

WHITEPAPER

Encryption Methods for Protecting Data



Introduction

Within the last ten years, there has been a vast increase in the accumulation and communication of digital computer data in both the private and public sectors. Much of this information has a significant value, either directly or indirectly, that requires protection.

It is common to find data transmissions equating to billions of dollars transferred daily. At all company levels from small to corporate, sensitive information concerning individuals, finances, product or business developments is held and processed in computer systems.

The development of large centralized storage repositories such as enterprise disk subsystems and tape silos increases the potential threats to personal and corporate privacy. Since data banks often are accessed from remote computer terminals, there is a threat of easy and unauthorized access to sensitive information from any place in the data communications system.

A large amount of effort has been put into place to protect "on-line", centrally held data i.e. in an enterprise disk subsystem, as this is seen as the highest access environment. However, despite the increasing awareness of the Information Technology industry of the importance of backup, and the cost benefits of archiving data to tape, the potential weakness of the security of sensitive data while resident on tape is not being addressed.

Encryption is a tool that may be used in a centralized data pool in a tape environment. It is not a panacea; improper implementation and use of data encryption may only provide an illusion of security. Inadequate understanding of encryption applications and data encryption could deter the utilization of other required protection techniques. However, with proper management controls, adequate implementation specifications and applicable usage guidelines, data encryption will not only aid in protecting data communications but can provide protection for a myriad of specific data processing applications.

Where Is The Threat?

It can be generally accepted that at least some proportion of the data held by a company is sensitive, be it to private individuals, members of staff, business rivals or because of the security legislation in the country or community where the business operates.

BOSaNOVA, Inc.

Phone: 866-865-5250 **Email:** info@theq3.com **Web:** www.theq3.com

The proportion of sensitive data is dependent upon the nature of the business in which the company operates. The prime environments where there will be the highest proportion of sensitive data are Research and Financial. Here the data is the company's most guarded asset, such as new product development in a pharmaceutical company, or highly private financial transactions.

All major companies have a backup strategy, so that in the event of a system failure, the data will be available for restore when the failure has been resolved. All large companies perform this operation to tape.

Tape technology has now advanced to the point where 500 GBytes of data can be stored on a cartridge small enough to fit into a suit pocket. Many companies operate stringent security procedures in their data centres; visitors must be with pass holding staff, key code door locks and the like. All of which are subject to abuse resulting in a security mirage. Even if these security measures are implemented they are totally reliant upon the loyalty of staff.

The most serious threat to sensitive data has arisen through the need for true Disaster Recovery policies. In the event of a catastrophe, where the entire IT environment is inaccessible, tapes held locally will not be available for restore at an alternative site. Many companies therefore remove the backup and archive tapes to their Disaster Recovery sites or to a remote storage facility. It is during transit from site to site when the tapes are at the highest level of risk, be it from organized theft or misplacement.

Data tapes which have been encrypted will have a higher level of security should they fall into the public domain or into the hands of competitors.

The cost of a data breach

A recent Ponemon Institute survey of 9,000 people found that 12% of respondents had been notified of a data breach or loss by a company with which they did business. Of those affected, 20% said they immediately stopped doing business with the companies that couldn't keep their data secure. Can you afford to lost 20% of your customers?

BOSaNOVA, Inc.

Phone: 866-865-5250 **Email:** info@theq3.com **Web:** www.theq3.com

What Is Encryption

The word “encryption” has been coined from the word “cryptography” which is derived from the Greek “kryptos” (hidden) and “graphia” (writing). Encryption is the process of transforming text into an unintelligible form called cipher. Data encryption is the process used to hide the true meaning of data.

Reversing the process of encryption is called decryption. Encryption and decryption comprise the science of cryptography as it is applied to the modern computer. Data encryption is achieved through the use of an algorithm that transforms data from its intelligible form to cipher. An algorithm is a set of rules or steps for performing a desired operation. An algorithm can be performed by anything that can be taught or programmed to follow a specific and unambiguous set of instructions.

Methods Of Encryption

There are two types of cryptosystems: secret key and public key. In secret-key cryptography, also referred to as symmetric cryptography, the same key is used for both encryption and decryption. The most popular secret-key cryptosystem in use today is known as DES, the Data Encryption Standard. IBM developed DES in the middle 1970's and it has been a United States of America Federal Standard since 1976.

In public-key cryptography, each user has a public key and a private key. The public key is made public whilst the private key remains secret. Encryption is performed with the public key, while the private key is used for decryption. The RSA publickey cryptosystem is the most popular form of public-key cryptography. RSA stands for Rivest, Shamir, and Adleman, the inventors of the RSA cryptosystem.

In both cases the encryption can be applied in either a block or stream cipher.

Block Cipher

A block cipher is a type of symmetric-key encryption algorithm that transforms a fixed-length block of plaintext (unencrypted text) data into a block of ciphertext (encrypted text) data of the same length. This transformation takes place under the action of a user-provided secret key.

Decryption is performed by applying the reverse procedure to the ciphertext block, whilst using the same secret key. The fixed length is called the block size, and for many block ciphers, the block size is 64 bits. In the coming years the block size will increase to 128 bits as processors become more sophisticated.

Iterated block ciphers encrypt a plaintext block by a process that has several rounds. In each round the same transformation, also known as a round function, is applied to the data using a subkey. The set of subkeys is usually derived from the user-provided secret key by a special function. The set of subkeys is called the key schedule. The number of rounds in an iterated cipher depends on the desired security level and the consequent trade-off with performance. In most cases, an increased number of rounds will improve the security offered by a block cipher, but for some ciphers the number of rounds required to achieve adequate security will be too large for the cipher to be practical or desirable other than in a dedicated hardware environment..

Stream Cipher

A stream cipher is a type of symmetric encryption algorithm. Stream ciphers can be designed to be exceptionally fast, much faster than any block cipher. While block ciphers operate on large blocks of data, stream ciphers typically operate on smaller units of plaintext, usually bits.

The encryption of any particular plaintext with a block cipher will result in the same ciphertext when the same key is used. With a stream cipher, the transformation of these smaller plaintext units will vary, depending on when they are encountered during the encryption process. A stream cipher generates what is called a keystream, which is a sequence of bits used as a key. Encryption is accomplished by combining the keystream with the plaintext, usually with the bitwise exclusive-OR operation. The generation of the keystream can be independent of the plaintext and ciphertext, yielding what is termed a synchronous stream cipher, or it can depend on the data and its encryption, in which case the stream cipher is said to be self-synchronizing. Most stream cipher designs are for synchronous stream ciphers.

What Is DES?

NIST¹ issued the Data Encryption Standard (DES) in 1977 to provide an encryption algorithm for use in protecting federal unclassified information from unauthorized disclosure or undetected modification during transmission, or while in storage. The standard required NIST to conduct a review every five years to determine whether the cryptographic algorithm specified by the standard should be re-affirmed, revised or withdrawn. The first review resulted in the re-affirmation of the

the standard in 1983; the standard was re-affirmed in 1988 following a second review; the third review was completed in 1993. FIPS 46-2, which was issued following the third review, re-affirmed the DES until 1998. In recent times the security of DES has been in question, however no other non-DES standard has been presented. In October 1999 NIST released FIPS 42-3. This reaffirmed DES and at the same time ratified DES3, Triple DES or TDEA as the standard required for the encryption of nonstrategic data. In a recent article, cryptography expert Bruce Schneier responding to an FBI statement on the security of DES and TDEA was quoted "...there isn't enough silicon in the galaxy or enough time before the sun burns out to brute-force triple-DES" (*Crypto-Gram*, Counterpane Systems, August 15, 1998).

The DES is based on work of IBM and has been adopted as the American National Standard X3.92-1981/R1987. The DES is a publicly known cryptographic algorithm that converts plaintext to ciphertext using a 56-bit key. The same algorithm is used with the same key to convert ciphertext back to plaintext, the process called decryption.

The DES consists of 16 "rounds" of operations that mix the data and key together in a prescribed manner using the fundamental operations of permutation and substitution. The goal is to completely scramble the data and key so that every bit of the ciphertext depends on every bit of the data plus every bit of the key (a 56-bit quantity for DES or 168-bit for Triple DES).

Authorized users of encrypted computer data must have the key that was used to encrypt the data in order to decrypt it. The unique key chosen for use in a particular application makes the results of encrypting data using the algorithm unique. Using a different key causes different results. The cryptographic security of the data depends on the security provided for the key used to encrypt and decrypt the data.

The outcome of implementing the DES algorithm is that a cipher text is produced which has one key of a sequence of 2^{56} or 70,000,000,000,000,000 (seventy quadrillion).

Figure 1, below shows the fundamental operation of the DES encryption process, (implementing ECB mode).

Notes

1 NIST – National Institute of Standards & Technology, formerly the National Bureau of Standards (USA)

Encryption Methods for Protecting Data

A 64 Bit block of data is presented to the encryption engine. An initial permutation of the block is made. The block is then divided into two 32 bit segments, LBlock and RBlock. Using a 56 bit derivative of the 64 encryption bit key, a complex non-linear operation (\circledast) is performed on RBlock.

The modified RBlock is then XORed with LBlock and the resultant fed to the next RBlock register. The unmodified RBlock is fed to the next LBlock register.

With another 56 bit derivative of the 64 bit key, the same process is repeated.

The sequence is performed a total of 16 times before the LBlock and RBlock segments are recombined and an inverse of the initial permutation is performed.

The result is the 64-bit ciphertext block.

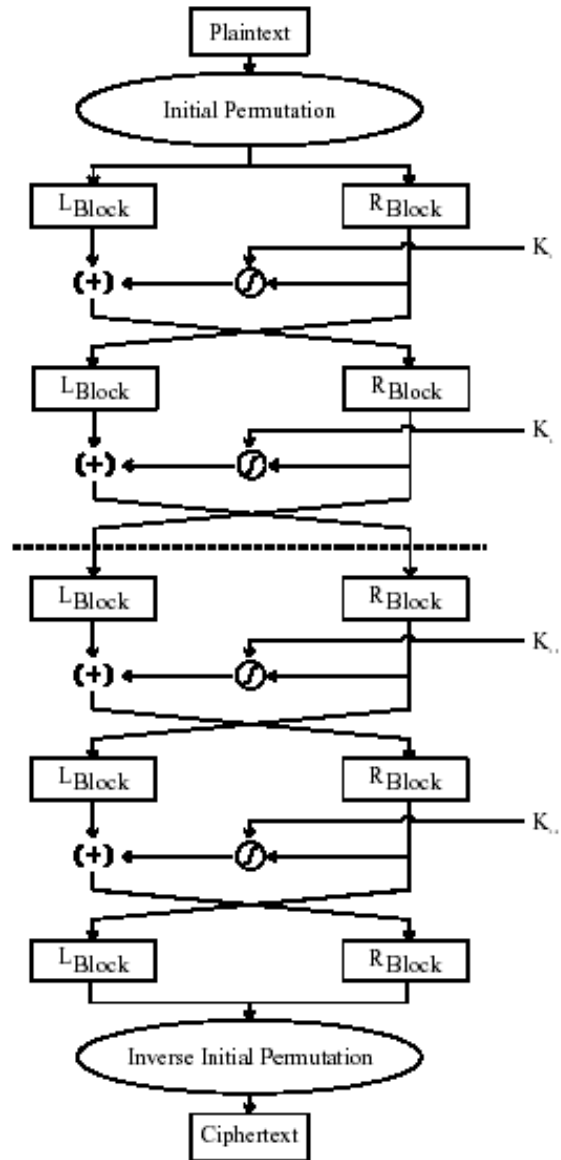


Figure 1. (DES Function)

DES Modes Of Operation

The Federal Information Processing Standard (FIPS) defines four modes of operation for the DES that may be used in a wide variety of applications. The modes specify how data will be encrypted and decrypted.

The modes included in this standard are the **Electronic Codebook (ECB)** mode, the **Cipher Block Chaining (CBC)** mode, the **Cipher Feedback (CFB)** mode, and the **Output Feedback (OFB)** mode.

The following sections offer a simplistic demonstration of the operation of the various encryption modes. A more detailed description of the modes of operation can be found in FIPS Publication 81, DES Modes of Operation.

Electronic Codebook (ECB)

In ECB encryption, a plaintext data block is applied to the model featured in Figure 1. Plaintext is presented to the DES encryption engine for processing. Upon completion a ciphertext block is output. The function continues converting plaintext to ciphertext on an individual block by block basis.

The ECB decryption process is the same as the ECB encryption process except that the decrypt state of the DES device is used rather than the encrypt state.

Cipher Block Chaining Mode (CBC)

In CBC mode, the initial plaintext block is XORed with an Initialisation Vector, IV, which is derived from a double length encryption key. The result of the XOR is processed by the encryption engine and output as the ciphertext related to the input plaintext block.

The XOR output is also used as the IV in the next plaintext block encryption function. As a result,

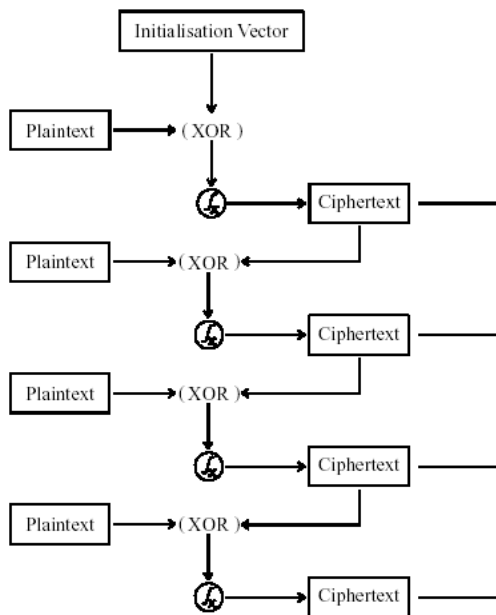


Figure 2. (CBC Encryption)

each ciphertext block output is directly related or chained to the previous encryption function.

Cipher Feedback Mode (CFB)

In CFB mode, the initial plaintext block is XORed with an Initialisation Vector, which is derived from a double length encryption key. The resultant XOR function is output as the first ciphertext block.

The XOR result is also fed into the encryption engine and the output presented as the IV for the next plaintext block.

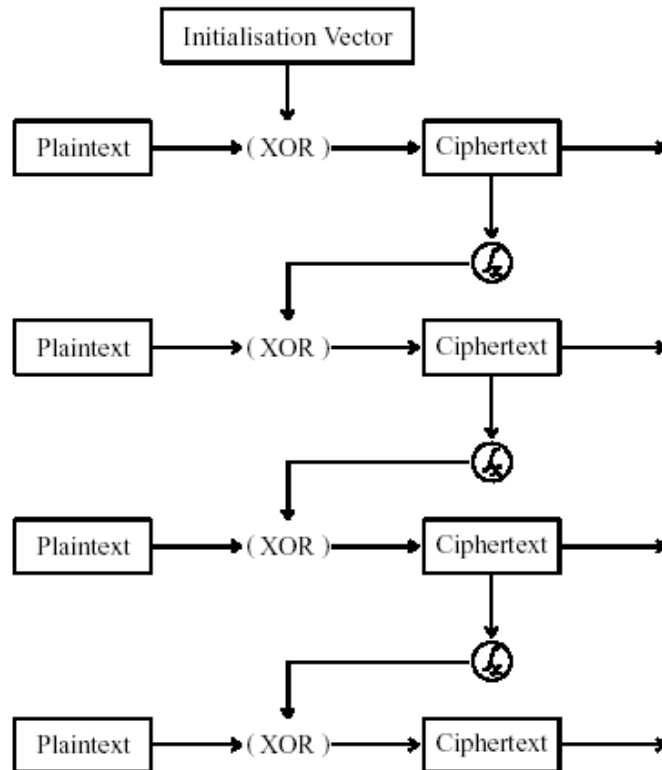


Figure 3. (CFB Encryption)

Output Feedback Mode (OFB)

In OFB mode, an Initialization Vector is extracted from a double length encryption key. The IV is presented to the encryption engine and the resultant encrypted cipher key is XORed with the plaintext block to produce the ciphertext block.

As a second plaintext block is presented the encrypted cipher key used for the previous XOR is cycled through the encryption engine again to produce the next cipher key.

As a result the initialization vector is continually processed as a stream through the encryption engine until no more plaintext blocks are presented.

Figure 4. (OFB Encryption)

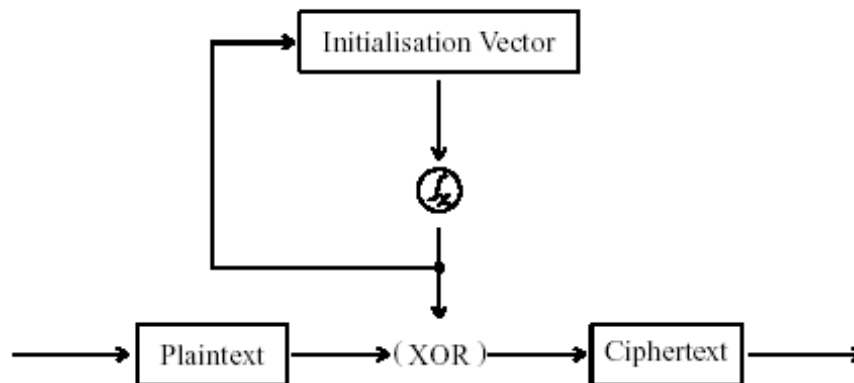


Figure 4. (OFB Encryption)

Triple DES

In recent years, increased microprocessor performance has left a concern that the already high security level offered by the DES encryption algorithm may be threatened by a concerted “*brute force*” attack. As a result the DES3 or Triple DES standard has been ratified to provide a still higher encryption level. DES3 operates by performing three DES1 processes using a 168 bit key. Figure 5 below shows the DES3 process.

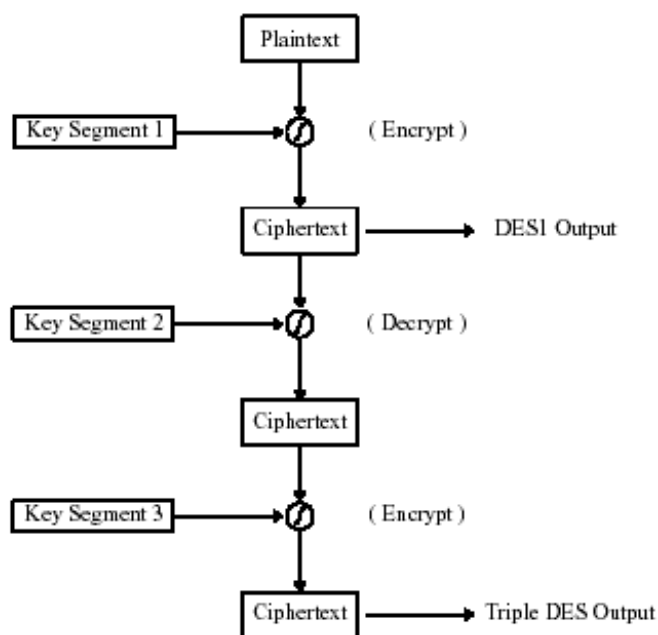


Figure 5. (Triple DES)

The 64 bit plaintext block is entered into the DES1 encryption engine, where the first 56 bits of the encryption key are used to perform the first encryption function. The ciphertext product is returned through the engine and decrypted using the second 56 bits of the key, having the effect of a second phase encryption.

Finally the ciphertext is run through the engine again using the last 56 bits of the key producing a “tripling” of the DES1 function.

As the encryption algorithm requires the full 168 bit key to be input, the possible key permutations are 2^{168} which equates to approximately 3.5^{50} or 350,000,000,000,000,000,000,000,000,000,000,000,000,000,000.

BOSaNOVA, Inc.

Phone: 866-865-5250 **Email:** info@theq3.com **Web:** www.theq3.com

Security Implications Of Key Ciphers

The strength of key based ciphers is the vast quantity of possible key permutations that are available to the user. However this is also the weakness. The algorithm is declared public, therefore only the integrity of the key, or more to the point the user with access to the key, is susceptible.

Data can be retrieved by an unauthorized party in a number of ways; Assuming an unauthorized party has gained access to a tape cartridge which holds sensitive data there are a number of ways that the encryption can be compromised.

1. A *brute force* attack is possible. With enough financial and computer resource, every possible key combination can be generated and tested. But, without knowing exactly what the unencrypted data was in byte or binary terms, how will the perpetrator know when the correct cipher key has been found? Assuming the data was generated by an application with a known file structure, in theory the crack should be easier to achieve. That said the application would have to restore the data before it can be seen to be correct. The crack time has now risen from the time taken to generate and apply the keys to a few cipher blocks, to up to 2^{168} restore operations.

2. A hardware solution could be designed or a software program written if a portion of unencrypted data is known to be on the tape in encrypted form. A full key sequence could be generated and a comparison to the data on tape made. This would again involve multiple tape passes. To avoid incurring the penalty of multiple tape comparisons the data could be copied to disk and the test data compared, however if there was fifty gigabytes of data on the tape and this was copied and compared at disk level, there could be up to 2^{168} fifty gigabyte comparisons

3. If there is access to the cipher key, the data is compromised.

Securing The Key

It is clear that if an encrypted tape becomes available to an unauthorized party, the danger to the data is minimal, unless there is an “inside” source from whom the encryption key used to generate the tape is available.

Security can be introduced on a procedural basis, however this will not protect from a malicious act other than to highlight that person’s actions after the data has entered the public arena, such as in a press disclosure.

If however the data is used in a private manor for business or financial advantage it is unlikely the rightful owner of the data will ever know that security has been compromised.

DES does not have the facility to safeguard against disclosure of the encryption key; therefore it is necessary for any device that introduces encryption to overcome this problem.

A system whereby both a “User Key” and a “Device Key” must be implemented is the most reliable method. In this case a key is input at each session by the user/operator and the encryption unit itself also has an internal key unique to itself. The unit accepts the user key and performs a non linear algorithm with its own key. The resultant product is a composite key that is used to encrypt the data. More to the point data can only be decrypted by a combination of the correct encryption unit combined with the correct user key. Figure 6 illustrates the principle.

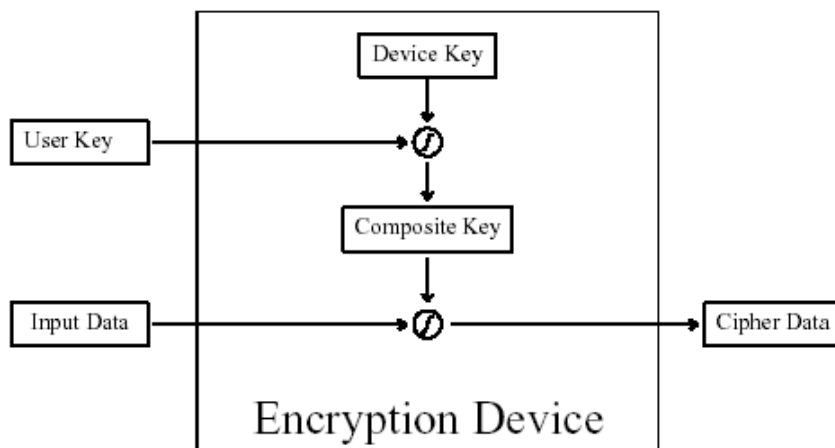


Figure 6. (Split Key Subsystem)

Encryption Device

The Hardware Solution

There are many software encryption solutions on the market today, however as well as not addressing the issue raised above they are inappropriate in a modern storage environment.

In an enterprise backup environment, in almost all cases data being written to tape is routed from a disk server of some description through a number of backup servers to a tape drive or silo. If a software solution is implemented it will introduce a performance overhead either at the disk or storage server level.

It is a fact of the modern computing industry that due to international access requirements to data, the backup window is shrinking. Data is growing at an almost exponential level. The corporate disaster recovery capability is deemed to be of higher importance than data security. As a result, any encryption system that bottlenecks the data coming from the servers is unlikely to be implemented.

Only an external encryption device meets all of the criteria specified for effective data security in a tape storage environment. Furthermore the subsystem must be platform and data management software independent. Using an integrated encryption card is not feasible unless the on board key is either reproducible in the event of a failure of the card. If encryption is used as a means of securing data while tapes are in transit, duplicate subsystems will have to be at the source and destination sites.

A further limitation of an embedded encryption card is in a heterogeneous environment where data could be produced on a system implementing S-Bus technology but may need to be read on a system implementing PCI for example.

Conclusion

When investigating how to implement a program for ensuring the security of data when held on tape it is vital that a combination of appropriate technology is operated under a strict procedural policy.

It must be established how much of the corporate data is sensitive.

Upon deciding what data is sensitive it must be established whether the medium on which the data is stored is portable, or is susceptible to theft from the data center.

BOSaNOVA, Inc.

Phone: 866-865-5250 **Email:** info@theq3.com **Web:** www.theq3.com

An encryption device must be selected which can be moved from platform to platform with the minimum interference to operations and which will not introduce software related problems.

The device must support recognized encryption standards.

The device must have an integral key to prevent compromise of data in the event of the user key becoming public.

The device must have a “standalone” facility and not be dependent upon the data path server for configuration of the user key.

A device range must be selected which is flexible enough to produce unique key or common key units dependent upon the corporate data sharing requirements.

Further Reading

FIPS PUB 46-2

Data Encryption Standard (DES)

(<http://www.itl.nist.gov/fipspubs/fip46-2.htm>)

FIPS PUB 46-3

Data Encryption Standard (DES)

(<http://csrc.nist.gov/fips/fips46-3.pdf>)

FIPS PUB 81 Modes of Operation

(<http://www.itl.nist.gov/fipspubs/fip81.htm>)

NIST Special Publication 800-20 Modes of Operation Validation System for the Triple Data Algorithm

(<http://csrc.nist.gov/nistpubs/800-20.pdf>)

NIST Special Publication 800-17 Modes of Operation Validation

(<http://csrc.nist.gov/nistpubs/800-17.pdf>)