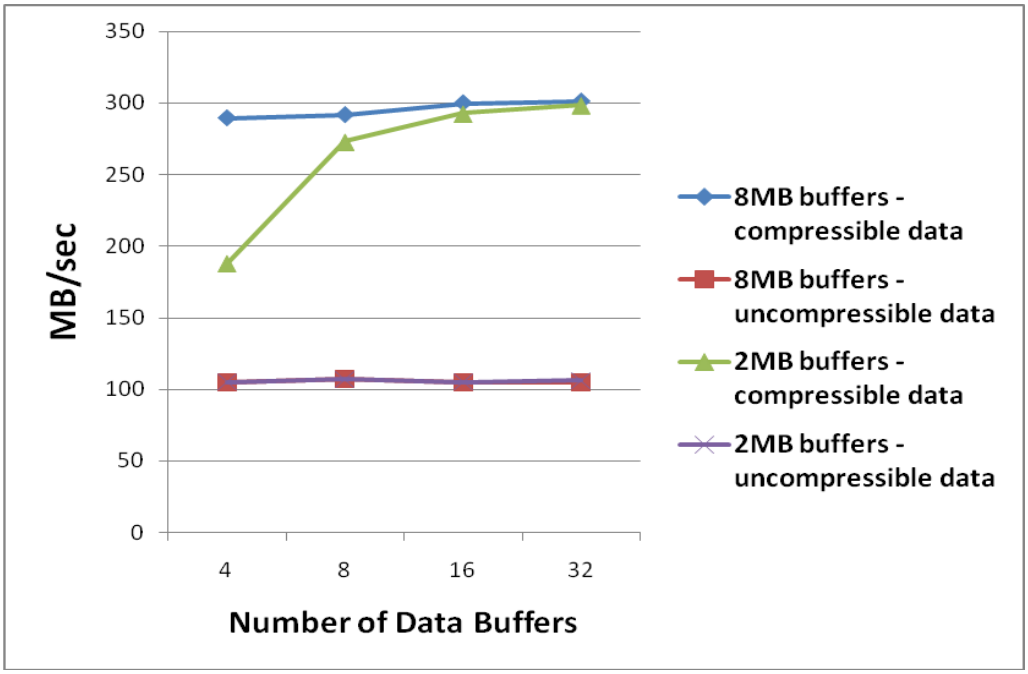


As the graph indicates, when using buffers of size 256KB to write uncompressible data, it is advisable to set the number of data buffers (threads) to at least 16. A marked gain in performance may be achieved by using 32 data buffers when writing compressible data.

Large Data Buffers

The next graph depicts encrypted HP LTO4 performance when using 8MB or 2MB data buffer transfers. This example is typical of a customer environment where HP LTO4 drives are not mixed with other tape drives, or they are mixed with other drives that can support at most 2MB data buffer transfers. The graph characterizes HP LTO4 performance across various numbers of data buffers (threads).

All data is drawn from backup speeds executed within Netbackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and non-compressible data is displayed in the graph. All HP LTO4 drives are operating in encrypted mode.



As this graph shows, when writing uncompressible data optimal throughput can be obtained using only 4 threads regardless of buffer size. Also, only 4 threads with 8MB buffers produces nearly optimal performance when writing compressible data. However, when writing compressible data using 2MB buffers, at least 16 threads must be used to obtain optimal performance.

Characterization Data Charts

All data contained in the following charts was obtained using Netbackup speeds with customer emulated file system data. The hosts and file systems used in these tests were tuned for large sequential IO transfers. Small file transfer speeds could be improved with a file system that was tuned specifically for this. All results show speeds of a single stream data set from just one client. This is done to effectively characterize the HP LTO4 drive. In production situations, any backup that fails to achieve tape streaming speeds would be multiplexed together with other backups to achieve native tape speed. More detailed information is provided on multiplexing later in this chapter.

This first table shows average Netbackup speeds when 32 data buffers of 256KB, 512KB, 2MB and 8MB sizes are executed to both encryption enabled and non-encrypting HP LTO4 drives.

Using 32 Data Buffers (MB/sec)	Data Buffer Transfer Size							
File Size of Data Set	256KB (No Enc)	256KB Encrypted	512KB (No Enc)	512KB Encrypted	2MB (No Enc)	2MB Encrypted	8MB (No Enc)	8MB Encrypted
32KB - non compressible data	5.769	5.757	5.764	5.759	5.767	5.762	5.762	5.762
32KB - compressible ~2:1	5.810	5.814	5.807	5.808	5.809	5.805	5.806	5.801
256KB - non compressible data	18.135	18.133	20.243	20.343	22.231	22.201	22.715	22.716
256KB - compressible ~2:1	18.554	21.735	20.701	20.781	22.732	22.834	23.338	23.392
1MB - non compressible data	46.061	46.000	54.893	54.645	63.521	63.641	67.212	66.986
1MB - compressible ~2:1	46.191	46.159	55.393	55.579	64.618	64.692	67.453	67.965
1GB - non compressible data	105.807	106.979	105.417	105.208	105.974	106.363	107.588	105.041
1GB - compressible ~2:1	209.268	208.964	291.234	285.963	299.218	298.336	301.398	301.396
10GB - compressible ~10:1	229.136	225.623	288.910	288.952	292.475	297.069	301.393	300.998

The following table shows average Netbackup speeds when 16 data buffers of data buffers of 256KB, 512KB, 2MB and 8MB sizes are executed to both encryption enabled and non-encrypting HP LTO4 drives.

Using 16 Data Buffers (MB/sec)	Data Buffer Transfer Size							
File Size of Data Set	256KB (No Enc)	256KB Encrypted	512KB (No Enc)	512KB Encrypted	2MB (No Enc)	2MB Encrypted	8MB (No Enc)	8MB Encrypted
32KB - non compressible data	5.761	5.777	5.758	5.767	5.764	5.971	5.760	5.799
32KB - compressible ~2:1	5.802	5.840	5.799	5.817	5.807	5.828	5.806	5.814
256KB - non compressible data	18.113	15.680	20.237	18.634	22.222	21.972	22.739	22.954
256KB - compressible ~2:1	18.486	15.666	20.730	18.805	22.740	22.238	23.312	23.190

1MB - non compressible data	45.884	30.325	54.581	39.888	63.376	53.137	67.127	58.651
1MB - compressible ~2:1	46.183	31.780	55.501	41.760	64.527	53.477	67.312	58.804
1GB - non compressible data	106.243	103.833	106.065	106.456	105.673	105.318	105.865	104.760
1GB - compressible ~2:1	119.073	118.831	204.748	205.847	279.933	282.900	298.726	299.069
10GB - compressible ~10:1	125.731	126.239	222.210	225.623	288.539	292.394	298.822	300.166

This table shows average Netbackup speeds when 8 data buffers of data buffers of 256KB, 512KB, 2MB and 8MB sizes are executed to both encryption enabled and non-encrypting HP LTO4 drives.

Using 8 Data Buffers (MB/sec)	**Data Buffer Transfer Size**							
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**	**8MB (No Enc)**	**8MB Encrypted**
32KB - non compressible data	5.755	5.755	5.759	5.759	5.761	5.755	5.757	5.753
32KB - compressible ~2:1	5.813	5.781	5.801	5.769	5.811	5.797	5.787	5.789
256KB - non compressible data	18.153	18.151	20.340	20.337	22.205	22.219	22.823	22.811
256KB - compressible ~2:1	18.512	18.511	20.695	20.692	22.750	22.721	23.249	23.377
1MB - non compressible data	45.703	45.701	54.575	54.573	63.376	63.495	65.417	65.822
1MB - compressible ~2:1	46.021	46.020	55.631	55.629	64.636	64.398	67.531	67.358
1GB - non compressible data	63.502	63.955	105.302	105.159	107.482	106.867	105.820	107.199
1GB - compressible ~2:1	64.895	65.346	126.042	126.615	247.133	262.198	295.090	290.312
10GB - compressible ~10:1	64.816	65.211	127.354	126.645	265.499	272.669	295.218	291.807

This table shows average Netbackup speeds when 4 data buffers of data buffers of data buffers of 256KB, 512KB, 2MB and 8MB sizes are executed to both encryption enabled and non-encrypting HP LTO4 drives.

Using 4 Data Buffers (MB/sec)	Data Buffer Transfer Size							
File Size of Data Set	256KB (No Enc)	256KB Encrypted	512KB (No Enc)	512KB Encrypted	2MB (No Enc)	2MB Encrypted	8MB (No Enc)	8MB Encrypted
32KB - non compressible data	5.745	5.753	5.765	5.753	5.751	5.756	5.748	5.755
32KB - compressible ~2:1	5.799	5.805	5.798	5.800	5.787	5.798	5.781	5.808
256KB - non compressible data	18.136	18.129	20.176	20.284	22.238	22.230	22.672	22.777
256KB - compressible ~2:1	18.399	18.414	20.672	20.682	22.703	22.760	23.325	23.373
1MB - non compressible data	30.496	30.469	51.141	51.082	63.056	62.865	65.611	65.379
1MB - compressible ~2:1	32.825	32.807	51.655	51.568	64.715	64.665	67.223	67.252
1GB - non compressible data	32.601	32.717	63.560	63.750	105.692	105.233	105.824	105.100
1GB - compressible ~2:1	34.559	35.181	64.820	64.786	184.783	185.264	283.218	283.602
10GB - compressible ~10:1	34.814	34.485	64.878	65.109	185.700	187.690	285.097	289.473

The Benefit of Multiplexing

Media multiplexing is the process of sending concurrent backups from one or several clients to a single storage device. Multiplexing can serve to reduce backup windows and stream tape drives that could not normally achieve tape speed with a single backup.

The HP LTO4 attempts to match the tape speed to the rate of data coming in from the host to allow streaming of data to the tape at lower throughputs. However, there is a limit to the drive's ability to adapt. When client processing power, network throughput or source disk performance does not allow the application to push data to the tape drive as fast as it can accept it, it is beneficial to use media multiplexing to send several concurrent backup streams to a single drive. This will serve to reduce backup window times by increasing tape performance and limiting time that backup jobs are waiting for resources. When utilizing multiplexing, best performance will be seen when multiplexing data streams from separate clients that use separate source disk.

Multiplexing increases the load on the media server and requires more memory allocation. Because of this, it is recommended to reduce the number of data streams (threads) setting that is used to write to tape. If moderate multiplexing (2-6 concurrent streams) is regularly used, then it is recommended

to reduce the number of data streams setting by one half (i.e. 16 to 8). If heavy multiplexing (8+ streams) is regularly used, then it is recommended to reduce the number of data streams setting by three quarters (i.e. 16 to 4). There is no special action that is needed to restore a multiplexed backup but it does negatively impact restore performance as the backup application is required to sort through data from multiple backups.

Since Sun Tape Encryption is managed at a device level, there are no issues with multiplexing several backups in an encrypted environment. For steps on configuring multiplexing, please reference your backup application's system administrator's guide.

Using Multiple Write Drives

Using multiple write drives entails splitting a backup into multiple data streams and concurrently sending these data streams to several tape drives. This is beneficial in situations where a single large database that resides on a powerful server and very fast disk array is required to be backed up.

Due to the high speed of the HP LTO4 drives, situations that call for multiple write drives are rare. To effectively utilize multiple HP LTO4 drives for a single large backup, the source disk, media server and SAN need to be capable of very high throughput. Multiple write drives can shrink the time it takes to do a single lengthy backup and serve to reduce backup windows.

For steps on configuring multiple write drives, please reference your backup application's system administrator's guide.

Chapter 9: T10000B Tape Drive Characterization

Characterization Overview

The purpose of this chapter is to provide some baseline characterization data on how the T10000B encryption capable tape drive performs in an emulated customer environment. This is not intended to be a systems level benchmark and as a result of this, all characterization tests are performed at a file system level. Customer emulated data of varying file sizes are used to most accurately characterize performance within typical real world configurations. The test results in this section were derived from lab testing using Veritas NetBackup on Solaris operating systems but the performance concepts as well as the numbers obtained within can also be applied to other configurations as well.

An emphasis is placed on comparing performance in encrypting and non-encrypting modes. A drawback of many encryption models is a significant performance impact. The Sun StorageTek Encryption Solution has a competitive advantage in this regard.

Data Buffer Transfers

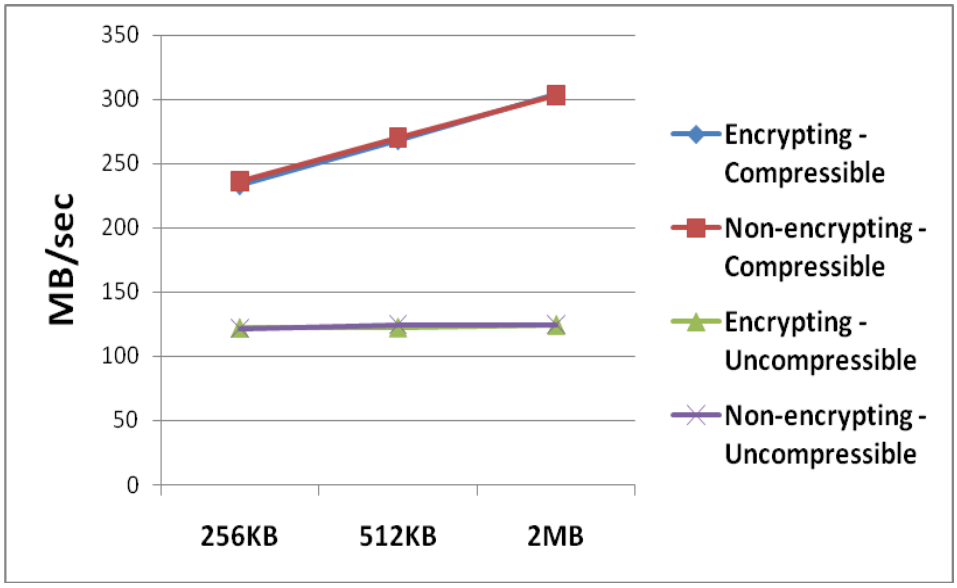
The T10000B tape drive can support block size transfers of up to 2 megabytes. This means that when configuring applications to utilize the drive, data buffers can be set as high as 2MB. It is advantageous to utilize large data transfers to tape as this will enable the high speed T10000B to more easily achieve streaming speeds.

One thing to note when setting data transfer sizes is that other tape drives, including the STK 9x40 series, only support block size transfers up to 256KB. If T10000Bs are being installed in a mixed drive environment that includes some of these drives, then it is recommended to set the size of the data buffer transfers to 256KB. If the T10000B drives are not in a mixed drive environment or are mixed with other tape drives that also support large block size transfers, then it is recommended to set the size of the data buffer transfers to at least 512KB. Archiving scenarios can benefit from a data buffer size as large as 2MB.

Lab Performance Data

The graph below shows characterization data captured using T10000B tape drives in both encrypting and non-encrypting modes. Data was obtained using single stream backups of verifiable data files. Data sets included both compressible and uncompressible data. As evidenced in the graph, enabling a T10000B tape drive for encryption in the Sun Crypto Key Management Solution has a negligible impact on performance. In fact, the impact on performance is significantly less than the average variance between test runs. As a result of this, some media operations in encrypting mode record a faster time than equivalent operations in non-encrypting mode.

Sun's encryption solution adds a fixed amount of 100 bytes overhead per data buffer transferred. Backup and archiving applications should be configured for a minimum of 256KB transfer size. In this model, encryption would impact performance by only 0.04%. In other configurations, the impact would be even less. The encryption algorithm occurs at the device level after the T10000B has already compressed the data for storage, allowing the maximum benefit of increased speed and capacity due to compression to be realized.



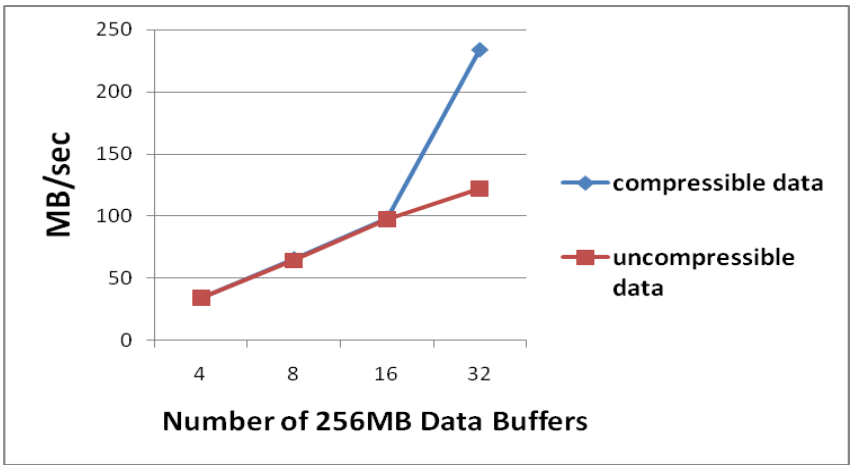
Native data transfer rate for the T10000B is 120MB/sec. This is observed in the results of our tests using uncompressible data sets. When compression is utilized, performance and capacity can be greatly increased.

Small Data Buffers

The following graph depicts encrypting T10000B performance when using 256KB data buffer transfers. This example is typical of an environment where T10000B drives are mixed with 9x40 or other drives that do not support large block size transfers and it is necessary to set size data buffers to 256KB. The graph characterizes T10000B encrypted performance across various numbers of data buffers. Numbers on the left show performance characterization while running 4 data buffers (threads) and the graph scales up to 32 data buffers (threads) on the right.

All data is drawn from backup policies executed within NetBackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and uncompressible data sets are show in the graph. All T10000B drives are operating in encrypted mode.

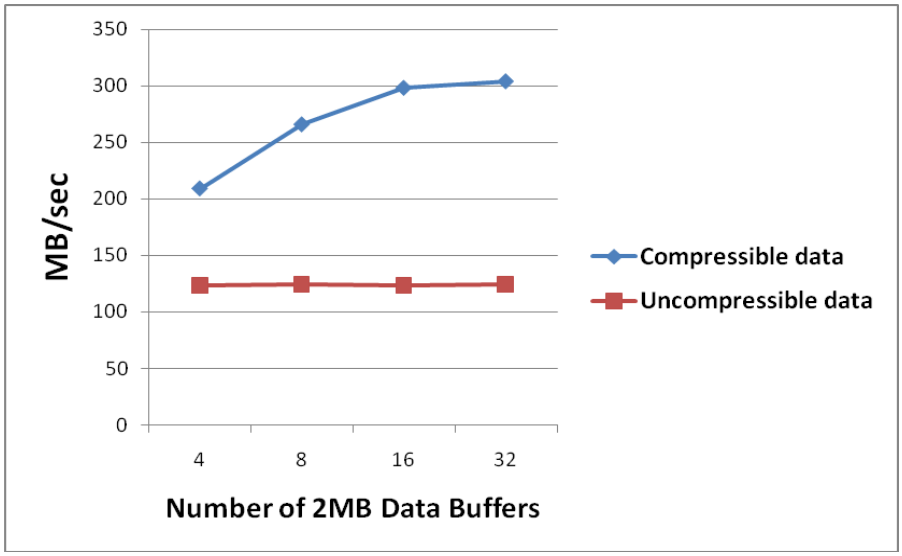
As the graph indicates, when writing data in 256KB blocks, setting the number of data buffers to 32 is advisable.



Large Data Buffers

This second graph depicts encrypted T10000B performance when using 2MB data buffer transfers. This example is typical of a customer environment where T10000B drives are not mixed with other tape drives, or they are mixed with other drives that can also support 2MB data buffer transfers. The graph characterizes T10000B encrypted performance across various numbers of data buffers (threads).

All data is drawn from backup speeds executed within NetBackup. Data sets are comprised of file system data with customer emulated files. The difference between compressible and uncompressible data is displayed in the graph. All T10000B drives are operating in encrypted mode.



As this graph shows, when writing uncompressible data in 2MB blocks, 4 data buffers (threads) is sufficient to obtain optimal throughput. However, at least 16 data buffers are required to obtain optimal throughput when writing compressible data. Use of 2MB buffers is recommended for archiving solutions or other operations where a customer is writing large data files to tape.

Characterization Data Charts

All data contained in the following charts was obtained using NetBackup speeds with customer emulated file system data. The hosts and file systems used in these tests were tuned for large sequential IO transfers. Small file transfer speeds could be improved with a file system that was tuned specifically for this. All results show speeds of a single stream data set from just one client. This is done to effectively characterize the T10000B drive. In production situations, any backup that fails to achieve tape streaming speeds would be multiplexed together with other backups to achieve native tape speed. More detailed information is provided on multiplexing later in this chapter.

This first table shows average NetBackup speeds when 32 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000B drives.

Using 32 Data Buffers (MB/sec)	**Data Buffer Transfer Size**					
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**
32KB - uncompressible data	5.774	5.811	5.772	5.818	5.767	5.828
32KB - compressible ~2:1	5.829	5.876	5.828	5.876	5.818	5.883
256KB - uncompressible data	18.242	19.343	20.373	21.706	22.496	24.150

256KB - compressible ~2:1	18.614	19.771	20.791	22.328	22.954	24.838
1MB - uncompressible data	46.294	46.287	55.273	54.754	64.633	62.873
1MB - compressible ~2:1	46.154	45.927	55.644	54.634	65.083	62.990
1GB - uncompressible data	121.725	122.137	123.294	122.368	123.456	124.508
1GB - compressible ~2:1	217.357	218.741	267.064	268.038	300.349	302.665
10GB - compressible ~10:1	236.088	233.456	269.892	268.483	303.051	303.722

This table shows average NetBackup speeds when 16 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000B drives.

Using 16 Data Buffers (MB/sec)	**Data Buffer Transfer Size**					
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**
32KB - uncompressible data	5.826	5.855	5.805	5.855	5.814	5.859
32KB - compressible ~2:1	5.869	5.919	5.849	5.905	5.868	5.922
256KB - uncompressible data	19.435	19.346	21.695	21.785	23.965	24.134
256KB - compressible ~2:1	19.792	19.882	22.357	22.369	24.735	24.836
1MB - uncompressible data	46.168	46.251	54.867	54.608	63.002	63.070
1MB - compressible ~2:1	46.132	46.093	54.610	54.425	62.859	63.027
1GB - uncompressible data	98.556	97.551	124.424	122.204	123.424	123.280
1GB - compressible ~2:1	98.265	97.873	206.190	207.386	295.549	297.922
10GB - compressible ~10:1	97.932	98.235	222.060	219.700	296.098	297.857

This table shows average NetBackup speeds when 8 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000B drives.

Using 8 Data Buffers (MB/sec)	**Data Buffer Transfer Size**					
File Size of Data Set	**256KB (No Enc)**	**256KB Encrypted**	**512KB (No Enc)**	**512KB Encrypted**	**2MB (No Enc)**	**2MB Encrypted**
32KB - uncompressible data	5.818	5.755	5.816	5.759	5.812	5.755
32KB - compressible ~2:1	5.853	5.781	5.862	5.769	5.863	5.797
256KB - uncompressible data	19.307	18.151	21.692	20.337	23.849	22.219
256KB - compressible ~2:1	19.832	18.511	22.278	20.692	24.628	22.721
1MB - uncompressible data	45.923	45.701	54.843	54.573	62.828	63.495
1MB - compressible ~2:1	46.895	46.020	54.682	55.629	62.514	64.398
1GB - uncompressible data	64.983	64.454	121.233	121.303	124.339	124.263
1GB - compressible ~2:1	65.210	65.382	125.621	126.011	249.427	255.037
10GB - compressible ~10:1	65.378	65.330	128.583	127.515	273.499	265.580

The following table shows average NetBackup speeds when 4 data buffers of 256KB, 512KB and 2MB sizes are executed to both encryption enabled and non-encrypting T10000B drives.

Using 4 Data Buffers (MB/sec)	Data Buffer Transfer Size					
File Size of Data Set	256KB (No Enc)	256KB Encrypted	512KB (No Enc)	512KB Encrypted	2MB (No Enc)	2MB Encrypted
32KB - uncompressible data	5.807	5.753	5.802	5.753	5.792	5.756
32KB - compressible ~2:1	5.851	5.805	5.860	5.800	5.855	5.798
256KB - uncompressible data	19.399	18.129	21.689	20.284	23.990	22.230
256KB - compressible ~2:1	19.729	18.414	22.320	20.682	24.612	22.760
1MB - uncompressible data	32.968	30.469	51.925	51.082	62.981	62.865
1MB - compressible ~2:1	33.011	32.807	51.916	51.568	62.890	64.665
1GB - uncompressible data	34.378	34.484	64.420	64.242	123.067	123.605
1GB - compressible ~2:1	34.103	34.672	64.469	64.391	190.977	195.926
10GB - compressible ~10:1	34.090	34.456	64.582	65.162	206.896	208.910

The Benefit of Multiplexing

Media multiplexing is the process of sending concurrent backups from one or several clients to a single storage device. Multiplexing can serve to reduce backup windows and stream tape drives that could not normally achieve tape speed with a single backup.

The T10000B is a high speed tape drive and in many circumstances client processing power, network throughput or source disk performance will not be great enough to achieve streaming speeds. When this happens, tape drives are forced to start and stop as data cannot be supplied as fast as the drive can write. The T10000B buffer size is chosen so that any start-stop activity does not impact system performance and, unlike mid-range drives, the T10000B mechanism is designed to operate in a start-stop environment with no degradation in reliability. Under these circumstances it is beneficial to use media multiplexing to send several concurrent backup streams to a single T10000B. This will serve to reduce backup window times by increasing tape performance and limiting time that backup jobs are waiting for resources. It will also reduce wear on the tape drive and media. When utilizing multiplexing, best performance will be seen when multiplexing data streams from separate clients that use separate source disk.

Multiplexing increases the load on the media server and requires more memory allocation. Because of this, it is recommended to reduce the number of data streams (threads) setting that is used to write to tape. If moderate multiplexing (2-6 concurrent backups) is regularly used, then it is recommended to reduce the number of data streams setting by one half (i.e. 16 to 8). If heavy multiplexing (8+ concurrent backups) is regularly used, then it is recommended to reduce the number of data streams setting by three quarters (i.e. 16 to 4). There is no special action that is needed to restore a multiplexed backup but it does negatively impact restore performance as the backup application is required to sort through data from multiple backups.

Since Sun Tape Encryption is managed at a device level, there are no issues with multiplexing several backups in an encrypted environment. For steps on configuring multiplexing, please reference your backup application's system administrator's guide.

Using Multiple Write Drives

Using multiple write drives entails splitting a backup into multiple data streams and concurrently sending these data streams to several tape drives. This is beneficial in situations where a single large database that resides on a powerful server and very fast disk array is required to be backed up.

Due to the high speed of the T10000B drives, situations that call for multiple write drives are rare. To effectively utilize multiple T10000B drives for a single large backup, the source disk, media server and SAN need to be capable of very high throughput. Multiple write drives can shrink the time it takes to do a single lengthy backup and serve to reduce backup windows.

For steps on configuring multiple write drives, please reference your backup application's system administrator's guide.

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA Phone 1-650-960-1300 or 1-800-555-9SUN Web sun.com

Part Number: xxxx 08/04/2008