Encryption Key Management:
A Technical White Paper
CONTENTS

Introduction .................................................................................................................. 3
Encryption and Encryption Keys Overview ................................................................. 3
Quantum Encryption Key Manager (Q-EKM) Overview .............................................. 4
Q-EKM Advantages over Competitive Key Management Solutions ......................... 5
Using Q-EKM .............................................................................................................. 5
Q-EKM Deployment and Library Setup ....................................................................... 7
Backing Up Q-EKM ...................................................................................................... 9
Quantum Key Management Best Practices ............................................................... 9
Conclusion ................................................................................................................ 10
Introduction

Quantum’s Encryption Key Manager (Q-EKM) is a centralized key manager application that manages the encryption keys used as part of the LTO Ultrium 4 (LTO-4) drive-based data encryption process. Support for the EKM application is an optional, licensed feature that must be enabled from the library in order to begin encrypting data using the LTO-4 tape drive encryption capabilities.

Encryption and Encryption Keys Overview

An encryption key is a random string of bits generated specifically to encrypt and decrypt data. Encryption keys are created using algorithms designed to ensure that each key is unique and unpredictable. Using keys of greater length provides encryption of greater strength (i.e., they are more secure). All LTO-4 (Fibre Channel and SAS) tape drives use a method of encryption based on 256 bit AES algorithm keys to encrypt the data. The 256 bit AES is the encryption standard currently recognized and recommended by the U.S. government for even “Top-Secret” data.

"The design and strength of all key lengths of the AES algorithm (i.e., 128, 192 and 256) are sufficient to protect classified information up to the SECRET level. TOP SECRET information will require use of either the 192 or 256 key lengths. The implementation of AES in products intended to protect national security systems and/or information must be reviewed and certified by NSA prior to their acquisition and use." — [1]

Two types of encryption algorithms are used by Q-EKM: symmetric algorithms and asymmetric algorithms. Symmetric algorithms involve a key known by both sides; asymmetric algorithms involve a combination of a ‘public’ key and a ‘private’ key. Symmetric, or secret key encryption, uses a single key for both encryption and decryption. Symmetric key encryption is generally used for encrypting large amounts of data in an efficient manner. The 256-bit AES keys are symmetric keys. Encryption keys generated by Q-EKM are transferred to the Library “out-of-band” (i.e., not via the data path used to backup the customer’s data). Applications that support encryption by using the standard SCSI commands SECURITY PROTOCOL IN (SPIN) and SECURITY PROTOCOL OUT (SPOUT) for example, IBM’s Tivoli Storage Manager and CommVault’s Galaxy, and in the future Symantec’s NetBackup, EMC’s Networker, etc. create one or more AES encryption keys natively and transfer them to the LTO-4 tape drive “in-band” just ahead of the backup data going to the LTO-4 drive. The out-of-band methodology for transfer is preferable as it allows for a centralized key management application like Q-EKM to support multiple data storage systems and has no “same system” requirements for restoration (i.e., Some encryption solutions limit the customer to restoring through the same system setup that wrote the original tape cartridge). There is no performance penalty for using out-of-band versus in-band encryption key management.

For LTO-4 tape encryption, unencrypted data is sent to the LTO-4 tape drive and converted to encrypted data (ciphertext) using a pre-generated symmetric Data Key (DK) that is sent from Q-EKM’s key store, and is then written to tape. Q-EKM selects a pre-generated Data Key in round robin fashion. Data Keys are reused on multiple tape cartridges after each pre-generated key has been used at least once. The Data Key is sent to the LTO-4 tape drive in encrypted (“wrapped”) form by Q-EKM. The LTO-4 tape drive decrypts (“unwraps”) this Data Key and uses it to perform encryption or decryption. However, no key is stored anywhere on the LTO-4 tape cartridge. Once the encrypted volume is written, the Data Key must be accessible and available to Q-EKM in order for the volume to be read. A key “alias” or label is used for this purpose. Figures 1 and 2 on page 1 illustrates this process.

(1) http://www.cnss.gov/Assets/pdf/cnssp_15_fs.pdf
Only one Data Key per tape cartridge is allowed (used). The LTO-4 tape drive attempts to apply encryption uniformly across a tape. Any subsequent data added to a tape cartridge will only be encrypted and written if the LTO-4 drive being used to write the new data has access to the same Data Key that was used to encrypt the previous data. If the Data Key is not available, the data will not be written, and an error will be reported. If data is added to a tape cartridge where the existing data is unencrypted, the drive will not allow any data to be written encrypted. If the tape is an unencrypted scratch tape, then the data has to be written from Beginning of Tape (BOT) for it to be written encrypted. Tapes that have been labeled by an application without being encrypted will not encrypt data on subsequent data write operations.

**Quantum Encryption Key Manager (Q-EKM) Overview**

Q-EKM is a Java application that generates, protects, stores, and maintains encryption keys for the LTO-4 tape drives in Quantum’s Scalar i2000 and Scalar i500 tape libraries. These keys are used by the LTO-4 tape drive to encrypt information being written to, and decrypt information being read from, tape media. The encryption keys are sent via the library-to-drive interface; therefore encryption is transparent to the application writing the data. Q-EKM is designed to generate and communicate encryption keys for LTO-4 tape drives in Quantum libraries throughout the customer’s environment. The illustration below shows Q-EKM’s four main components (Configuration File, Drive Table, Keystore and the Key Group XML File).
Q-EKM Advantages over Backup Application provided Key Management Solutions

- Q-EKM is an Enterprise Encryption Key manager that was designed from the ground up for that purpose. ISV key management support is very limited today and requires the customer to setup, manage and backup encryption keys for each ISV application instance in their environment.

- Most ISV applications currently use password or pass phrases as keys to encrypt data. While the encrypted tape is AES 256 encrypted, passwords and pass phrases tend to be written down or saved using other unsecured methods.

- Using Q-EKM, security policies may be managed separately from backup policies. Backup administrators or operators need not be burdened with managing data encryption processes – this role may be turned over to a security officer or manager, providing another layer of protection.

Using Q-EKM

Figure 3. EKM’s four main components

Figure 4 – Q-EKM Architecture
Figure 4 shows a library managed encryption solution using a Scalar i2000 library, containing encryption-enabled LTO-4 Tape Drives. The library facilitates the encryption policy and passes all key requests between the LTO-4 dives and Q-EKM. This library is connected to a Q-EKM server using a standard IP network attached to the i2000’s management interface. Q-EKM runs on a standard Windows or Linux server and manages the creation and storage of the encryption data keys.

When an LTO-4 tape drive in the library requests an encryption data key the Q-EKM retrieves an existing key from the keystore and wraps (encrypts) it for secure transfer to the tape drive. When the key arrives at the LTO-4 drive, it is unwrapped (decrypted) and used to encrypt the data being written to tape. When an encrypted tape is read by an LTO-4 tape drive, Q-EKM retrieves the required key from the keystore, based on the information in the Key ID on the tape, and sends it securely to the tape drive where it is unwrapped and used to allow the tape drive to decrypt the data.

The following simple example explains the Q-EKM key management process in more detail:

Once Q-EKM is installed and configured on its server, it needs to be set up on each library or libraries by configuring the libraries for the IP address of the server where the Q-EKM software resides (this is explained further in the next section). Once this is complete, here’s how the encryption key management process works:

1. The backup application sends a command to the drive to write data onto Media Cartridge # 1 (for example).

2. The drive requests a key (via the library) from Q-EKM located at the IP address that was configured in the library.

3. Q-EKM returns Key # 1 to the drive (via the library).

4. The drive encrypts the backup data and writes the encrypted data to Media Cartridge # 1. At this point, a relationship is created between Media Cartridge # 1 and Key # 1. This relationship is stored within the tape header and within the Q-EKM database, and is based on the media identifier stored on LTO-4 tape header KAD (key associated data) field.

5. When Media Cartridge # 1 is returned to its original storage slot or it is removed from the tape library, the data on that cartridge is encrypted. The correct key (Key # 1 stored at the IP address entered above) is needed in order to decrypt and read the data stored on Media Cartridge # 1.

6. When a restore is required from Media Cartridge # 1, a read command is sent from the backup application. Media Cartridge # 1 is loaded into an LTO-4 drive, and the drive, upon recognizing this piece of media as Media Cartridge # 1, requests the correct key from Q-EKM to decrypt the data on Media Cartridge # 1. The drive sends this request to the library, which in turn sends it to the IP address of the Q-EKM server. Upon receiving the request, Q-EKM sends the correct key - Key # 1 in this example.

7. Once the drive receives Key # 1 (the key that matches Media Cartridge # 1), it can then decrypt the data to support a restore.
Q-EKM Deployment and Library Setup

Q-EKM is designed to be installed by Quantum or Quantum’s trained partners. It is deployed on a customer-provided Xeon-class server with at least 1GB of RAM, 10GB of free disk space, running Windows 2003 Server or Red Hat Enterprise Linux 4. The Server must have IP connectivity to all Quantum libraries for which it will manage encryption keys. This server should be protected and backed up in the data center following your company’s data protection practices (but not using encrypted tapes served by this Q-EKM). More information is available in the Configuration section below.

Q-EKM may be set up in a Normal or Advanced configuration. Quantum recommends that the Q-EKM be installed in the High Availability configuration where possible.

The Scalar i2000 and Scalar i500 tape libraries require a Q-EKM license that is sold based on the library size (i500) or number of drives (i2000). The Scalar i2000 and Scalar i500 libraries each have two Q-EKM setup screens that are used to enable encryption key management support. Once the correct license has been entered into the library, the IP address and port number of the Primary Q-EKM server (and optionally the Secondary Q-EKM server) must be provided via the Setup - EKM Server Access page. A sample screen shot from the Scalar i500 user interface showing this step in figure 5:

Q-EKM support can be enabled on a per partition basis in the Scalar i500 and Scalar i2000 as shown in the example Scalar i500 screen shot (figure 6):

In the Setup – Partition Encryption page, each partition that contains an LTO-4 drive can be set for either Library Managed Encryption (LME) or Application Managed Encryption (AME). The LME option is used when connecting your Library to Q-EKM. To connect to Q-EKM, a partition MUST BE set for LME. To disconnect a partition from EKM so that data is no longer encrypted or to use the backup application to manage the LTO-4 encryption and keys, a partition needs to be set to AME mode. The customer’s backup application must have LTO-4 encryption support enabled to encrypt data when using AME mode, otherwise no data is encrypted.
Basic Configuration – Using Single Q-EKM Server

A single server configuration, shown in Figure 7, is the most basic Q-EKM configuration. However, because of the lack of redundancy it is not a recommended configuration. In this configuration, all tape drives rely on a single key manager server. Should the server go down, Q-EKM, the keystore, configuration file, key groups XML file, and drive table would be unavailable, preventing any data from being written and any encrypted data from being read.

In a single server configuration you must ensure that backup copies of the keystore, configuration file, key groups XML file, and drive table are maintained in a safe place, separate from Q-EKM, so its function can be restored on a replacement server if the original copies are lost. If the keys are lost, all data encrypted with those keys will be unrecoverable.

Advanced Configuration – Synchronized Q-EKM Servers

The high availability configuration option is designed to keep two Q-EKMs synchronized. This is the recommended configuration, both for its inherent backup of critical data and also for its failover capability to avoid any outages in your tape operations. The library will automatically failover to the secondary key manager should the primary be inaccessible for any reason.

It is critical to note that when different Q-EKM servers are used to handle requests from the same set of tape drives, the information in the associated keystores MUST be identical. This is required so that regardless of which key manager server is contacted, the necessary information is available to support requests from the tape drives.

In an environment with two Q-EKM servers having identical configurations, such as those shown in Figure 8, the library will automatically failover to the secondary key manager should the primary become inaccessible. In such a configuration it is essential that the two key manager servers be synchronized.
**Backing Up Q-EKM**

Q-EKM and its associated data must be backed up regularly, and without being encrypted. Additionally, encrypted Save or Archive operations must not be performed on the system where the Q-EKM server is running.

**Exporting Keys** *(Supported in a Future Release)*

Q-EKM supports the exporting of keys. This is usually done to share them with another site that needs to restore tapes encrypted with those keys (e.g. other company location or partner/hosting location).

**Importing Keys** *(Supported in a Future Release)*

Q-EKM supports the importing of keys from another Q-EKM installation. This is usually done to read shared data from another site (Partner, Vendor or Customer) that needs to read tapes encrypted with those keys.

**Adding Keys**

Q-EKM is installed with a default keystore of 1000 keys. Those keys are rotated and reused as needed. Adding additional keys will be supported in a future release.

**Removing/Expining Keys** *(Supported in a Future Release)*

**Quantum Key Management Best Practices**

Below are some general guidelines that Quantum advises for data center best practices:

- Q-EKM should be installed on a server that is separate from your backup application
- Q-EKM should be installed and set up in an Advanced (High Availability) configuration with synchronized keystores.
- Procedures should be put in place to determine:
  - Who has access to the Q-EKM server
  - Where should the primary and secondary Q-EKM servers be located
  - Who can create keys
  - Who is responsible for backing up the Q-EKM keystore
  - That the keystore is not backed up to any library for which it provides keys
  - Where the backup is located
  - What data needs to be encrypted
  - What is the procedure to protect the Q-EKM keystore should someone leave on less than good terms
  - That a disaster recovery process is in place should the keystore need to be restored, or tapes need to be read from an alternate site (as applicable)
Conclusion

Quantum’s Q-EKM provides industry-leading, transparent key management for encryption on LTO-4 based Quantum Libraries. Quantum believes that Q-EKM provides enterprise class encryption key management, security and scalability for customer’s data at rest on LTO-4 Tapes. Q-EKM provides maximum flexibility while still being a cost effective solution for Quantum’s customers.