SNIA Technical Tutorial

Storage Network Security

SNIA TECHNICAL TUTORIAL

Storage Networking Security

The SNIA Technical Tutorial booklet series provides introductions to storage technology topics for users of storage networks. The content is prepared by teams of SNIA technical experts and is open to review by the entire SNIA membership. Each booklet corresponds with tutorials delivered by instructors at Storage Networking World and other conferences. To learn more about SNIA Technical Tutorials, email snia-tutorialmanagers-chair@snia.org.

SNIA TECHNICAL TUTORIAL

Storage Networking Security

Roger Cummings
SAN Technologist, VERITAS Software

Hugo Fruehauf
CEO and CTO, Zyfer, Inc.

SNIA
Storage Networking Industry Association

Contents

Preface

Even as recently as the mid-1990s, the words "storage" and "security" would rarely have been used in the same sentence. Storage was a mostly unappreciated component of a computer system and security, at least in computer terms, was something associated with government contracts and semi-anonymous bodies known mostly by acronym, such as CIA, NSA, etc.

However, even then several trends were under way that would change this situation. Storage networks were beginning to be deployed, with a consequent separation of storage from the computer system. The growth of the Internet is also well-known to have led to a wide variety of publicly accessible services that needed to be available 24 hours a day, seven days a week. The result was a vast increase in the amount of information being handled by computer systems, which made the scalability provided by storage networks critical. The 100% availability requirements imposed by the services also made organizations aware as never before of the importance of that information in terms of their businesses' continued ability to operate and prosper. As a result, storage has gained importance in the eyes of many people as the container of, and protection system for, one of a business's key assets, namely its information. And assurance techniques, such as backup and restore, replication, and "frozen images" became key in ensuring that information was never lost, and that a permanent audit trail could be maintained.

As powerful as these assurance techniques are, they have solved only part of the problem. If data is corrupted "in flight" on its way to storage, or if data on the storage device is corrupted by an access from another computer system, the techniques will likely not be able to recover the original information. Therefore, the Security Technical Working Group of the Storage Networking Industry Association (SNIA) was chartered in 2000 to look at methods of ensuring security of information both in flight through a storage network and at rest on a storage device. That charter also included an education component, because the history of the storage and storage networking industries does not suggest a great familiarity with security tools and techniques. As a result a security tutorial specifically targeted at people in those industries was created. At the time of this writing, that tutorial has been extended from an initial 45 minutes to two hours in length, and is being presented at the twice-yearly StorageNetworkingWorld conferences and at other venues on request. Much of this booklet, although it has been enhanced in several areas, is based on the tutorial material.

The approach that has been used to good effect in the tutorial, and repeated here, is to begin with a brief history of cryptography, because encryption is the first thing that people tend to think of when security is mentioned. The history begins with the "Caesar cipher," so named because Julius Caesar mentioned its use

during the Gallic Wars, but which is better known to many of us as the "secret writing" letter substitution schemes that comics taught us to use to protect the confidentiality of messages between members of some club or other from the prying eyes of nonmembers and parents. The development of ciphers from that point through the mechanization of the Enigma machine to the present day is then traced, and the characteristics of the most popular current schemes compared. The other three major facets of security techniques—authentication, nonrepudiation, and integrity assurance—are also defined. For further information, a reading list—located in Appendix B—is provided that contains resources the authors found useful.

One of the major difficulties facing the SNIA Security Technical Working Group is the fact that the creation of storage networks did not result in a redesign of the entire storage "stack" (i.e., the software that is traversed in carrying a request from an application to a storage device). Quite to the contrary, to ease migration much of the existing stack that was used for direct-attached devices was carried forward to support storage networks, and the remainder was only enhanced to the absolute minimum needed for successful operation. For example, the SCSI command set that was used to control direct-access devices is still employed as the method by which devices attached to the storage network are controlled. Therefore, given that security was little considered in the direct-access case, few enhancements were made in the early phases of transition to storage networks.

Therefore, few of the features of storage networks that are now used for security purposes were conceived entirely on that basis. For example, Fibre Channel zones were conceived as a method of limiting the connectivity viewed by a single system to something close to that of the previous direct-attached schemes. Logical unit mapping and masking abilities were created for a similar purpose. However, all three techniques can be very effectively used to limit the exposure of the entire network to security problems, if not to prevent their occurrence in the first place.

Specific security features are now also being included in the transports used in storage networks, leveraging to the maximum extent possible the experiences with other types of networks and the security schemes developed for those situations. As an example, a new optional header is being defined for Fibre Channel that contains an analog of the encapsulating security payload header defined by IP Security for use in protecting information transferred by TCP/IP infrastructures. Such features are expected in the next few years to enable the same type of security protection for storage networks as is now available for other types of networks. Note that in many cases these techniques have not been widely employed in the other situations, but they are proven and their use to protect the vital information in flight in a storage network is expected to be compelling.

Even though progress is being made to protect the "in-flight" data, this addresses only half of the Security Technical Working Group charter; methods of protecting data "at rest" on a storage device must also be created. Present research indicates that much less attention has been paid so far to these requirements, and thus more innovation is likely to be needed. Long-term key man-

agement is also an essential part of these methods, because it is clearly not useful to secure information stored on devices for long periods if, when the information is finally needed, the key required to confirm its integrity and perform the required decryption cannot be found. Thus, new methods of key escrow will have to be created, as well as methods to protect against loss of information. Schemes to allow storage devices to become self-securing are now starting to be considered in research circles.

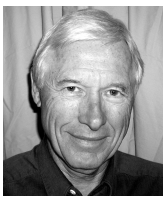
This booklet is therefore being created at a highly significant time in the development of storage network security. Although developments are still in the very early stages of their life cycles, definition of the basic tools that will form the foundation of the required higher level of security is nearing completion, and interesting approaches to the problem of securing information while at rest on storage devices are starting to be formulated. But it is still vital that the currently available tools be applied to the best extent possible in advance of the above developments. The risk assessment process developed by the SNIA Security Group is one effort to get managers of storage networks to analyze their security problems and requirements now rather than later.

It is essential that both approaches—optimizing the use of existing tools and developing new ones to support a higher level of security—continue and develop. The sustained growth of e-commerce, and the use of the Internet to access sensitive information, demand it. The ever-increasing amount of information created in binary form is making it imperative that at some point in the near future such information be accepted as providing a record of legal quality. This will not happen unless the security facets of authentication, confidentiality, integrity, and nonrepudiation become attributes of the handling of such information throughout its life cycle. It is most important that storage networks are seen to be a strong point in the life cycle, not a weak link.

About the Authors

Roger Cummings

Roger Cummings is a SAN Technologist at VERITAS Software, reporting to the CTO. He is based in Heathrow, Florida. He is a Co-Chair of the SNIA Security Technical Working Group and a member of the SNIA Technical Council. Roger has more than 30 years of development experience with Logica (UK), Control Data (Canada), and StorageTek and DPT (USA). He has been involved in storage standards work since the early 1980s, most notably as an officer of the iNCITS T11 (Fibre Channel) committee from 1990–1998, and is currently also active in the iNCITS T10 committee (SCSI), various IETF Working Groups, and the Infiniband Trade Association.



Hugo Fruehauf

Hugo is CEO and CTO of Zyfer, Inc., located in Anaheim California, a wholly owned subsidiary of Odetics, Inc. He has more than 35 years of experience in commercial and military communication networks, satellite systems design, and tactical weapons systems. He was Group VP at Alliant Techsystems (GPS-aided tactical weapons systems), President of Efratom (time/frequency synchronization and link security), and Chief Engineer at Rockwell for the initial GPS Block-I satellite systems design. He is the architect of Zyfer's new network security processor development, called "StealthKey," for cryptographic IP data security.

Acknowledgments

The authors would like to acknowledge the assistance of members of the SNIA Security Technical Working Group, and Vijay Ahuja in particular, for their review of many revisions of the tutorial on which this booklet is based. Our thanks to the SNIA Education Committee in general, and Paul Massiglia and Paula Skoe in particular, for their efforts in establishing the SNIA tutorial series, which now covers over a dozen topics in addition to storage networking security. The staff of the *ComputerWorld,* a division of IDG, and Nanette Jurgelewicz in particular, deserve our thanks for providing the opportunity for us to present the Storage Networking Security tutorial at StorageNetworkingWorld. And finally, thanks are due to the many people who have not only attended the seminars, but stayed behind afterward to give us excellent feedback and suggestions for improvement. We hope you will find this booklet all the more useful for their feedback.

CHAPTER

Introduction

Storage network security is a relatively new subject, but one that is rapidly gaining in importance in the minds of both users and product developers. This increase is born of a general realization of the increasing importance and value of the information held in on-line systems, and of the separation of processing and storage functions enabled by the development of storage area networks (SANs).

Information security is, of course, not a new subject. A brief introduction to the history of cryptography and other techniques for securing communication will be presented in this booklet. But while storage network security seeks to learn from the application of similar techniques to communications security in general, and to network security in particular, it has some unique requirements that will necessitate the development of new and specialized techniques. Though the development of such techniques is in its infancy, this booklet will summarize the current state of the art.

This booklet was conceived as an educational resource for the members of the Storage Networking Industry Association (SNIA), and other designers, product strategists, and developers within the storage and SAN industries. SNIA's mission is "to ensure that storage networks become efficient, complete, and trusted solutions across the IT community." Clearly, that mission cannot be achieved unless the SNIA membership is able to employ the best technologies, techniques, and practices to create storage networking products. Therefore, an important part of SNIA's mission is the education of its members, and this booklet forms one part of many initiatives in this area. More detail about SNIA's mission and current activities can be found on its Web site at www.snia.org.

The intended audience of this booklet includes people who are storage literate, but not necessarily security literate. Thus, although this booklet provides an introduction to a number of security techniques, it is not a storage network primer.

It is undoubtedly a cliché, but security is a process rather than a product. Security is only ensured by the constant analysis and review of all aspects of the operation of a storage network. This booklet describes a process for creating a security assessment that can be used by designers in planning products, by user organizations in planning specific configurations, and by system administrators as their production configurations expand and change over time.

Why do all these different groups of people have to be involved in security? Because there is no "one size fits all" security solution for storage networks: The configurations are simply too diverse, and the way that data is used in such networks is not yet well enough understood. As an example, in some storage network configurations all data is freely shared between all of the connected servers (e.g.,, in a single application system running on a cluster for redundancy), while in another configuration each server runs a different operating system and as a result no data is shared across the entire network. These two situations clearly have much different security requirements. Therefore, the level of security appropriate to a specific storage network has to be determined at the time that the configuration is created. The aim of this booklet therefore is twofold: to illustrate to the persons creating a storage network configuration how they might determine and enforce an appropriate level of security, and to demonstrate to the designers of products that might be used in such configurations why specific security features might be required.

We begin with a definition of security terminology, including detailed definitions of the four key terms used to describe many security techniques. Many of these terms are extracted from the storage networking dictionary that is also found at the SNIA Web site. This is followed by a brief history of codes—or, more properly, cryptography—from its origin in ancient Egypt through modern techniques, along with various approaches that have been used to break through such protections and read the information. An introduction to the standard tools and approaches used in security today—schemes such as 3DES and AES—then follows, along with some estimates of the security provided by each scheme.

A method of assessing the risks to security contained in storage area networks is described, and the method is applied to a generic SAN and all of the potential attack points are inventoried and assessed. This is followed by a description of the tools and practices that are currently available to provide security in a SAN. The early work on future security technologies and techniques is then described, and we conclude with a summary.

Two appendices are also included and contain a list of applicable industry standards and a recommended bibliography.

CHAPTER

Definition of Terminology

This chapter contains two sections: *full definitions* are given for four important concepts that are used extensively in describing security methods and technologies, and *brief definitions* are given for terms used elsewhere in this booklet. For more details, and for a more comprehensive dictionary of terms used throughout storage networking, please see the Web site at www.snia.org.

Key Concepts

The four important terms are authentication, confidentiality, integrity, and nonrepudiation.

Authentication is a security measure designed to establish the validity of a transmission, message, or originator. In its most straightforward form, authentication allows a receiver to have confidence that information received originated from a specific known source. Often protocols require that senders and receivers be authenticated to one another in order to enable bidirectional information transfer. Authentication is a necessary condition of authorization, which is the definition of actions a specific party is permitted to execute, but for the purposes of this booklet, authorization will be regarded as a separate concept.

Confidentiality is a security measure that protects against the disclosure of information to parties other than the intended recipient(s). Confidentiality is often ensured by means of encoding the information using a defined algorithm and some secret information known only to the originator of the information and the intended recipient(s) (a process known as cryptography). However, this is by no means the only way of ensuring confidentiality. An example of an alternative means is "steganography," in which confidentiality is ensured by disguising the type of information—for example, by sending a message as part of the bitmap of an image.

Integrity is a security measure intended to allow the receiver to determine that the information received has not been altered in transit or by other than the

originator of the information. Integrity schemes often use some of the same underlying technologies as confidentiality schemes, but they usually involve adding information to a communication to form the basis of an algorithmic check, rather than encoding all of the communication.

Nonrepudiation is the security measure intended to prevent the subsequent denial that an action happened or that a communication took place. In communication terms this often involves the interchange of authentication information combined with some form of provable time stamp.

Terminology Used in this Booklet

Terms used elsewhere in this booklet are:

Cipher Any cryptographic system in which arbitrary symbols or groups of symbols represent units of plain text or in which units of plain text are rearranged, or both

Ciphertext Information that has been encoded by use of a cipher to ensure confidentiality

Cryptography Literally the study of secret writing, the science of transforming information to and from a form that has confidentiality

Decoding The process of transforming ciphertext back to the original plaintext by use of a cipher and a key

Denial of Service The result of any action or series of actions that prevents any part of an information system from functioning

Encoding The process of transforming plaintext to ciphertext by use of a cipher and a key

Hash A computationally efficient function mapping binary strings of arbitrary length to binary strings of some fixed length that uses a "one-way" function (i.e., the original strings cannot be derived from the fixed-length ones)

Key A piece of information known only to the sender of some information and the intended receivers. A key is usually a sequence of random or pseudorandom bits used to direct cryptographic operations and/or for producing other keys. The same plaintext encrypted with different keys yields different ciphertexts, each of which requires a different key for decryption

Plaintext Information that has not been encoded, or has been decoded from ciphertext

CHAPTER

A Brief History of Cryptography

The idea of protecting a message, or information in general, so that it can only be understood by those "in the know" has probably been around for as long as human history. This chapter traces the development of that idea from ancient Egypt to the present day, not only as a historical pageant, but as a demonstration of the common principles upon which many of these schemes are built, and the common weaknesses that have been used to attack them.

Ancient History

Ancient Egyptians placing hieroglyphs on public monuments are known to have substituted some common symbols with special ones, and ancient Hebrews replaced some common words in their scriptures using a defined substitution scheme. The term we use for such processes today is **cryptography,** derived from the Greek *kruptos* (hidden) and *graphia* (writing), which emphasizes that anything that can be written can be so protected, and not just numbers or letters.

For most of us, our first introduction to cryptography was in the form of a game suggested by a children's magazine. A secret code book was provided to allow children to exchange messages without parents being able to read them. The code book contained something of the following form, in which two alphabets are written on successive lines offset by a specific number of characters.

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
```

Each letter in the message was then located in the first alphabet, and replaced by a corresponding letter in the second alphabet, so that:

```
T H E Y   W A N T   Y O U   T O   C L E A N   Y O U R   R O O M
```

becomes

```
W K H B   Z D Q W   B R X   W R   F O H D Q   B R X U   U R R P
```

In fact, the particular substitution used above—the replacement of each letter by the third letter on—is known as the Caesar Cipher because its use during the Gallic Wars was reported by Julius Caesar. Because there are only 26 letters in the alphabet, the generalization of this scheme supports only 25 possible keys (because the 26th value performs no encoding), and therefore it takes a maximum of that many attempts to decipher an unknown message. In modern terms this is known as a monoalphabetic substitution cipher, because only one alphabet is used at a time.

However, a fairly simple extension of such a scheme can offer much better security. Rather than define the translation alphabet by a simple shift, a "jumble" of letters could be used with no pattern. Now the key to the cipher is 26 letters long, and a total of 26 factorial (approximately 4×10^{26}) keys are possible. But the downside of this is that the key is much more difficult to remember, and it is much easier to make a mistake when transcribing such a key. Thus, an easier way to specify the key was required, and a scheme using keywords was created in which the translation alphabet was defined by writing a key word without duplicate letters and then the rest of the alphabet following the last letter of the key word. For example, given the key of "secret writing," the cipher would become:

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
S E C R T W I N G H J K L M N O P Q U V X Y Z B D F
```

and the message

```
T I M E   F O R   B E D
```

becomes

```
V G L T   W N Q   E T R
```

This is, of course, not quite as secure as the real jumble, but is much easier to handle. Note that this remains a monoalphabetic substitution cipher, because all of the letters in the message are translated using the single alphabet.

For many centuries, a cipher such as the one above was used along with key words to aid in key generation, and it was thought to be completely secure. However, toward the end of the first millennium A.D., such a cipher was apparently "broken" (i.e., deciphered without knowing the key) by Arab scientists using a method based on analyzing the frequency of characters in the ciphertext. This is possible because of the relative frequency of specific letters. A very clear description of such a method can be found in the Sherlock Holmes story, *The Riddle of the Dancing Men.* Incidentally, the story also shows that codes do not have to involve letters, because the dancing men in question are stick figures in different positions, with each position representing a letter.

Holmes' exposition is that in English "E" is by far the most common letter. By using this fact and knowledge of the name of one of the people involved and some of the situation, he is able eventually, after obtaining three short segments and two full messages, to completely discover the cipher. Holmes is then able to create a message himself that one of the protagonists accepts without question as coming from someone else, and is thereby lured to a trap that leads to his capture.

A key, if unstated, ingredient in the above situation is that Holmes knows when he has succeeded in breaking the cipher because the message makes sense to him. This is only possible because he knows the language being used, and because only a small proportion of the possible letter combinations make words that exist in that language. The same approach could not be used with a modern message that consisted of something such as an image bitmap.

Later History

Despite the fact that there were known methods of breaking monoalphabetic substitution ciphers, they continued in use for many centuries because of their simplicity. For example, the imprisoned Mary Queen of Scots communicated with her supporters using messages so encrypted, and the capture and reading of several of those messages by the government of Queen Elizabeth I ultimately led to Mary's execution for treason.

The next major event in cryptography was the invention of "polyalphabetic substitution ciphers" by several mathematicians in the 16th century. These ciphers are the equivalent of using many monoalphabetic substitution alphabets in turn, according to a fixed cycle or a known key. Such a scheme allows each letter in a message to be replaced by many others. Although polyalphabetic substitution ciphers are significantly more secure than their predecessors, they are also significantly more complex, and require the use of mechanical aids such as concentric wheels or cylinders with alphabets inscribed to be practical.

An example of such a cipher is shown below. In this case the key word (coder in the example) is repeated as many times as necessary to match the length of the message, and then each letter in the key word determines the Caesar substitution alphabet to be used to encode that particular letter. The codebook is therefore:

```
THE  GERMANS  ARE  COMING
COD  ERCODER  COD  ERCODE
```

In this case, with the keyword "coder," the following five translation alphabets are used:

```
C=CDEFGHIJKLMNOPQRSTUVWXYZAB
O=OPQRSTUVWXYZABCDEFGHIJKLMN
D=DEFGHIJKLMNOPQRSTUVWXYZABC
E=EFGHIJKLMNOPQRSTUVWXYZABCD
R=RSTUVWXYZABCDEFGHIJKLMNOPQ
  ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

and the ciphertext is:

```
VVH  KVTADRJ  CFH  GFOWQE
```

For many years, the polyalphabetic substitution was known as the "chiffre indèchiffrable," the indecipherable cipher. But when its use increased markedly in the 19th century as a result of the growth of the "e-commerce" of the day, namely the use of the electric telegraph, so did increased efforts to break the cipher. A break was apparently first achieved by the English inventor Charles Babbage in the 1850s, but Britain was involved in the Crimean War at the time and the discovery was kept secret. However, less than 10 years later Friedrich Kasiski, a retired Prussian officer, made a similar discovery and published his findings.

Breaking a polyalphabetic substitution cipher involves separating it into its constituent monoalphabetic ciphers. Because typical key words are not very long, and suffer from the same character frequency traits as the language in general, many less than the maximum possible 26 translation alphabets are used in most messages. This repetition made the separation possible. Once again, though, these ciphers continued in common use even after the breaks were known. The famous Zimmermann telegram, the break of which brought the U.S. into World War I, was encoded by a polyalphabetic substitution cipher.

An important point needs to be made here. The breaks of polyalphabetic substitution ciphers were made possible by the characteristics of the keys used, and not because of the characteristics of the cipher itself. It is theoretically possible to make a polyalphabetic substitution cipher that is completely secure, by using a new key for each message where the key is composed of a truly random arrangement of letters and is the same length as the message itself. Such a scheme is known as a "one-time pad," but the problems involved in the secure distribution of such keys make their use infeasible in all but very specialized situations, such as the Cold War–era "hot line."

Early 20th Century

The next major advance in cryptography was more about mechanization than a new form of cipher. It was the realization that an electrical machine could be created to perform the encoding, rather than relying on error-prone manual transcription using a code book or one of the mechanical aids described above. In such a machine, the wiring would define the substitution pattern and thus a switch labeled "A" would be wired to a lamp labeled "Q," for example. Various schemes were developed to allow the wiring to be changed easily.

With this mechanization came the ability to far more easily use truly "jumbled" alphabets rather than ones defined by keywords. If the recipient has a machine with the correct wiring, he can still decode the message. The key to the cipher then becomes the wiring diagram of the machine.

Once it becomes possible to perform the substitutions by means of automata, it becomes practical to perform multiple substitutions one after the other (see Figure 1). By itself, however, this multiplicity does not add any greater security, but the simple addition of a letter shift reminiscent of the Caesar cipher (as is shown in Figure 2) does add some strength.

The famous German Enigma machine mounted the wiring for each substitution on a separate wheel. This allowed the shifts to be easily performed by rotat-

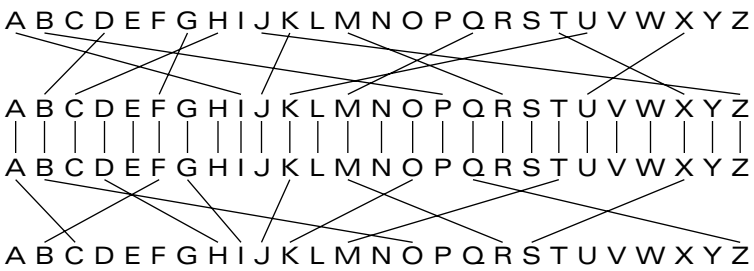


Figure 1

ing one wheel in relation to another. The version of the Enigma used by the German military employed three separate wheels and a fixed reflector plate that swapped letters in pairs and then fed back through the three wheels again, for a total of seven substitutions. The reflector was apparently added to allow the same machine configuration to perform both encryption and decryption, but, as in earlier schemes, this "reuse" of the same wiring added some redundancy that was exploited in breaking the key.

Five different wheel designs were used in the scheme. The three wheels in use, the placement in the machine, and the rotation offset with respect to one another were determined by a daily schedule distributed in paper form on a monthly basis. One principle of the scheme was that the absolute wheel start position should be different for each message (a "session key" in modern parlance), and this was sent encrypted according to the daily setup in a message preamble. To guard against reception problems corrupting this vital information, the entire sequence was also repeated, and again this redundancy was vital in cracking the scheme.

The exact details of how the Enigma code was broken are beyond the scope of this booklet. The process has been documented in several books and at least one dramatic presentation, all of which are referenced in the appendices. Suffice it to say that even when an Enigma machine was available to the decoders and they understood how it worked, a significant advance in the state of the art of the mechanization—the creation of something very much akin to a modern computer—was necessary to be able to process the very large number of possible

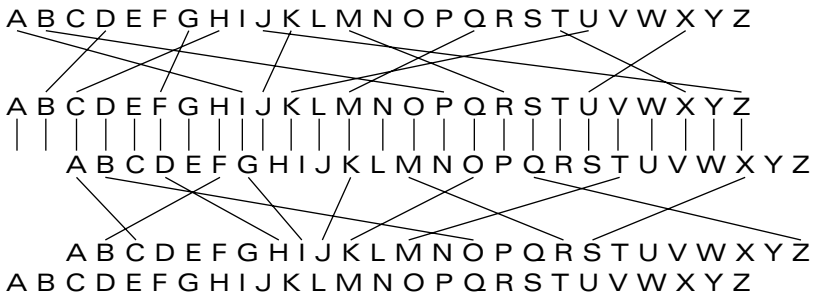


Figure 2

combinations to determine the configuration of the machine on a particular day. Again, the fact that this approach was possible depended on specific weaknesses in the "key" due to the dual use of the wheels, and the redundant information in the message preamble, rather than any inherent limitation in the multiple substitution scheme.

Modern Times

Although modern cryptographic schemes are designed to protect messages composed of binary data rather than text, they are built on the same principles as the historic schemes reviewed above. The basic operations of substitution and shifting (normally called a transposition) are still used. Two types of cipher are commonly defined: a stream type that operates on a single byte (character) at a time, as for the historical schemes above, or a block type that operates on a fixed number of bytes and requires the message to be padded to an integral number of blocks. Block ciphers have the advantage of permitting more parallelism in implementation, and thus they often have higher performance, but they do require all of the bits in the block to be available before the operation can commence.

The most significant event in cryptography in recent times has not invented new operations to be applied to information, but has revolutionized the management of keys. Its underpinning is a challenge to one of the most common assumptions in cryptography, namely that the encryption and decryption processes are mirror images of one another and use the same (symmetric) key. Public key encryption, however, defines a pair of keys, one of which is needed to decrypt what has been encrypted by the other. The two keys are mathematically related, but cannot feasibly be derived from one another. In addition, the encryption process is resistant to attacks that can determine either of the keys.

Public key encryption, therefore, presents the seeming paradox that a message can be encrypted by a known process using a publicly known key, and still only be able to be decrypted by someone holding a different and private key.

The first practical public key encryption scheme was invented in the early 1970s by three mathematicians at MIT: Ronald Rivest, Adi Shamar, and Leonard Adleman. Their solution is known after their initials as RSA cryptography. RSA is based on the multiplication of large prime numbers, which are easy to calculate but very difficult to derive (to determine the original factors given the result). RSA is also used in Pretty Good Privacy, often known by its initials, PGP.

Public key encryption, sometimes known as asymmetric cryptography, has vastly advanced the state of the art of cryptography, and made everything from secure e-commerce to secure e-mail systems practical. But public key encryption is extremely computationally intensive. In many storage-related applications, therefore, its main use will be to distribute to the (symmetric) keys used for both encryption and decryption in modern versions of the historic schemes described above. Both symmetric and asymmetric cryptography will be described in detail in the following chapter.

CHAPTER

Standard Security Tools and Approaches

This chapter will begin by describing in detail the operation of the three types of cryptography scheme in wide use today. It will then outline the principles of the algorithms that are pertinent to storage usage, and describe in detail the operation of three specific algorithms, namely the Data Encryption Standard (DES) and its later version called triple DES (3DES), the Secure Hash Algorithm (SHA-1), and the latest algorithm to undergo standardization, called the Advanced Encryption Standard (AES). The chapter will conclude with some information about how ciphers are cracked, and give some comparative information on the resistance of the various Data Encryption Standard variants.

Cryptographic Schemes

There are three types of cryptographic scheme in common use in high-speed information transfer applications today. These are symmetric cryptography, asymmetric cryptography, and hash function cryptography. They are described in detail in the following sections.

Symmetric Cryptography

Symmetric cryptography uses the same key for both encryption and decryption, as is shown in Figure 3. All of the historic ciphers in the previous chapter were examples of symmetric cryptography. A requirement of symmetric cryptography is that the key must be kept secret. It must also be communicated between the participants by secure means before any exchange of information can take place. The method of communication does not have to be electronic; the exchange of a physical device containing the key or a paper transcription of the key can be equally acceptable, as long as security is maintained.

The advantages of symmetric cryptography are that it is an efficient and fast process, with low latency, and it provides a high level of confidentiality, assuming that the exchange of the keys themselves is secure. The disadvantages of symmetric cryptography are all related to the logistics of the key exchange: Requiring a

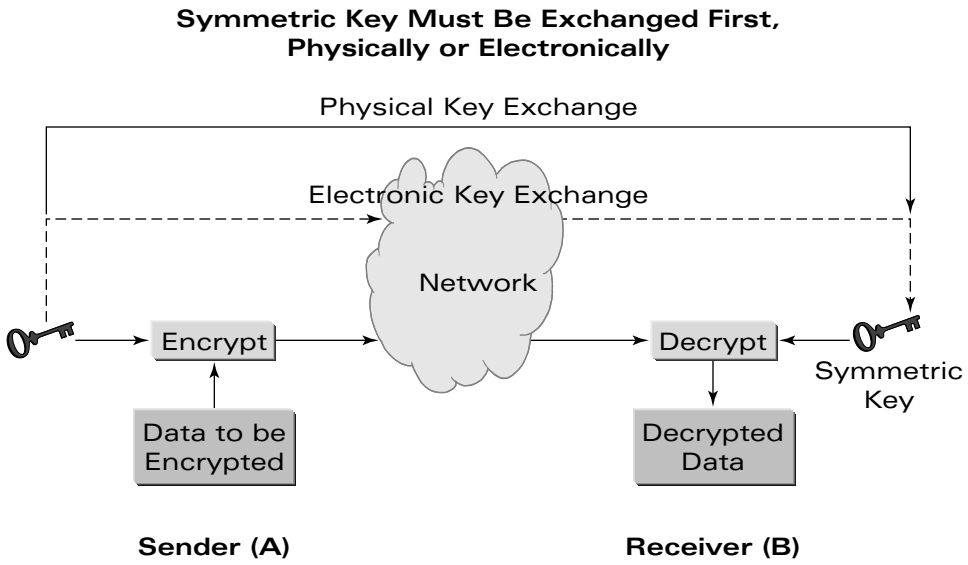


Figure 3

secure exchange between the parties to transfer the key *before* cryptographic communication is possible is very definitely a "chicken and egg" problem.

Asymmetric Cryptography

Asymmetric cryptography is based on theories of public key encryption that have been under continuous development since the 1970s. In asymmetric cryptography, a pair of keys are used; what one of these keys encrypts, the other can decrypt, and vice versa. These keys are separate entities, not two parts of a longer key. The keys are mathematically related, but it is not feasible to derive one of the pair from the other.

Asymmetric cryptography is shown in simplified form in Figure 4. In this scheme the sender's first act is to retrieve the public key for the intended recipient from a public database or some other repository. Note that there is no requirement for this retrieval process to be secure. The sender then encrypts the information with that public key, and sends it to the receiver, who is able to decrypt it with the compatible private key that is present at the receiver and has never been placed in a public database.

The major advantage of asymmetric cryptography is that it obviates the need for the secure key exchange required in symmetric cryptography before any communication can take place. Because of the relationship between the keys, one can be made public (or communicated to by an insecure method), yet the information protected by that key can only be determined by a party knowing its pair key, which has remained private. Asymmetric cryptography provides good facilities for nonrepudiation. That a specific piece of information can be successfully

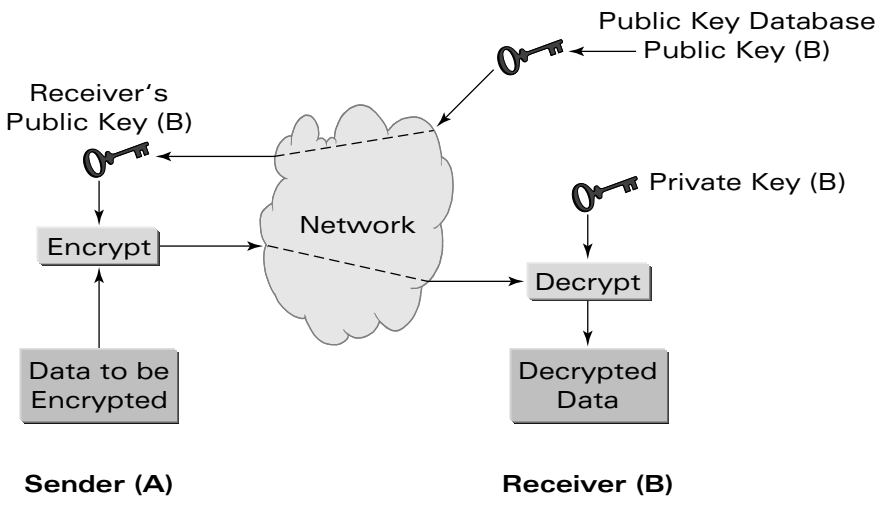


Figure 4

decrypted with a public key is sufficient proof that it must have been encrypted with the appropriate private key. The private key can therefore be said to have digitally "signed" the information. Such a signature can be attached to the transmission of a document such that the source cannot deny creating and transmitting it. Such a signature can also be attached to the receipt acknowledgement of a document, such that the receiver cannot deny receiving it.

The major disadvantage of asymmetric cryptography is that it is extremely computationally intensive. And although security is not an absolute requirement for the public key retrieval process, a lack of security does provide some avenues for an attacker seeking to impersonate the intended receiver.

With asymmetric cryptography, as with many things, the devil is in the details. Figure 5 shows in detail the nine-step process that is required for an entity to obtain an acceptable certificate containing its public key, which can be made available in a public database and used as described above.

The process begins with a Registration Authority delivering a certificate application to the applicant (step 1), who then completes the application (step 2) and generates the appropriate key pair using a browser or other local software program (step 3). One of the keys is then returned to the Registration Authority as part of the completed application (step 4). The Registration Authority reviews the application (step 5), and if it is acceptable creates a certificate request (step 6) and sends it to a Certification Authority (step 7). Note that in some cases the Registration Authority (RA) and Certification Authority (CA) functions may be performed by the same organization, but this is not necessary and the separation is shown here for illustration. The Certification Authority then generates a certificate incorporating the public key (step 8) along with its identification information, and returns this certificate to the applicant (step 9), who can then post the certificate in a public database.

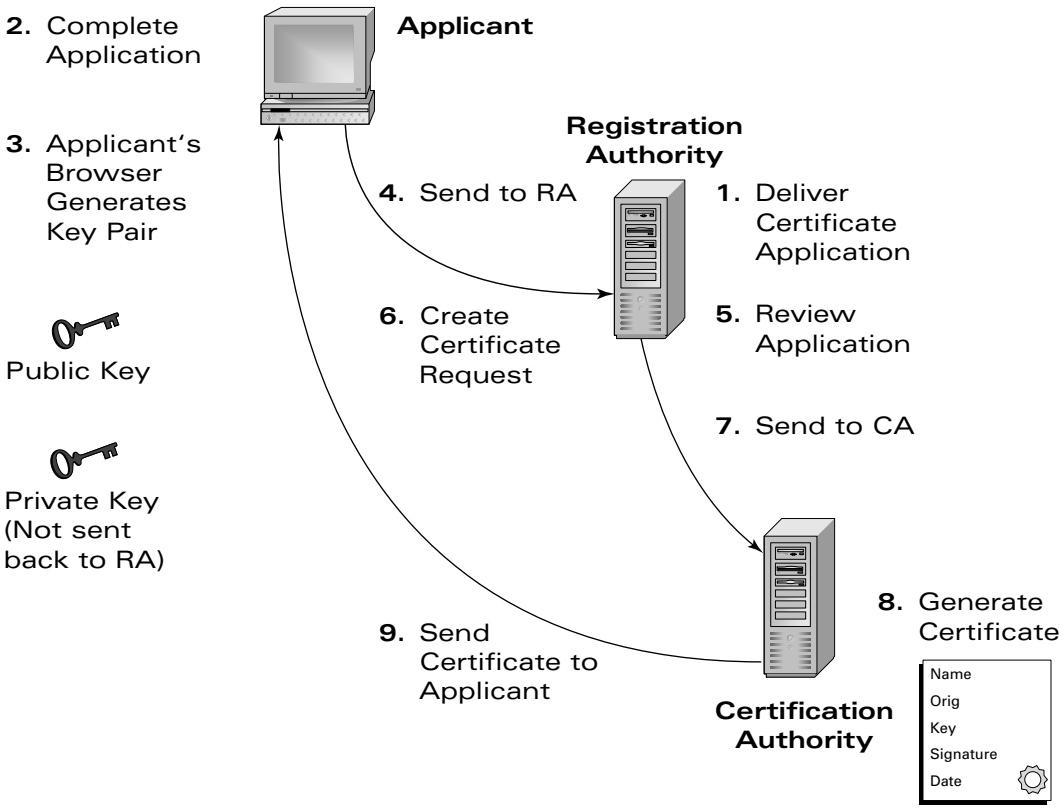


Figure 5

Although the Registration Authority and Certification Authority may be separate functions in the registration process, both functions must be authenticated by, and derive authority from, other entities in a hierarchy that leads to an eventual root authority. This hierarchy is shown in Figure 6. Root authorities are normally commercial organizations that exist either purely for that purpose, or because trust relationships are an integral part of their business (e.g., banks and credit card companies). The process of authenticating entities using certificates requires that both certificates be able to be traced back to a common root. For this reason, certificates from several root authorities are included with most common Web browsers. For example, the Internet Explorer Revision 5.5 browser includes over 100 certificates from trusted root certification authorities, and 20 more from intermediate certification authorities.

One of the major uses of asymmetric cryptography is to provide a secure method for exchanging the common key required for use by both the sender and receiver in symmetric cryptography, and this is shown in Figure 7. This usage is a good marriage of the strengths of both schemes because the computational intensity of the asymmetric cryptography has only to be performed on the limited size of the key and not on all the information exchanged. However, a further refinement is possible. When a sender and receiver share each other's public keys and have exchanged some additional numerical information, then each is capa-

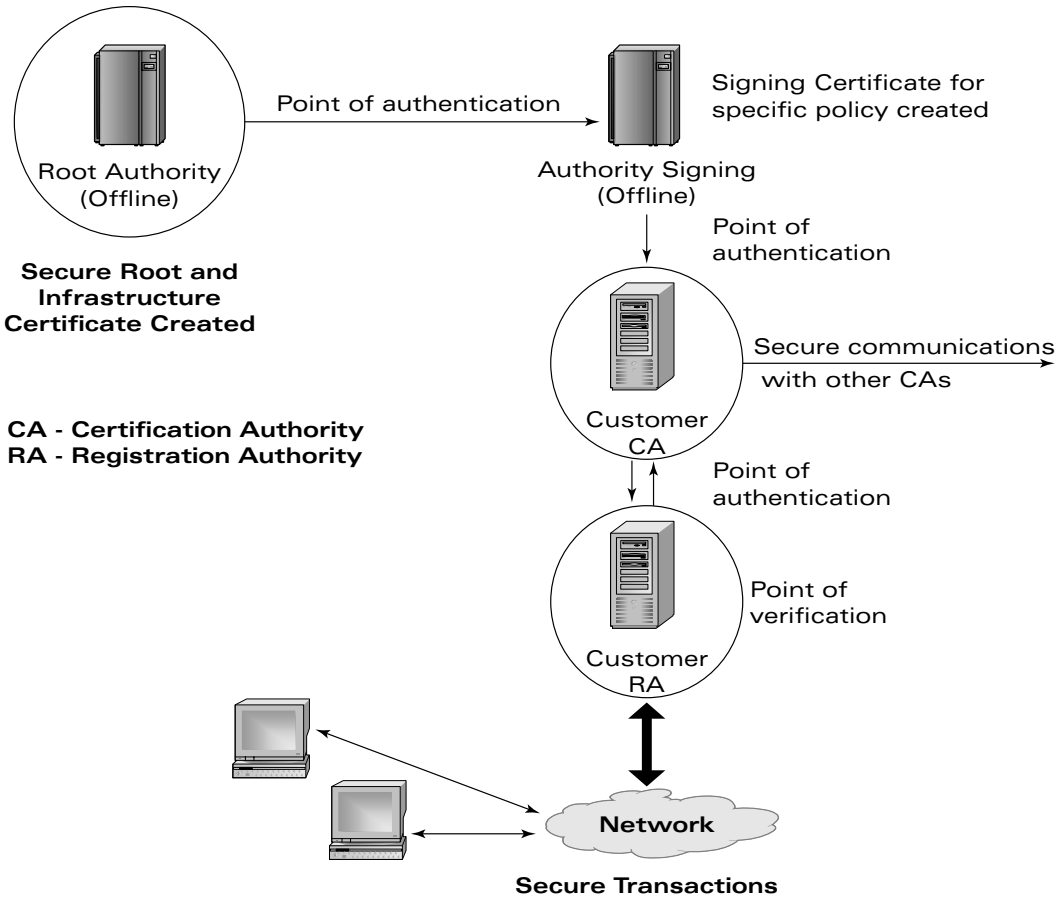


Figure 6

ble of using the combination of its private key and the received public key to create an identical key to the other party, which can then be used for symmetric cryptography. This process is shown in Figure 8.

Hash Function Cryptography

Hash function cryptography differs from the previous two types in that its function is to provide a guarantee of integrity instead of confidentiality. The cipher can be applied to either the plaintext or ciphertext to produce a fixed-length hash value, representative of the bits and sequence of bits of the data. The value can then be appended to the data and transmitted by the sender. The receiver applies the same cipher to the received data, and compares the hash that is generated with the one received. If the two match, then the receiver has a guarantee that the data has not been altered in transit. This guarantee is based on the fact that, not like the previous

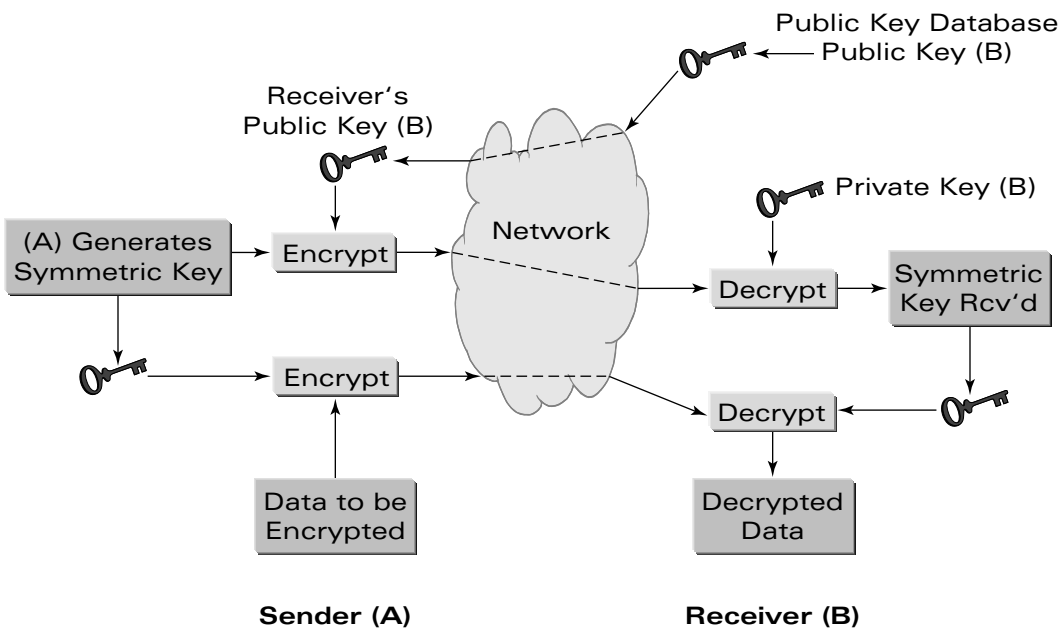


Figure 7

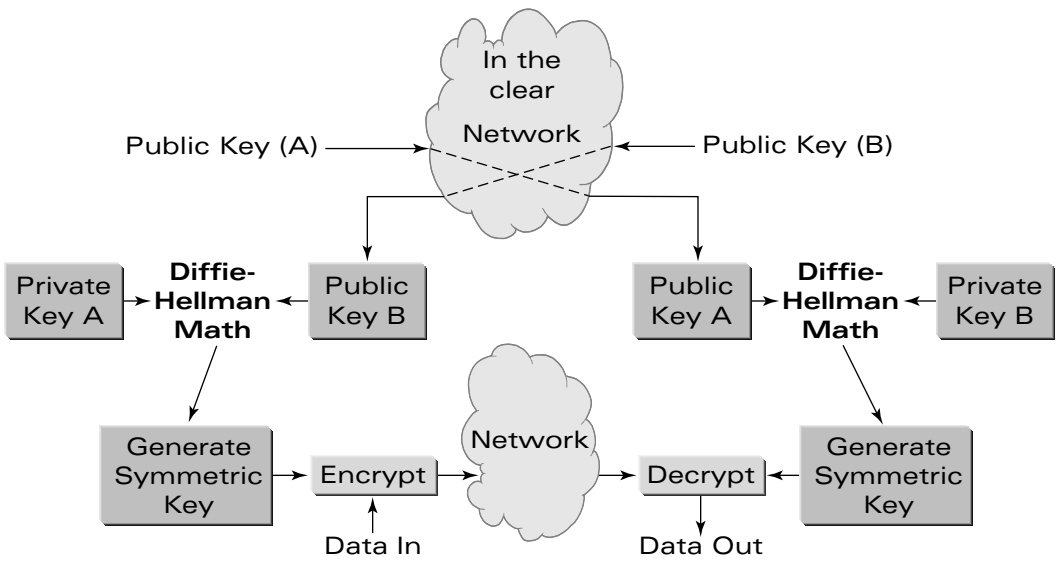


Figure 8

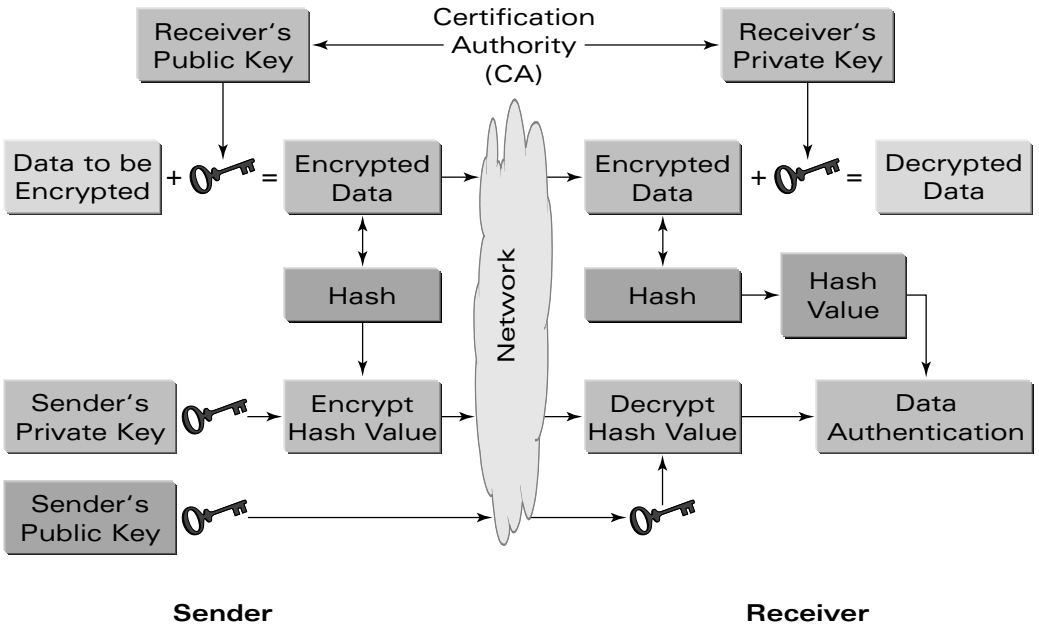


Figure 9

types of cryptography described, it is infeasible to recreate the original plaintext or ciphertext from hash, because hash is "one-way" nonreversible cryptography. For greater security, and to allow this scheme to provide authentication as well as integrity, a key can be included in the creation of the hash, the hash value can be encrypted before transmission, or both. Figure 9 illustrates this procedure with a public/private key pair being used to provide the authentication of the sender as well as the integrity of the received data. Note that in this case, it is the private key that is used to encrypt the hash value. The public key of the pair can then be used by the recipient to decrypt the hash value. The fact that the public key decryption succeeds is proof that the private key (known only to the sender) must have been used during the encryption process. This provides a level of assurance of the source of information.

The advantage of hash function cryptography is that a single scheme can provide both integrity checks of received data and authentication of the source. The disadvantage of hash function cryptography is that the hash computation can be computationally intensive, and the time it takes is added to the communication latency, as the received information cannot be used until the received and computed hashes have been proven to match.

Algorithms

Each cryptographic scheme above may be implemented with one of a number of ciphers. The ciphers chosen by the storage community will be described in the following, but first, a recap of the general principles of algorithms is given

and the terminology developed in the previous chapter is translated into more modern usage. Because all of the ciphers that will be described in detail are block ciphers, the general principles will focus on these. In addition, all of these ciphers would generally be used in symmetric cryptographic schemes, and ciphers suitable for use in asymmetric cryptography are omitted because of their complexity and because the point of this booklet focuses on storage network applications that require the high performance provided today only by symmetric cryptography.

General Principles

In general terms, a cipher takes two inputs, namely a key of a defined length and some plaintext, and produces one output, namely the ciphertext. While a stream cipher operates on each bit or character of the plaintext in turn, a block cipher operates on a defined number of bits or characters of the plaintext in parallel, and generally produces ciphertext of the same size as output. This process is illustrated in Figure 10. Note that the above description is for the sender, and that the roles of the plaintext and ciphertext are interchanged in the receiver.

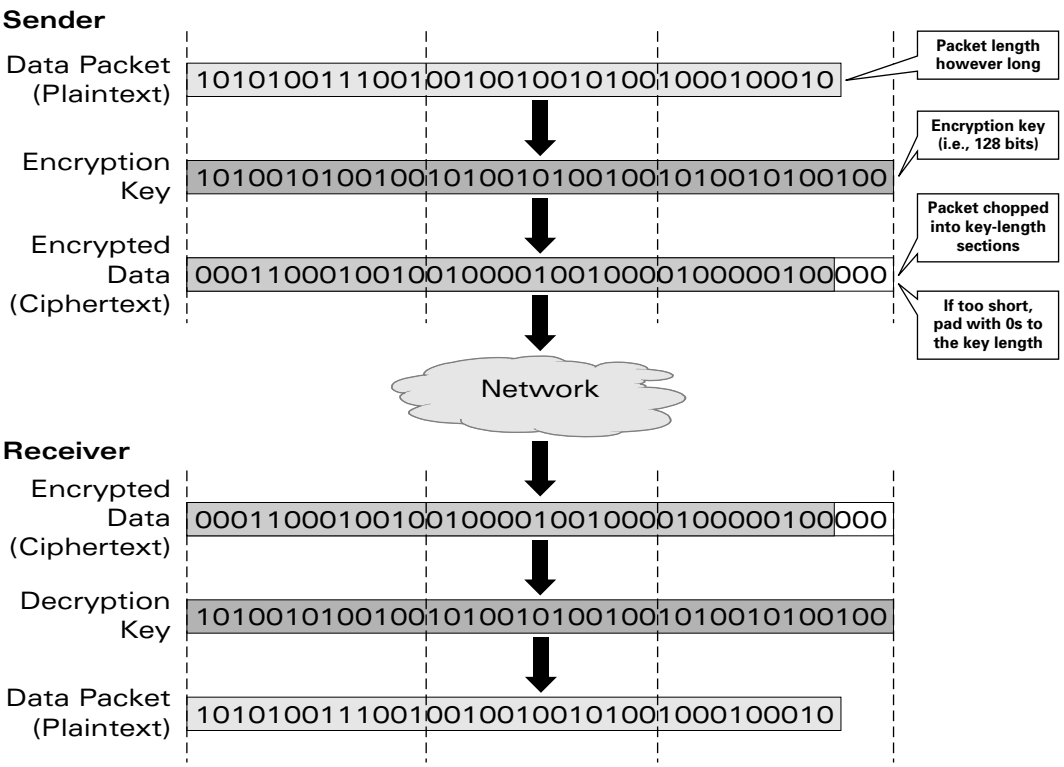


Figure 10

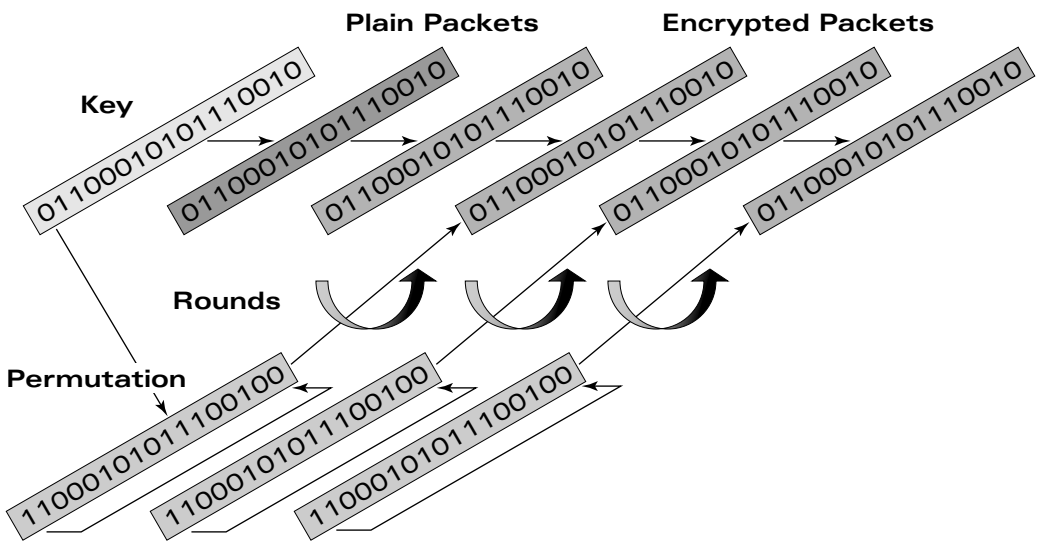


Figure 11

The process described above is known as a "round," and modern ciphers normally involve multiple rounds in which the (ciphertext) output of one round becomes the input to the following round, along with a permutation or shifted version of the previous key. This process is often repeated a number of times, as shown in Figure 11.

Note that the strength of most ciphers, including those described here, is very much subject to the randomness of the chosen key. Here true randomness, not pseudorandomness, is the order of the day. If an attacker can make even substantially inaccurate guesses about the choice of the key, it can be much easier to break a specific transmission. The majority of the supposedly random number generators incorporated in many programming languages are at best pseudorandom and are not of sufficient quality for use in cryptographic applications. A variety of schemes have been tried to create true randomness, including the management of white noise and the intervals between key presses.

Most ciphers also become less and less secure as more data is encrypted with the same key value. And with the growing speed of the interconnections being protected by the ciphers, it becomes more necessary to build a method of rekeying into the cryptographic schemes, that is, a secure method of providing a new key value and then indicating to the receiver that the new value should be used. The amount of data that can be safely transferred with one key value is a function of the key length itself (longer keys normally last longer) and of the characteristics of the particular cipher. For 10-gigabit interconnections, rekeying intervals in the order of minutes are anticipated for some popular ciphers.

Some ciphers also have defined key values that are known as weak keys. These provide an advantage to attackers, and use of them should be avoided.

Data Encryption Standard

The Data Encryption Standard (DES) is a block cipher designed for use in symmetric cryptography that encrypts data in 64-bit blocks and uses a key length of 56 bits. Actually, a key of 64 bits is used, but every eighth bit is ignored and these bits can be used for other purposes such as a parity check to ensure that the key is error free. The cipher consists of an initial permutation, after which the block is broken into a right half and a left half, each of 32 bits, followed by 16 key-dependent rounds on each half, after which the resulting halves are joined and the final permutation (the inverse of the initial one) is performed.

Two modes are popular with this cipher: an Electronic Code Book (ECB) mode, in which each block of the message is encrypted independently, and a Cipher Block Chaining (CBC) mode, in which each plaintext block is Exclusively-OR'd with the previous ciphertext block before encryption.

Triple DES

Triple DES (3DES) is, as its name implies, three DES processes executed one after the other. This is considered twice as secure as DES itself, which gives it an effective 112-bit key length. This cipher therefore involves 48 key-dependent rounds, which of course implies a larger processing latency.

As before, this cipher can be used in a number of modes. One mode involves three DES encryptions performed serially with three different keys; another involves using the same key for the first and third encryptions; and there are even modes in which the second operation is performed as a decryption. Of all of these, the two key, three encryption mode seems to be the most popular.

Advanced Encryption Standard

The Advanced Encryption Standard (AES) is the result of a competition held by the National Institute of Standards and Technology in the U.S. in 1997. It is sometimes still called Rijndael, because that was the name of the candidate algorithm that was selected.

AES is a block cipher designed for use in symmetric cryptography that encrypts data in 128-bit blocks, and can use key sizes of 128, 192, and 256 bits. The number of rounds required varies by the key length, being 10, 12, or 14 rounds for the key lengths shown above, respectively. The processing in each round is much more efficient than DES, and is much better suited to high-speed parallel operations in hardware. Before the rounds are performed, a step in which a subkey is XOR'd with the plaintext block is performed, and after the rounds the mix column operation occurs.

Several modes are being developed for use with AES. In addition to equivalents of the ECB and CBC modes described above, there is a counter mode in which a sequence number is Exclusively XOR'd with the plaintext before processing and then incremented for use with the next block.

Because AES is very new, it has been used in limited applications to date, but is expected to replace 3DES in time, with very high speed hardware implementations on the horizon.

Secure Hash Algorithm

The Secure Hash Algorithm (SHA-1) is currently the algorithm of choice for hash function cryptography. The National Institute of Standards and Technology originally designed SHA-1 for use with the digital signature standard. SHA-1 operates on messages that are padded to be a multiple of 512 bits (and less than 2^{64} bits) in length, involving four rounds of 20 operations each to produce a 160-bit hash value.

How Ciphers Are Cracked

There are two generic groups of methods by which ciphers are cracked. The first group attempts to go directly to the key, either by "brute force" (that is, by trying all possible keys against a message), or by compromising stored keys in some way.

Table 1 gives estimates of the time necessary to break the various DES schemes using brute force by three different types of attackers with vastly different budgets. These time estimates were obtained from a number of different published sources, listed in the footnote of the table. The effect of increasing key length, assuming truly random keys of course, is clearly evident and the three-key version of a 3DES is completely secure for all practical purposes.

TABLE 1 Cost and Time to Break DES Keys

		Time to Break Key		
Type of Attacker	**Budget**	**40-Bit**	**56-Bit**	**112-Bit 3DES**
Individual Hacker	$400	5 Hours	38 Years	Too long
Dedicated Hacker	$10,000	12 Minutes	556 Days	10^{19} Years
Intelligence Community	$10 Million	0.02 Sec 0.005 Sec*	21 Minutes 6 Minutes*	10^{17} Years

Source: Blaze, et al. (1996). "Minimal key lengths for symmetric ciphers to provide adequate commercial security," report by an ad hoc group of cryptographers and computer scientists, www.fortify.net/related/cryptographers.html, Chicago, IL. Schneier, B. (1996). *Applied Cryptography,* Second Edition. New York, New York: John Wiley & Sons, Inc.

*Kessler, Gary C. (1999). "An overview of cryptography," www.garykessler.net/library/crypto.html. Colchester, VT: Gary Kessler Consulting.

Note, however, that the increasing speed of networks actually works against security, because the more data that an attacker can collect and analyze, the faster a cipher can be cracked.

The normal types of compromises used are social engineering (obtaining the key by deceiving someone who knows it), obtaining access to systems containing the key, or being able to observe such systems and making deductions based on environmental factors. As an example of this latter category, the latency in processing a message, or the power consumed in decrypting a message might indicate whether asymmetric or symmetric cryptography is being used. Certain ciphers also have weaknesses based on data patterns, and if the attacker is able to cause information of his choosing to be encrypted, known as a "chosen plaintext" attack, this can also make it much easier to deduce the key.

The second group attempts to go directly to the plaintext. Many of these methods rely on problems with initializing ciphers, such as the same counter value being used continuously in error. So-called collisions in Cipher Block Chaining mode, where two different plaintexts produce the same ciphertext, can also give an attacker useful information.

An example of a cipher that is vulnerable to this type of attack is of the Wired Equivalent Privacy scheme used by wireless LANs. This uses a stream cipher that has problems with data loss, a 40-bit key shared by all users on the same segment, no rekeying, a poor choice of initialization vector (stream ciphers are particularly prone to initialization problems), and a CRC as an integrity check. The result of all of the above is that tools are available to passively monitor traffic (easy to do on a wireless LAN) and to derive the key being used from only a few hours of traffic while using limited computing resources (a single laptop of average processing power).

CHAPTER

Storage System Risk Assessment

With a storage network, as with any equipment or system, the first stage in determining the appropriate level of security is an assessment of risk. During this assessment, the exposure to various types of attack is determined and quantified, and then countermeasures are proposed and their effectiveness evaluated.

It is most important that this process be done in quantifiable rather than general terms. Security costs money, both directly for the additional hardware and functions involved, and indirectly because of the reduced usability and performance of the secured equipment. A cynic has noted the following rule: As security increases, the cost tends to go to infinity, and the supported performance to zero. Therefore, if a storage network is to be secured, both the direct and indirect costs involved must be justifiable. Performing the risk assessment process described in detail in the following sections should provide the basic data for this justification.

The sections that follow the description of the risk assessment process seek to apply it to a generic storage network configuration. Because of the limitations inherent in this generic approach, it is only possible to give a very limited quantification in this booklet. The process described here, and specifically the questions raised in the consideration of the generic SAN, will provide a basis for interested parties to conduct their own assessments of real (actual or planned) storage networks and produce the quantified results necessary to justify the inclusion of security features.

Assessment Process

The risk assessment process described below is generic in that it can be applied to a number of different situations, but it is particularly appropriate for storage networks.

The process has 10 steps, as follows:

1. Identify resources to be protected.
2. Identify categories of risk (e.g., confidentiality, authentication, and data availability).
3. Identify attack points (vulnerable places).
4. Identify methods of attack at each attack point for each category.

5. For each method, categorize the expected loss (low, medium, or high).
6. Estimate threats (who may attack) and probabilities.
7. Calculate the severity of risk (= probability of threat × expected loss).
8. Develop a countermeasure for each attack method.
9. Estimate the reduction in severity of risk for each countermeasure.
10. Estimate the cost effectiveness of all the countermeasures for the system (benefits vs. direct and indirect costs).

Like any good process, multiple iterations of some or all of the steps will probably be necessary to achieve the desired result, which is a level of cost effectiveness that is appropriate for the situation and addresses the most severe security risks.

The application of this process in qualitative terms to a generic storage network is described in the following sections.

Resources and Categories of Risk

Figure 12 shows a generic storage network, consisting of a number of servers with host bus adapters connecting to an infrastructure element (a generic term used to cover hubs at all types of switches), to which are also connected a number of storage devices. For reference, the figure also shows a number of information sources/sinks (users, agents, or applications), which are the ultimate creators and consumers of information held within the storage network.

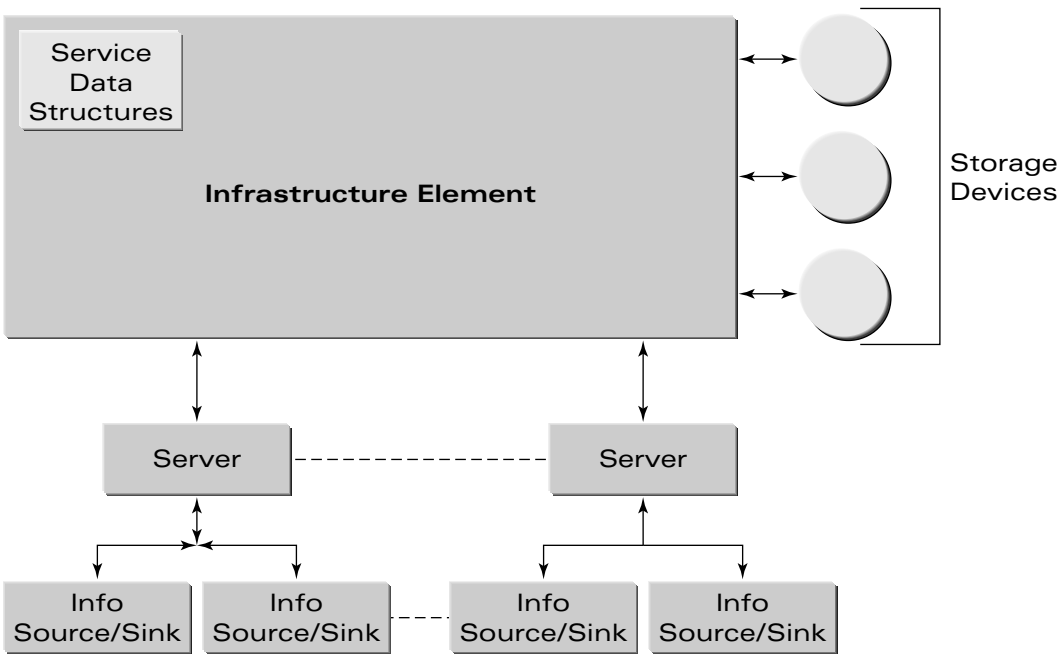


Figure 12

An analysis of this configuration leads to the conclusion that there are two types of resources to be protected in this situation: the data held by the storage devices, and the communications ability of the storage network itself.

Categories of risk against the data held by the storage devices are unauthorized access to the data, undetected deletion and modification of the data, and the creation of false data. The first of these categories specifically depends on the identification of the accessor of data and can be the basis for a requirement of some type of authentication.

Categories of risk against the communications ability of the network are the availability of the infrastructure (which leads to the unavailability of the data referenced above), the inability to discover and manage configuration changes, and the rerouting, corruption, and deletion of data in flight.

For many of the existing and planned storage networks, the connections between the servers, infrastructure elements, and storage devices are defined by Fibre Channel (FC), and use the shorter-length physical variants, so that the entire storage network is contained within the data center. Similarly, storage networks based on TCP/IP tend to use a separate infrastructure from the corporate LAN, and this is also completely contained within the data center.

A superficial analysis may therefore lead to the conclusion that a storage network as described above can be physically secured. The door to the data center can be locked, and the use of an access control system based on badges or magnetic cards should limit the number of threats to the system. However, a deeper analysis indicates that this is a dangerous fallacy.

Attack Points

It is an established truism in security that defenders have to be concerned with protecting an entire perimeter, whereas attackers only have to choose a single weakest point. A full analysis of the generic storage area network reveals that there are many more attack points than previously identified, and therefore a truer picture of a generic SAN is that shown in Figure 13. The major difference between this figure and the previous one is that it shows that the infrastructure elements and the storage devices have two additional classes of interface, namely out-of-band management interfaces and control terminals. All of the interfaces are described in detail in the following sections.

Inband Interfaces

The inband interfaces consist of the interconnections between servers and storage devices and an infrastructure element. With the advent of 10-kilometer Fibre Channel physical variants, as well as TCP/IP-based interfaces, it is no longer clear that these interfaces can be completely protected by physical security. There are established security mechanisms that can be used to protect the TCP/IP systems, including IPsec and the Secure Sockets Layer (SSL) protocols, but so far none of

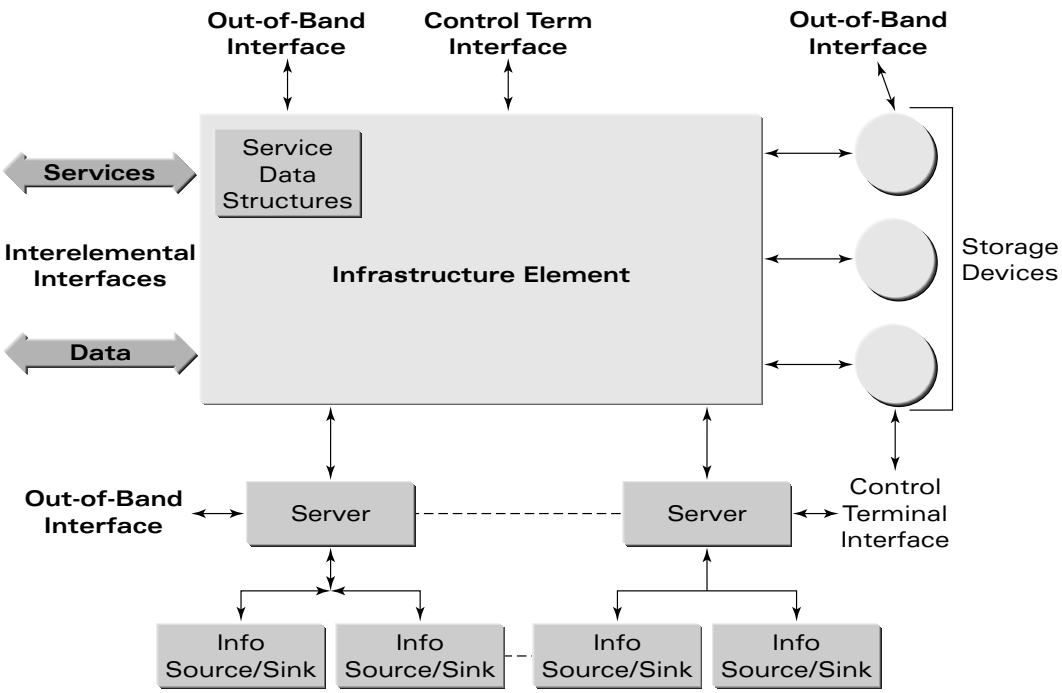


Figure 13

these techniques are being widely deployed. In addition, all of them have significant performance impacts at the present gigabit speeds and may be impractical at higher speeds. While Fibre Channel does not yet have a confidentiality option, a project is under way to allow any scheme modeled on one of the more popular options of the IPsec to be applied to Fibre Channel, and more details are given in Chapter 7. Fibre Channel devices are also required to have node worldwide names, and these may be optionally used for authentication purposes. Such a scheme is of course not completely secure, but has proven to be useful in the field. Some of the features of the infrastructure element that can be used for security in a Fibre Channel storage network are described in detail in a later section.

Management Interfaces

Nearly all of the infrastructure elements in a storage network, and many of the higher end storage devices, have an out-of-band interface (such as an Ethernet port) for management purposes. This typically has an SNMP agent capable of reporting management state and statistics to a management console, or a simple Web server and a number of Web pages designed to provide facilities for controlling and configuring the device.

The feasible methods of attack through this point are related to the accessibility of the interface and of the features provided behind it. In situations where the interface is connected to the corporate LAN, or through standard remote access privileges to the corporate LAN, the accessibility may be quite straightforward for an attacker. However, even if such basic connectivity is provided, if the interface forces use of a Virtual Private Network (VPN) protocol, or IPsec or one of the other security protocols mentioned in the previous section, the accessibility may be significantly diminished. A similar effect may be obtained by forcing the use of special credentials (username and password) to access the interface, and not allowing access using standard e-mail or network use credentials.

Depending on the specific device, some features may be accessible through this interface that have a significant effect on the operation of the entire storage network. In the case of an infrastructure element, it may be possible to access its name service and either create or delete entries within that service, causing existing devices to disappear or phantom devices to be created. There may also be test and diagnostic features that can be accessed through this interface that can be suborned by an attacker to provide significant disruption. For instance, an ability to create an arbitrary frame or packet and have it routed by the network could be used to issue a SCSI format command from an address recognized as a legitimate server, and the result will be the untraceable destruction of vital data. In the case of a storage device, it may be possible through this interface to create, delete, or modify existing volume definitions. It also may be possible to suborn the access control system contained in the device to prevent access by various other SAN-attached devices.

Control Terminals

Many infrastructure elements and some storage devices also have a serial interface intended to be connected to a "control terminal." Though some years ago a standard dumb terminal would have been connected to this interface, it is more usual now to connect a PC device running a terminal emulator. This interface is typically used to provide the basic configuration setup for the equipment, and to perform such tasks as updating firmware and reading or clearing logs.

Again, the methods of attack that are possible through this point relate to the accessibility to it and the features provided behind it. If the serial interface has no device connected to it unless a maintenance procedure is being performed, then accessibility will be very limited. However, it is common to leave a PC connected to this interface permanently and also to connect that PC to the corporate LAN. Where this is done, it is usually for the purpose of providing remote access to facilitate off-hours maintenance. However, in this case all of the comments on accessibility of the out-of-band interface given in the previous section also apply. In particular, in this situation it is vital that the credentials needed to access the control terminal interface be secure and used for that purpose alone.

Depending on the specific device, the features accessible via this interface may have a major impact on the operation of the entire storage network. For an

infrastructure device, it may be possible to load a set of outdated or unproven firmware. It may also be possible to access the service data structures contained within the element and manipulate or remove access control lists and other vital configuration data. For storage devices, it may also be possible to change firmware, to disable interfaces, and perhaps even change the versions of a command set that the device supports. For all types of devices, it may be possible to change both address information and identification information like FC worldwide names or iSCSI names.

Interelement Interfaces

Storage networks that contain more than one infrastructure element have interelement interfaces, and though these may use the same physical interface as the interfaces to the storage devices and servers, they do require some special consideration. A much larger percentage of the traffic carried by the network infrastructure flows over these interfaces than over a server or storage device interface, and some of the control traffic between infrastructure elements has a key role in configuring and managing the fabric. In addition, new technologies specific to this situation are becoming available, including wide-area network extenders based on the developing FC-IP standard and other vendor-specific solutions.

Where a storage network is spread over multiple locations on a campus, it is not unusual for these interfaces to leave the data center and be routed through the usual building components, including patch panels, wiring closets, and various cable trunks. Maintenance access has to be provided to a number of these components in order for the corporate LAN and the corporate phone system to be serviced, and the same access can therefore be used for these interelement links. It is often completely impractical to separate the storage network infrastructure from the other building wiring at this level. Therefore, access to these interfaces will always be possible.

Methods of attack that are possible through this point include attaching an authorized infrastructure element that causes all of the addressing in the rest of the storage network to change, and by such a method even the fabric name can be changed. Some infrastructure elements also employ a common port type that discovers its function at system initiation, and in that case it may be possible to connect a server to a port that is intended to be an interelement link and thus to access the service information contained in other elements. It might also be possible at this point to passively monitor and collect large amounts of the storage network traffic and use that as the basis of an offline attack, or perhaps even modify the traffic in passing. And there are substantial possibilities for denial-of-service attacks by deleting traffic from these interfaces, repeating illegal traffic, or generating false traffic.

Because of the possibilities outlined above, the interelement link is one of the few places in current storage networks that incorporate confidentiality. Some of the extenders mentioned above use existing network schemes such as IPsec to pro-

vide authentication and integrity protection as well as confidentiality for the traffic flowing across the wide-area network.

Threats and Probabilities

It is serious fallacy to believe that most of the threats against a storage network will come from outside of the data center. In fact, the most serious threats—and those with the highest probability—come from situations that occur within the data center. Current research has indicated that the most serious security threats originate with data center staff and with operating systems and other applications running on the servers in the data center that are not designed to handle the concept of sharing storage.

It is also a mistake to believe that all threats are intentionally malicious. During a presentation of the SNIA tutorial on which this booklet is based, the authors were made aware of a corruption of Occam's Razor that apparently originated in the hacker community. It is known as Hanlon's Razor, and it states:

> "Never attribute to malice that which can be adequately explained by stupidity."

The maintenance procedures for storage networks are so complex and so intricate that, even with the best of checking procedures and continuous practice, mistakes do happen. Stories abound about administrators of storage networks making a single bit error in their infrastructure configuration which allows an NT-based server to access a Unix disk, which it subsequently formatted with a considerable loss of data.

Other identified threats are people with access to the storage network via the out-of-band maintenance or control terminal interfaces, users with legal accounts on the servers connected to the storage network who exploit problems in the server operating environment's security, and theft and misuse of credentials that allow special access to the storage network.

The most probable threats are data center staff and server operating environment issues. Server user and credential theft threats are very much dependent on the creation and enforcement of policies with specific characteristics, which will be discussed below.

Risk Severity, Countermeasures, and Cost Effectiveness

These three final stages in the risk assessment process are very difficult to describe in terms of a generic storage network. Clearly, the severity of a specific risk, represented by a probable threat employing a defined attack method against a specific attack point, is dependent upon the details of a particular implementation of a storage network. Qualitative expectations are all that can be offered, given this generic description.

From the preceding descriptions it would seem that the security risk caused by data center staff misinterpreting or not understanding procedures and using both the normal access to storage from a user account on the server, or the out-of-band maintenance or control terminal interface attack points described previously, would be the most severe. The accessibility of the latter attack points can be mitigated by implementing specific policies addressed in the following chapter. The lack of understanding of shared storage by many applications and operating environments also has the effect of significantly increasing the expected loss from the above risks. Policies to mitigate the effect of credential misuse or theft are not specific to storage networks, but nevertheless may have a significant impact on the relative severity of some specific risks.

CHAPTER

Current SAN Security Tools and Practices

The "toolbox" of storage network countermeasures will be described in detail but is not as yet very extensive. Most of the tools are currently concerned with access control, and some facilities for a very basic form of port-based authentication are also available.

The good news is that many of the tools described here are low cost or free in that they are inherent features of components of the storage network, but they are quite effective in mitigating the severity of a range of risks. Obviously the specifics, and the quantitative effects, depend on the characteristics of the real storage network, its intended use, and its relationship to various business assets.

Few of the tools described in this chapter were conceived for their security properties alone. In fact, most of them originated due to various operational concerns or because of the need to support software that had been developed for previous generations of storage interconnections—interconnections that supported far fewer devices than a storage network. Despite this, these tools have considerable value and are quite effective in mitigating the impact of various security problems, if not eliminating them entirely.

The majority of these tools are forms of access control, but the exact instantiation of the controls is a function of the type of interface used in the storage network. In a Fibre Channel network, zones are used, but in a TCP/IP network, one of two equivalent schemes is normally employed, and such schemes are described separately. Another type of access control is logical unit mapping and masking, and this is described in generic terms because such a facility can occur at many points within a computer system. This chapter concludes with a set of requirements for security policies, and the identification of current best practices for storage networks.

Access Control

Two different types of access control are available. These are zone controls, which operate at the Fibre Channel port level and logical unit masking and mapping controls, which operate at the SCSI command set level.

Fibre Channel Zoning

The Fibre Channel standards define a zone as being a subset of all the components of a storage network that are aware of each other and can communicate. A number of different parameters may be employed in a zone definition, including the infrastructure port number, the port's worldwide name, the component's address identifier, or the component's (device's) worldwide node name. Both hard zones and soft zones are supported, the former being enforced by the infrastructure (which will prevent information being passed between components that are not members of the same zone) and the latter being defined only by software or firmware within each of the components. The infrastructure definitions further include the concept of a zone set, which is one or more zones that may be activated or deactivated as a group. Both zones and zone sets may be assigned names for ease of management. An infrastructure will typically be able to maintain the definition of several zone sets, but only one will be active at any one time.

Zones were created to limit the number of devices that could be seen by each system for administrative and operational reasons. However, this feature can also clearly have a strong security function. In particular, it is the primary defensive mechanism employed in storage network configurations containing servers with multiple types of operating environments that are not aware of one another.

Logical Unit Mapping and Masking

The logical unit mapping and masking facilities have their genesis in the management facilities provided by some early high-end disk arrays. These products were capable of reporting a single physical or virtual disk volume to different SCSI Initiators with differing logical unit numbers (LUNs). This process is known as Logical Unit (LU) mapping. Some of this equipment also had the capability of preventing a specific SCSI initiator from detecting or accessing some volumes, and this process is known as LU masking.

In recent times the logical unit mapping and masking facilities have been incorporated at a number of points in the path that an application uses to access storage. Many of the host bus adapters that are used to access storage networks incorporate one or both of these facilities. In particular, many modern virtualization engines, whether they be hardware or software, incorporate such facilities as a matter of course. It is not unusual for a single request to transit a number of these facilities in traveling between an application and a storage device.

Clearly these facilities can have a strong security function. By imposing a limited scope of access to the SAN resources by any particular component, they can significantly mitigate the severity of some security risks. They have a major weakness, though, in the fact that no central control of all the mapping and masking facilities in the storage network is normally possible. Manually configuring the multiple separate facilities, each with a different type of management interface, to try to achieve a consistent security scheme is a process fraught with difficulties and potential errors.

TCP/IP Security Mechanisms

A suite of security protocols and functions called IPsec, that has been developed to protect the TCP/IP protocols used on the worldwide Internet, has been referenced before in this booklet in the context of a local-area network. These protocols are capable of providing authentication, confidentiality, and integrity protection on a selective basis for higher level protocols supported by the TCP/IP infrastructure.

The iSCSI protocol, which is under development within the IETF and which defines how SCSI command sets are transported by the TCP/IP infrastructure, makes a subset of IPsec mandatory to implement, if not to use. iSCSI also makes use of a further protocol, a secure remote password (SRP) scheme, to provide mutual authentication of the iSCSI Initiators and targets.

Two different schemes have been implemented by TCP/IP infrastructure vendors to provide equivalent functionality to the zones in Fibre Channel. These are called a virtual private network (VPN) or a virtual LAN (VLAN). Again, the schemes define which pieces of equipment are permitted to communicate on a basis other than the physical topology of the infrastructure. VPNs or VLANs can be defined on a number of bases, including the IP address or MAC address, and the differences between them are outside the scope of this booklet. At the highest level, a VLAN is formed by configuring one or more network routers to permit only certain routes, and a VPN normally involves the use of a tunneling protocol with confidentiality features to create an overlay network. Both have the same substantial security impact as for zones described above.

Policy Requirements

A good security policy is an important tool for creating good security. The policy should be implemented through procedures and guidelines, be enforceable with security tools and sanctions, and should clearly assign responsibilities, accountabilities, and penalties. The policy should cover privacy, authentication, confidentiality of specific types of information, requirements for backup and restore, and required levels of monitoring and auditing. Wherever possible, the implementation of the policy should result in overlapping and reinforcing layers of security protection. Both data residing on storage devices and data in transit through the infrastructure should be considered. The policy should seek to protect the operation of the storage network and control or eliminate attack points that have the most effect on the operation of the storage network.

The following items should specifically be considered for inclusion in the security policy:

1. System upgrades should be promptly installed, but only after first testing them on an isolated nonproduction system.
2. Only proven technologies should be installed in the storage network. Specifically, references should be sought for all new technologies in exactly the same way as they would be sought for new employees.

3. Collaboration with other organizations with similar storage networks and configurations should be encouraged. Regular pooling of resources and experience of security problems can be most beneficial in raising storage network security.
4. Wherever possible, key servers in a storage network should be "hardened." This involves strictly restricting the types of applications that can be installed on the servers, and the credentials that give access to the servers.
5. Security audits should be performed frequently, and system logs should routinely be scrutinized for unusual activity.
6. An awareness program should be conducted for key employees to keep them up to date with expected threats and countermeasures.

Best Practices

The following best practices in some cases repeat points that have been made elsewhere in the booklet, and in other cases relate to the specific experiences of storage network creators and managers who provided feedback on the storage network tutorial on which this booklet is based.

The recommended best practices are:

1. Make sure you've identified all of the interfaces to your storage network.
2. Create a separate infrastructure for the out-of-the band management and control terminal interfaces to the storage network. If connectivity is required to the corporate LAN, provide it via a firewall or a secure router. Provide a dedicated remote access facility if this type of access is required, and use all of the appropriate network security tools, such as virtual private networks.
3. Use dedicated user IDs for storage network maintenance access, and enforce the use of strong passwords either by policy or by configuration. Use separate credentials again for infrastructure configuration functions.
4. Define zones containing the smallest possible number of components, and use different zone sets for different system loads, such as the off-hours backup time.
5. Control access to all of the unused ports in the storage network infrastructure. Wherever possible, configure the infrastructure element so that unused ports must be specifically enabled before use and so newly attached devices are not automatically added to any zone. Always use hard zones in preference to soft zones.
6. Only install software and firmware on storage network components from authorized sources, and never do so when a device is connected to a production storage network. When such a procedure is necessary, swap out the equipment and use an isolated storage network for the process. Where possible, configure storage devices to not accept firmware upgrades via the storage network interfaces.
7. Always change default passwords before equipment is connected to a production storage network. Ensure that strong passwords are required by policy, and educate key personnel as to their importance.

CHAPTER

Future Security Tools

A number of standards on which future security tools will be based are in development in iNCITS, the IEEE, and the IETF. In addition, a number of research projects are under way in academia to investigate different architectures and approaches that can be used to make computer systems more secure and more resilient to attacks.

This chapter gives details on all of the standards developments, and on one particular academic initiative. Note, however, that by definition all of these activities are ongoing and the results are changing rapidly. For the latest information, therefore, it is recommended that the sources identified in Appendix B be consulted.

iNCITS

The iNCITS Technical committee T11 is creating standard definitions related to security in two areas. Firstly, it is enhancing the Fibre Channel frame formats with the definition of the new optional header, which will contain a close analog of the encapsulating security payload (ESP) header defined by the IPsec scheme previously defined by the IETF. This optional header will allow Fibre Channel SANs to be extended in a backward-compatible way to support authentication, confidentiality, and/or integrity protection as required by a specific configuration. The use of a common format with IPsec should promote the reuse of existing firmware and software and also hardware function macros. Secondly, the committee is producing a Technical Report called FC Security Protocols (FC-SP) to define how components of a storage network may authenticate each other and cooperate in the distribution of key material.

Note that an existing T11 standard already incorporates support for integrity protection. The Fibre Channel third-generation Generic Services draft (FC-GS-3) describes a number of common services provided by fabric, such as a name service, a management service, and a service to create and manage zones and zone sets. FC-GS-3 incorporates a common transport definition capable of including an optional preamble that may contain a security association identifier, and a field that can contain a hash block to be used in authentication and integrity-verification mechanisms.

IETF

The IPsec Working Group of the transport area of the IETF is extending the existing Internet key exchange (IKE), an IPsec protocol to support traversal of network address translators and firewalls, and use of new transports such as SCTP. The group is also working on new cipher documents that use AES, including Cipher Block Chaining and Message Authentication modes and a fast counter-based mode suitable for hardware-based encryption.

The IP Storage Working Group has also produced a draft that defines security requirements for its three families of standards, iSCSI, iFCP, and FC-IP. This draft gives excellent guidance, much of which is applicable to storage networks in general and not just the three standards identified previously.

SNIA

Much of this booklet is based upon the existing work of the SNIA Security Technical Working Group. The group is continuing its investigation of storage network risk assessment, and has plans to produce a more detailed set of best practices for storage network management and operation in the coming year. The group's educational activities are also continuing, with the refinement of existing tutorials and production of new collateral items of which this booklet is the first. The group will be participating with other entities within SNIA to define a new management interface (based on the submission called Bluefin from a group of SNIA member companies) and will be specifically considering how access control should be defined and enforced on this interface.

IEEE

IEEE has formed a Task Force on Information Assurance, which champions a holistic approach to development of information assurance technology by asserting an information assurance view across numerous closely related technologies: networking, software engineering, distributed processing, pattern recognition, real-time computing, visualization, cryptography, simulation, man-machine interface, data engineering, mass storage, and others. Within this organization, a newly formed Security in Storage Working Group (SISWG) is seeking to create original cryptography to meet requirements unique to the security of information at rest on a mass storage device. An example of such a unique requirement is the need to be able to determine that a backup information set retains integrity without having the necessary credentials to access and read the data that it contains.

Academic Initiative

The Parallel Data Laboratory at Carnegie-Mellon University in Pittsburgh, Pennsylvania, has a considerable history of original work in storage system research. Recently, with funding from the Department of Defense's Critical

Infrastructure Protection program, the laboratory has defined a self-securing storage device.

Self-securing devices promise greater flexibility for security administrators dealing with intrusions. By having each device erect an independent security perimeter, the storage network gains many strong points of defense from which to act when under attack. Such devices can not only protect their own resources, but they can observe, log, and react to the actions of other nearby devices. In such an architecture, infiltration of one security perimeter will compromise only a small fraction of the environment and a small amount of the total amount of information stored within the network. When an infiltration is detected, other devices can work to dynamically identify the problem, alert still-secured devices about the compromised components, raise the security levels of the environment, and so forth.

Given that the role of storage devices in computer systems is to persistently store data, a natural security extension is to protect stored data from attackers, preventing undetectable tampering and permanent deletion. A self-securing storage device does this by managing storage space from behind its security perimeter, keeping an audit log of all requests, and keeping clean versions of data modified by attackers. Since a storage device cannot distinguish compromised user accounts from legitimate users, all data versions must be maintained. Finite capacities limit how long such comprehensive versioning can be maintained, but 100% per year storage capacity growth will allow modern disks to keep several weeks of data in many situations. If intrusion detection mechanisms reveal an intrusion within this detection window, security administrators will have this valuable audit and version information for diagnosis and recovery, as well as for the retrieval of otherwise deleted information.

See Appendix A for references to more information on this subject.

CHAPTER

Summary

This booklet has provided an introduction to the theory of cryptography from a historical perspective, and given an overview of the different types of cryptography that are employed today and their applications. The most popular ciphers used in these types of cryptography have been introduced, and references to further details are contained in Appendix A. A general risk assessment process, which can be used to assess the security requirements of a number of different types of systems, has been formulated. With specific relation to storage networks, the process has been applied in a generic manner resulting in the qualitative identification of attack points and methods, threats, and threat probabilities. The countermeasures that are currently available in storage networks have been described and their effectiveness gauged. The present state of a number of developments of future security technologies has been presented, and their impact anticipated. In short, the booklet has provided a focused basic primer in security for members of the Storage Networking Industry Association. Those members are requested to use the information provided here, and in the references contained in the appendices, to enhance their products so as to increase the level of security possible in deployed storage networks, thus literally fulfilling the "trusted" part of the SNIA mission.

Appendix A: Referenced Standards

The following is a list of industry standard activities that define, or are relevant to, the security and interface technologies described in this booklet. The list is divided into two sections, one of which deals with completed and/or published standards and the other with known works in progress. Wherever possible, sources of each document are identified, and whether they are available online or only in print.

By definition, all of these documents have a life cycle, and therefore new standards are constantly being approved, new versions are becoming available, and old versions are being withdrawn. Therefore, if a specific version referenced here is not available, it is recommended that the relevant accredited standards organization be contacted to obtain the most up-to-date information. The contact information for each accredited standards body mentioned is given at the end of the list.

Published and/or Completed Standards

Chapter 3 A Brief History of Cryptography

Information about public key encryption can be obtained from http://www.rsa.com, and from the following published documents:

IETF RFC1170: Public key standards and licenses. R. B. Fougner. January 1991. (Format: TXT=3144 bytes)

IETF RFC3029: Internet X.509 Public Key Infrastructure Data Validation and Certification Server Protocols. C. Adams, P. Sylvester, M. Zolotarev, R. Zuccherato. February 2001. (Format: TXT=107347 bytes)

IETF RFC2510: Internet X.509 Public Key Infrastructure Certificate Management Protocols. C. Adams, S. Farrell. March 1999. (Format: TXT=158178 bytes)

FIPS-196: Entity Authentication Using Public Key Cryptography. February 1997.

Chapter 4 Standard Security Tools and Approaches

References for ciphers are:

FIPS-46-3: Data Encryption Standard (DES). October 1999.

FIPS-8:1: DES Modes of Operation. December 1980.

IETF RFC2405: The ESP DES-CBC Cipher Algorithm with Explicit IV. C. Madson, N. Doraswamy. November 1998. (Format: TXT=20208 bytes)

IETF RFC2419: The PPP DES Encryption Protocol, Version 2 (DESE-bis). K. Sklower, G. Meyer. September 1998. (Format: TXT=24414 bytes) (Obsoletes RFC1969)

IETF RFC2420: The PPP Triple-DES Encryption Protocol (3DESE). H. Kummert. September 1998. (Format: TXT=16729 bytes)

FIPS-197: Advanced Encryption Standard (AES). November 2001.

FIPS-180-1: Secure Hash Standard (SHS). April 1995.

IETF RFC2404: The Use of HMAC-SHA-1-96 within ESP and AH. C. Madson, R. Glenn. November 1998. (Format: TXT=13089 bytes)

FIPS-198: The Keyed-Hash Message Authentication Code (HMAC). March 2002.

Chapter 5 Storage System Risk Assessment

IP security is defined by:

IETF RFC2411: IP Security Document Roadmap. R. Thayer, N. Doraswamy, R. Glenn. November 1998. (Format: TXT=22983 bytes)

IETF RFC2406: IP Encapsulating Security Payload (ESP). S. Kent, R. Atkinson. November 1998. (Format: TXT=54202 bytes) (Obsoletes RFC1827)

IETF RFC2408: Internet Security Association and Key Management Protocol (ISAKMP). D. Maughan, M. Schertler, M. Schneider, J. Turner. November 1998. (Format: TXT=209175 bytes)

IETF RFC2409: The Internet Key Exchange (IKE). D. Harkins, D. Carrel. November 1998. (Format: TXT=94949 bytes)

SNMP is defined by the following RFCs:

IETF RFC1157: Simple Network Management Protocol (SNMP). J. D. Case, M. Fedor, M. L. Schoffstall, C. Davin. May 1990. (Format: TXT=74894 bytes) (Obsoletes RFC1098 and STD0015)

IETF RFC1441: Introduction to Version 2 of the Internet-Standard Network Management Framework. J. Case, K. McCloghrie, M. Rose, S. Waldbusser. April 1993. (Format: TXT=25386 bytes)

IETF RFC1901: Introduction to Community-Based SNMPv2. J. Case, K. McCloghrie, M. Rose, S. Waldbusser. January 1996. (Format: TXT=15903 bytes)

IETF RFC1902: Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2). J. Case, K. McCloghrie, M. Rose, S. Waldbusser. January 1996. (Format: TXT=77453 bytes)

IETF RFC1905: Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2). J. Case, K. McCloghrie, M. Rose, S. Waldbusser. January 1996. (Format: TXT=55526 bytes) (Obsoletes RFC1448)

IETF RFC1906: Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2). J. Case, K. McCloghrie, M. Rose, S. Waldbusser. January 1996. (Format: TXT=27465 bytes) (Obsoletes RFC1449)

IETF RFC1907: Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2). J. Case, K. McCloghrie, M. Rose, S. Waldbusser. January 1996. (Format: TXT=34881 bytes) (Obsoletes RFC1450)

IETF RFC1908: Coexistence Between Version 1 and Version 2 of the Internet-Standard Network Management Framework. J. Case, K. McCloghrie, M. Rose, S. Waldbusser. January 1996. (Format: TXT=21463 bytes) (Obsoletes RFC1452) (Obsoleted by RFC2576)

IETF RFC2571: An Architecture for Describing SNMP Management Frameworks. B. Wijnen, D. Harrington, R. Presuhn. April 1999. (Format: TXT=139260 bytes) (Obsoletes RFC2271)

IETF RFC2576: Coexistence Between Version 1, Version 2, and Version 3 of the Internet-Standard Network Management Framework. R. Frye, D. Levi, S. Routhier, B. Wijnen. March 2000. (Format: TXT=98589 bytes) (Obsoletes RFC1908 and RFC2089)

IETF RFC2578: Structure of Management Information Version 2 (SMIv2). K. McCloghrie, D. Perkins, J. Schoenwaelder. April 1999. (Format: TXT=89712 bytes) (Obsoletes RFC1902 and STD0058)

IETF RFC2579: Textual Conventions for SMIv2. K. McCloghrie, D. Perkins, J. Schoenwaelder. April 1999. (Format: TXT=59039 bytes) (Obsoletes RFC1903 and STD0058)

IETF RFC2580: Conformance Statements for SMIv2. K. McCloghrie, D. Perkins, J. Schoenwaelder. April 1999. (Format: TXT=54253 bytes) (Obsoletes RFC1904 and STD0058)

One reference for virtual private networks is:

IETF RFC2764: A Framework for IP-Based Virtual Private Networks. B. Gleeson, A. Lin, J. Heinanen, G. Armitage, A. Malis. February 2000. (Format: TXT=163215 bytes)

Telnet is defined by:

IETF RFC0854: Telnet Protocol Specification. J. Postel, J. K. Reynolds. May 1983. (Format: TXT=39371 bytes) (Obsoletes RFC0764 and STD0008)

Chapter 6 Current SAN Security Tools and Practices

Fibre Channel zones are defined by:

iNCITS 348-2000: Fibre Channel Generic Services 3.

iNCITS 355-2001: Fibre Channel - Switch Fabric - 2.

The SCSI command set is defined in:

iNCITS 351-2001: SCSI-3 Primary Commands 2.

For other related command sets, see the Technical Committee T10 Web site referenced on page 45 for a full list.

Chapter 7 Future Security Tools

Information on IPsec can be found in the referenced standards for Chapter 5 above.

Significant Works in Progress

Chapter 4 Standard Security Tools and Approaches

Several documents related to AES cipher modes can be found via the IETF IP Security Working Group page and the Internet Drafts directory.

Chapter 6 Current SAN Security Tools and Practices

New generations of both the Fibre Channel Generic Services (called FC-GS-4) and Switch Fabric (called FC-SW-3) standards are in preparation in iNCITS Technical Committee T11. In addition, a new generation of the SCSI Primary Command Set definition (called SPC-3) is in preparation in iNCITS Technical Committee T10, and it incorporates a new access controls feature.

Chapter 7 Future Security Tools

The new FC Optional Header that closely parallels the IPsec Encapsulating Security Payload (ESP) header is being defined as part of the FC Framing and Signaling Interface (FC-FS) project in iNCITS Technical Committee T11. An additional project called FC-SP (Security Protocols) is defining methods of key management in a fabric and protocols to allow SAN components to perform mutual authentication. As of this writing, no draft standard yet exists for this project, but details of a number of proposals for features to be included in such a draft can be viewed via the Web site of iNCITS Technical Committee T11.

The Web page of the IEEE Task Force on Information Assurance can be found at http://www.tfia.org, but to date, few draft documents are available.

For more information on the Parallel Data Laboratory at Carnegie Mellon University, see http://www.pdl.cmu.edu.

Accredited Standards Body Information

Internet Engineering Task Force (IETF)

The IETF's main Web site is: http://www.ietf.org. See: http://www.ietf.org/rfc/rfc-NNNN.txt, where NNNN is the RFC number prefixed with zeroes as necessary to make a four-digit number, to gain access to the IETF standards listed above.

See: http://www.ietf.org/html.charters/wg-dir.html to access pages of the active IETF Working Groups engaged in standards creation, and see: http://www.ietf.org/ID.html to access all current working drafts directly.

The IETF Secretariat is hosted by the Corporation for National Research Initiatives. It can be reached at:

IETF Secretariat
c/o Corporation for National Research Initiatives
1895 Preston White Drive, Suite 100
Reston, VA 20191-5434
USA
+1 703 620 8990 (voice)
+1 703 620 9071 (fax)

Institute of Electrical and Electronic Engineers (IEEE)

The main IEEE Web site is at http://www.ieee.org. See: http://shop.ieee.org/store/ to purchase copies of IEEE standards, and journals, magazines, and conference proceedings. See: http://grouper.ieee.org/groups/index.html to access pages of the active IEEE Working Groups engaged in standards creation.

The IEEE Secretariat is located at:

IEEE Operations Center
445 Hoes Lane
Piscataway, NJ 08854-1331
USA
+1 732 981 0060 (voice)
+1 732 981 1721 (fax)

International Committee for Information Technology Standardization (iNCITS)

The main iNCITS Web site is at: http://www.incits.org. See: http://www.techstreet.com/ncitsgate.html to purchase copies of iNCITS standards. See: http://www.incits.org/tcs.html to access pages of the active iNCITS Working

Groups engaged in standards creation. Also, for works in progress, see the Web sites of iNCITS Technical Committee T10 (responsible for standardizing SCSI) at http://www.t10.org, and iNCITS Technical Committee T11 (responsible for standardizing Fibre Channel) at http://www.t11.org.

The iNCITS Secretariat is administered by the staff of the Information Technology Industry Council (http://www.itic.org), located at:

Information Technology Industry Council
1250 Eye Street NW
Suite 200
Washington, DC 20005
USA
+1 202 737 8888 (voice)
+1 202 638 4922 (fax)

National Institute of Standards and Technology (NIST)

NIST is not, strictly speaking, an accredited standards body, but a government agency. However, under Section 513 of the Information Technology Management Reform Act of 1996 and the Computer Security Act of 1987, Public Law 104-106, NIST develops standards, guidelines, and associated methods and techniques for federal computer systems. The standards are known as Federal Information Processing Standards (FIPS).

The main NIST Web site is at: http://www.nsit.gov. See http://www.itl.nist.gov/fipspubs/ to obtain copies of FIPS, and http://www.itl.nist.gov/fipspubs/geninfo.htm for general information on the publications.

Inquiries about FIPS should be addressed to:

Public Inquiries Unit
NIST, 100 Bureau Drive
Stop 3460
Gaithersburg, MD 20899-3460
USA
Email: inquiries@nist.gov
+1 301 975 6478 (voice)
+1 301 975 8295 (fax)

Appendix B: Recommended Bibliography

The following is an annotated bibliography of information that the authors of this booklet, and other members of the SNIA Security Technical Working Group, have found useful in learning about security technologies and techniques. The bibliography is divided into two sections, one of which deals with publications and the other with online information sources. All of the items in the second section have been determined to be operable at the time of publication, but obviously no guarantee can be made of their continued availability. An up-to-date version of this list can be obtained from the SNIA Web site at http://www.snia.org.

Publications

For an introduction to the history of cryptography, and an in-depth treatment of the analysis of monoalphabetic substitution ciphers and the methods by which the German Enigma codes were broken, see:

Singh, Simon, *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography,* Anchor Books, August 2000 (ISBN 0-385495-32-3).

Kahn, David, *The Codebreakers: The Story of Secret Writing,* Revised Edition, Scribner, December 1996 (ISBN 0-684831-30-9).

For specific information on the Enigma and the people involved in breaking the code, see:

Hodges, Andrew, *Alan Turing–The Enigma,* Walker & Co; October 2000, (ISBN 0-80277-58-0)

Sebag-Montefiore, Hugh, *Enigma: The Battle for the Code,* John Wiley & Sons; January 2001, (ISBN: 0-471407-38-0)

Churchhouse, Robert, *Codes and Ciphers: Julius Caesar, the ENIGMA, and the Internet,* Cambridge University Press, March 2002, (ISBN 0-521008-90-5)

For fictional matter addressing the Enigma and breaking its code, see:

Harris, Robert, *Enigma,* Random House Publishers, August 1996 (ISBN 0-804115-48-6). (A film of the book was released in 2001 by Intermedia Film Equities, directed by Michael Apted and starring Dougray Scott & Kate Winslet.)

Whitemore, Hugh, *Breaking the Code,* a play performed in both London & New York, filmed in late 1995, as a production of THE DRAMA HOUSE and WGBH BOSTON for BBC NORTH (directed by Herbert Wise), shown in the U.S. in the Masterpiece Theatre series on February 2, 1997, and available on video starring Derek Jacobi (of *I, Claudius, Brother Cadfael,* etc. . . .)

For a general background on the development of computer system security requirements, NSA requirements, and an introduction to legal responsibilities, see:

Gangemi, Sr., G. T. & Russell, Deborah, *Computer Security Basics,* O'Reilly, July 1992 (ISBN 0-937175-71-4).

For a description of the human element in security, and a large number of stories about past security problems, their impact, and methods of avoiding such pitfalls, see:

Schneier, Bruce, *Secrets & Lies,* Wiley, August 2000 (ISBN 0-471253-11-1).

For algorithms and code details, see:

Dr. Dobbs' *Essential Books on Cryptography & Security,* consisting of 12 important texts on one CD-ROM. Texts includes Bruce Schneier's *Applied Cryptography* and *The Handbook of Applied Cryptography* (Menzes et al). All files are in PDF, with indexes and a complete search capability. See http://www.ddj.com for information.

Web Sites

The SANs Institute Reading Room has a wealth of information and tools related to network security, and it can be found at http://www.sans.org.

The lectures and slides for an excellent course on Cryptography and Computer Security, created by the University of New South Wales (the Australian Defense Force Academy), can be found at http://www.cs.adfa.oz.au/teaching/studinfo/csc.

The European Parliament report on Echelon, an interception system rumored to have a global reach, can be found at http://www.europarl.eu.int/committees/echelon_home.htm.

For a practical example of security in the business world, see the Visa Cardholder Information Security Program at http://usa.visa.com/business/merchants/cisp_index.html.

For more information on issues related to the Wired Equivalent Privacy scheme used in some wireless LANs, see the information at http://www.isaac.cs.berkeley.edu/wep-faq.html http://www.cs.rice.edu/~astubble/wep/ and http://www.infoworld.com/articles/op/xml/01/06/25/010625opsecurity.xml.

The tool for recovering keys is at http://freshmeat.net/projects/airsnort.

For information on the Enigma and other World War II cryptography developments, see the superb Web site created by Tony Sale, the founder and curator of the Bletchley Park Museum, at http://www.codesandciphers.org.uk/

The following site gives the history of solving the Enigma cipher, focusing on the pre-war Polish efforts (the site is also available in Polish): http://home.us.net/~encore/Enigma/enigma.html

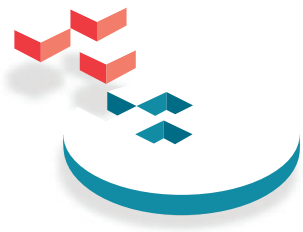
An Enigma simulator can also be downloaded from http://www.xat.nl/enigma/

Information on the building of an enigma replicator can be found at http://www.enigma-replica.com/

About the SNIA

The Storage Networking Industry Association (**SNIA**) is a not-for-profit organization, made up of over 300 companies and individuals worldwide spanning the entire storage industry. SNIA members share a common goal: to set the pace of the industry by ensuring that storage networks become efficient, complete, and trusted solutions across the IT community. To this end, the SNIA is uniquely committed to delivering standards, education, and services that will propel open storage networking solutions into the broader market.

For information, contact the SNIA at 650-949-6720 or via e-mail at **executivedirector@snia.org**, or visit the SNIA Web site at **http://www.snia.org**.



SNIA
Storage Networking Industry Association