



realtimepublishers.comtm

The Definitive Guidetm To

Information Theft Prevention

Blue  **Coat[®]**

Dan Sullivan

Chapter 4: Protecting Information during Transmission	66
Encryption: Preserving Information Confidentiality and Integrity During Transmission.....	67
Symmetric Key Cryptography	69
Asymmetric Key Cryptography	71
Computational Demands.....	71
Mathematical Attacks	71
Providing Cryptography Services with PKI	73
CAs	74
RAs	76
Certificate Repository and Key Revocation	76
Key Backup and Recovery System.....	76
Client Applications	76
Authenticating Users, Programs, and Devices.....	78
Authenticating Users.....	78
Authenticating Servers.....	79
VPNs.....	80
Network Tunneling.....	80
Challenges of Tunneling Protocols.....	82
SSL VPNs	83
SSL Sessions.....	84
SSL VPN Gateways.....	84
Benefits of VPNs	84
Limits of VPNs for Preventing Information Theft	85
Summary	86

Copyright Statement

© 2006 Realtimepublishers.com, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtimepublishers.com, Inc. (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtimepublishers.com, Inc or its web site sponsors. In no event shall Realtimepublishers.com, Inc. or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtimepublishers.com and the Realtimepublishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtimepublishers.com, please contact us via e-mail at info@realtimepublishers.com.

[**Editor's Note:** This eBook was downloaded from Realtime Nexus—The Digital Library. All leading technology guides from Realtimepublishers can be found at <http://nexus.realtimepublishers.com>.]

Chapter 4: Protecting Information during Transmission

Data can be well protected within a controlled infrastructure but is especially vulnerable to theft or tampering when it is transmitted. Consider what a thief would have to do to retrieve data stored on a secure server: infiltrate the network boundary protected by a firewall, authenticate to an access control system with a user identity authorized to access the information, gain access to an application—such as a database—that would allow the thief to find the information, transmit the data back to the thief's storage device, and finally avoid detection by tampering with highly secured audit logs that record details of such information access transactions. When that same data is transmitted outside the secure network, the thief's job gets easier.

Once information leaves the protected network, it travels through publicly accessible systems, such as the Internet, or mediums, such as the public airways. On the Internet, basic attacks, such as domain spoofing, can redirect traffic intended for a legitimate site to a bogus site controlled by information thieves. When the public airways are used to transmit data, anyone with easily purchased equipment can detect transmissions. (How many of us have detected a neighbor's wireless home network when scanning for our own?) As Figure 4.1 shows, outside the secure network, data may be stolen, redirected, or otherwise tampered with.

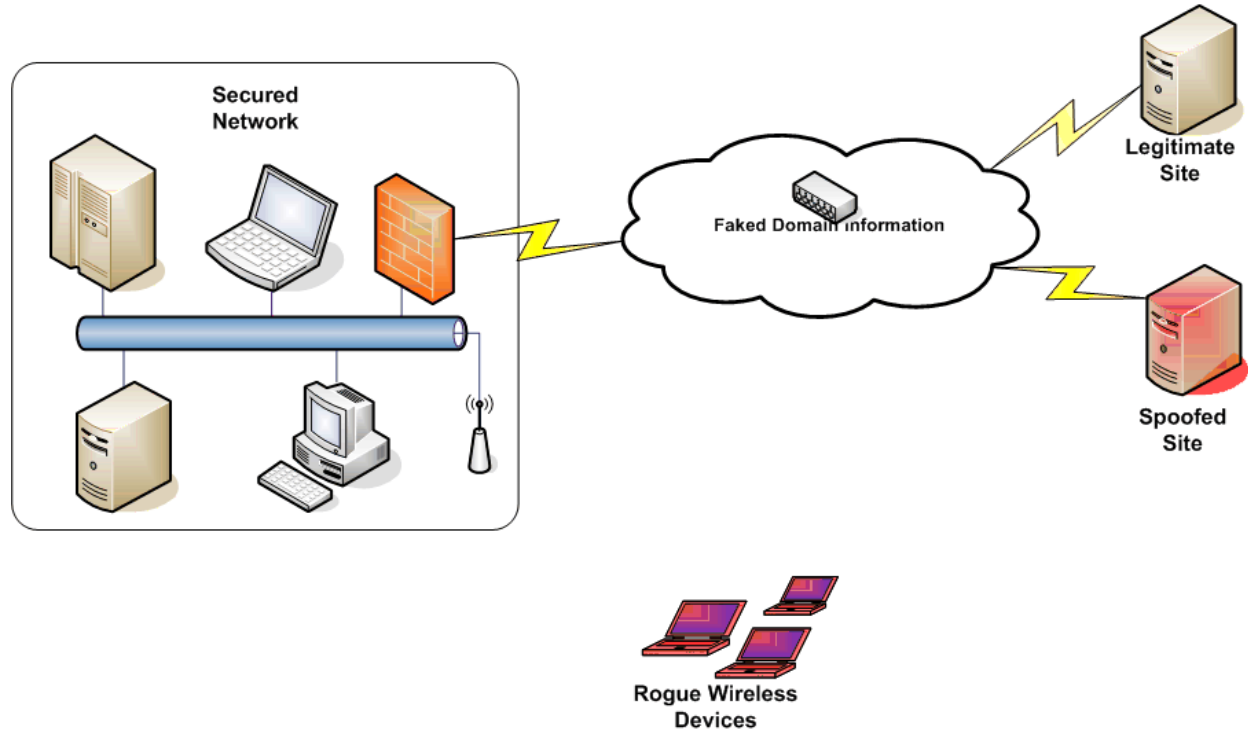


Figure 4.1: Once data leaves a secured network, it is vulnerable to multiple means of attack.

Within a secured network, data is protected by its environment. Identity management and access control systems, antivirus systems, intrusion prevention applications, and a host of other security measures all contribute to maintaining information integrity and confidentiality. When transmitted beyond the protective measures of the network, other measures must be employed. Three basic approaches have been used to address the need to protect information when it must be transmitted:


- Encrypting data
- Authenticating users, programs, and devices that receive information
- Extending the protection of a secure network through the use of virtual private networks (VPNs)

Encryption is a fundamental technology that can be used by itself or as an element of the other measures. Authenticating user programs and devices is a widely employed measure to ensure secure sessions between short-term users and service providers, for example, in e-commerce transactions. VPNs are used when mobile and off-site users require remote access to an organization's network, as is the case with sales staff constantly on the road.

This chapter will examine the technologies used to secure data during transmission along with management issues that arise when these technologies are deployed. The discussion begins with encryption, a technology essential to protecting the confidentiality of information during transmission.

Encryption: Preserving Information Confidentiality and Integrity During Transmission

Today's best encryption algorithms use mathematical techniques to transform data into difficult-to-break scrambled messages.

 For a brief introduction to cryptography, see Chapter 3.

Before delving into the details of how encryption works, it is useful to define several terms:

- Plaintext is the data to be encoded. This can be an email message that is easily read by humans or streams of binary data that need to be kept secret. Although the term uses the word text, it actually applies to any data prior to the encryption process.
- Ciphertext is data after it has been encoded. The better the encryption method, the more difficult it is to derive the plaintext of a message given just the ciphertext.
- Algorithms are sequences of instructions that calculate, transform, and otherwise manipulate data.

- Keys are sequences of bits that are used by the encryption process to encode plaintext. In general, the longer the key, the more time that is required to encode a message, and the more time required to decrypt it. Fortunately, longer keys also make messages more difficult to crack.
- Workload is the effort required to decrypt an encrypted message without a key. The stronger the encryption technique, the greater the workload.
- Strength of an encryption method is a comparative measure based on the algorithm used to encrypt a message and the length of the key. Even with well-designed algorithms and long keys, if the keys are not kept secret, messages can be decrypted by unauthorized persons.

These terms are essential for understanding modern encryption techniques. For example, the plaintext

```
This is a sample file to be encrypted.
```

is transformed into the ciphertext

```
ABICODER;ÔBØ_x]•_
```

using the 448-bit key

```
sfkj9aslk83p3r1aash8873561ndhdhdydydydydyqgbsbteravgfva
```

with the Blowfish algorithm.



The ABI-Coder application from Abi Software Development (<http://www.abisoft.net>) was used to encrypt this example message.

Most cryptography today is based on secret keys used with publicly known algorithms. At first glance, it might appear that using secret algorithms would provide greater protection than using publicly available algorithms, but that is rarely the case. Unless one has broad and deep expertise in cryptography, the chances of developing a secret algorithm that is as effective as the best publicly available algorithms are slim. This principal has been stated and expanded on since it was first stated in the 19th century and has come to be known as Kerckhoff's Law.



For more information about Kerckhoff's Law and modern corollaries, see http://en.wikipedia.org/wiki/Kerchoff's_law.

The basic principal behind Kerckhoff's Law is that even if someone other than the intended recipient knows how a message was encrypted, that person should not be able to decrypt the message without the key. This idea has a significant impact on the usefulness of a cryptographic system—only the key must be kept secret.

As Figure 4.2 shows, there are several components to the encryption process; as long as the plaintext and the decryption key are kept secret, the message remains confidential. (In the case of symmetric key cryptography, the same key is used for encryption and decryption; in that case, the encryption key copy must be kept secret as well).

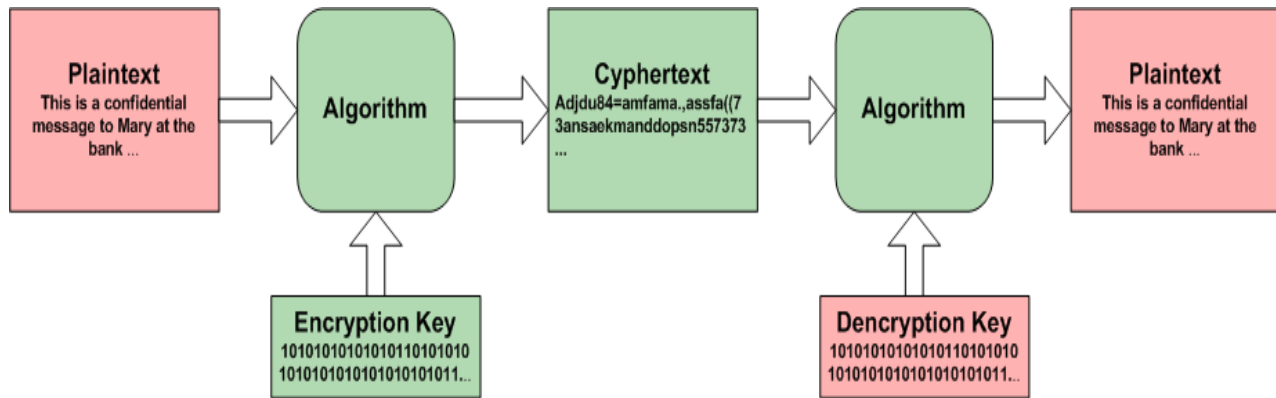


Figure 4.2: The encryption process requires a message, an algorithm, and a key to function; only the decryption key and the plaintext need to be kept secret.

Once a message is decrypted after arriving at its destination, it must be protected by means other than encryption. If an encrypted message is sent to an untrusted device, such as a shared public computer, the message is vulnerable to several methods of attack, including spyware, browser cache readers, and video frame grabbers. Additional security measures are required to prevent information theft at that point. Chapter 5 will provide details.

Algorithms that use the same key for encryption and decryption are known as symmetric key algorithms; those that use different keys are known as asymmetric algorithms. These two classes of algorithms have different properties that make them more or less suitable for particular uses.

Symmetric Key Cryptography

As the name implies, symmetric key cryptography uses one key for both the encryption and decryption operations. This immediately presents two problems.

Challenge 1: Key Exchange

First, both the sender and receiver of an encrypted message must have a copy of the key. It does Bob no good if Alice sends him an encrypted message and he does not have the key to decrypt the message. The question arises, how will Alice send Bob the key?

She cannot send it by the same method she wants to send the encrypted message because the key could be intercepted and used by someone else to decrypt messages between Alice and Bob. If the transmission method were secure enough to send the key, Alice and Bob would not need to encrypt their messages. Alice needs some other method, known as an out-of-band method, to get Bob the message. She could copy it to a disk or flash drive and hand it to Bob, she could be write it down and mail it (neither practical nor secure), or Alice could use a secure key exchange mechanism such as the Diffie-Hellman Key Exchange method which is described later. Even when the key exchange problem has been addressed, there remains another significant problem: key proliferation.

Challenge 2: Key Proliferation

The second problem that must be addressed when implementing symmetric key cryptography is key proliferation. Consider a simple case of five individuals who want to communicate confidentially with each other; no one, including others in the group of five, should be able to decipher a message sent except the sender and recipient. Figure 4.3 shows the problem graphically.

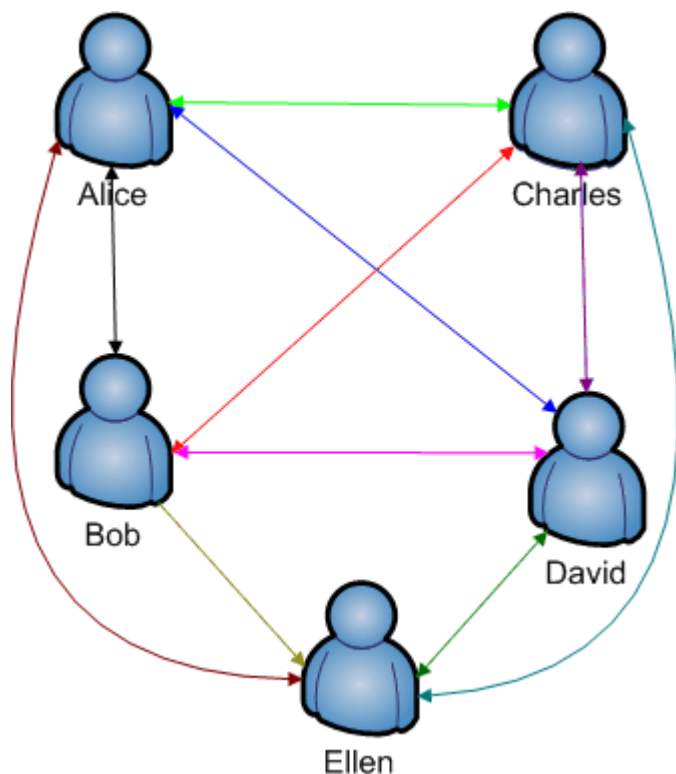



Figure 4.3: The demand for symmetric keys increases rapidly as the number of users increases.

When only two people need to exchange encrypted messages, one key is needed. When three people need to exchange messages in pairs, three keys are required; for four persons, six keys are required; for five users, 10 keys are required. The problem grows so rapidly, that in a group of 100 people, 4950 different keys would have to be used.

 The general formula for calculating the number of symmetric keys required for confidential communication in a group with N members is $N(N-1) / 2$.

The key exchange and the key proliferation problem are significant, but there are cases in which they are worth overcoming to realize the advantages of symmetric key cryptography. Perhaps the most significant is speed. The alternative, asymmetric key cryptography, is an order of magnitude slower than symmetric key methods.

Asymmetric Key Cryptography


In asymmetric key cryptography, also known as public key cryptography, two keys are used for encryption. One is made public and one kept private. The public key is used by message senders to encrypt a message. The private key is the only one that can decrypt the message so as long as that key is kept confidential, no one should be able to read the encrypted message. Public key cryptography solves the problem of key proliferation found in symmetric key cryptography. It also eliminates the key exchange problem; now only one key (the public key) needs to be shared and it does not matter who knows this key because it is used only to encrypt, and not decrypt, messages. Needless to say, as with any technologies, there are limitations and drawbacks.

Computational Demands

First, encrypting with asymmetric keys takes much longer than encrypting with symmetric keys, up to an order of magnitude longer. This shortcoming stems from the types of calculations required. The calculations are based on mathematical operations over large numbers, such as multiplying large prime numbers. The time required can be lessened if shorter keys are used, but shorter keys might compromise the encryption strength, depending on the algorithm.

The same key length used in different algorithms can provide different levels of cryptographic strength. For example, elliptic curve algorithms can provide the same level of cryptographic strength with short keys than is required for other algorithms.

The mathematical problems underlying asymmetric cryptography are computationally intensive in one direction but going in reverse is even much more difficult. For example, multiplying two large numbers will take some time, but given the product of those two large numbers and trying to find two prime factors is even more time intensive. It is this asymmetric property (relatively easy calculation in one direction, much more lengthy computation in the other direction) that makes these problems so useful for cryptography.

 The mathematics of asymmetric key cryptography are beyond the scope of this chapter. For the interested reader, see references on discrete logarithms and discrete exponentiation as well as elliptic curves. For an excellent introduction to elliptic curve cryptography, see http://www.certicom.com/index.php?action=ecc_tutorial_home. For more information about the Diffie-Hellman protocol, see Mauer and Wolf's "Diffie-Hellman Protocol" at <http://portal.acm.org/citation.cfm?id=343484&dl=GUIDE&coll=GUIDE>.

Mathematical Attacks

The public and private keys used in asymmetric cryptography are mathematically related. If one key is known, it is theoretically possible to calculate the other. This opens the possibility of two types of attacks.

Brute Force Attacks

The simpler of the two attacks is brute force calculations—for example, finding the prime factors of a large number can take quite a bit of time but it can be done. As the computational power of computers increases and the ability to harness large numbers of computers to focus on a single problem improves, the ability to carry out brute force attacks will increase. This threat is countered in two ways. Encryption users can increase the length of keys to improve the strength of the encryption. Users can also realize that confidential information may be useful for only a short period of time. For example, information about a pending merger may not to be kept confidential after the merger is made public; similarly, bank account and credit card information is no longer useful to a thief after the accounts have been closed. As long as the information remains encrypted throughout the useful lifetime of the information, that information has a sufficient level of protection.

Quantum Computing: Cryptography's Greatest Threat

A long-term threat to current encryption techniques is the possibility of quantum computing. Quantum computers use aspects of quantum mechanics, such as the superimposition of states. This is still a highly theoretical area of research and there are substantial technical problems yet to overcome before quantum computing becomes a reality; however, if it does, cryptography will have to change radically to remain useful.

As one researcher noted, "Virtually all encryption methods used for highly sensitive data are vulnerable to one quantum algorithm or another" (Source: Graham P. Collins, "Computing with Quantum Knots" at <http://www.sciam.com/article.cfm?chanID=sa006&collID=1&articleID=000552EC-2265-1417-A26583414B7FFE9F>).

For more information about quantum computing, see Barenoc, et. al. "A Short Introduction to Quantum Computing" at <http://www.qubit.org/library/intros/comp/comp.html>, and for an example of quantum computing's foray into cryptanalysis, see "IBM's Test Tube Quantum Computer Makes History" at http://domino.watson.ibm.com/comm/pr.nsf/pages/news.20011219_quantum.html.

Mathematical Analysis

Brute force attacks depend on computational prowess. In the case of finding a pair of prime factors of a number, it is a matter of trying many combinations of numbers. The simplest approach is to just try all possible combinations of pairs of numbers up to some point. Blind searching like this could go on, theoretically at least, forever. However, mathematicians have found useful properties of numbers and problems used in cryptography that can be exploited to reduce the number of possible solutions to a cryptanalysis problem.

For example, encrypted messages that depend on factoring a large number into a pair of prime numbers could take advantage of the fact that as numbers increase, the frequency with which prime numbers occur decreases. Thus, if a prime number is found, it is not likely that numbers near N will also be prime, so nearby numbers can be skipped. This is an overly simple example but it shows that mathematical properties of numbers, elliptic curves, and other mathematical objects can be used to improve brute force techniques.

Encryption is a fundamental technology—to use it effectively and efficiently, systems must be in place to provide cryptographic services. One model of such a collection of systems is known as public key infrastructure (PKI).

Providing Cryptography Services with PKI

PKI is a framework consisting of protocols, applications, procedures, and related security mechanisms that create an environment of trust among entities that otherwise would not have established trust relationships. The environment of trust is created by establishing several characteristics important to trusted communications:

- Authentication of communicating parties
- Confidentiality of information exchanged between parties
- Non-repudiation of messages sent by a party
- Integrity of messages sent

These properties guarantee parties to a communication are assured:

- The parties in communication are who they say they are
- No one other than the intended recipient of a message will be able to read the message
- When someone sends a message, they cannot later deny sending the message
- Messages cannot be tampered with in transit with some indication of a change

The asymmetric encryption techniques described in the previous section play a central role in PKI but are by no means the only part. A PKI includes:

- A Certificate Authority (CA)
- Registration Authority (RA)
- Certificate repository
- Certificate revocation system
- Key backup and recovery system
- Client software

Together, these elements provide the core services that implement the secure communications environment.

CAs

CAs are third parties trusted by parties that want to communicate securely. CAs vouch for the identity of the parties and provide an electronic certificate attesting to the identity of the person or organization possessing a certificate.

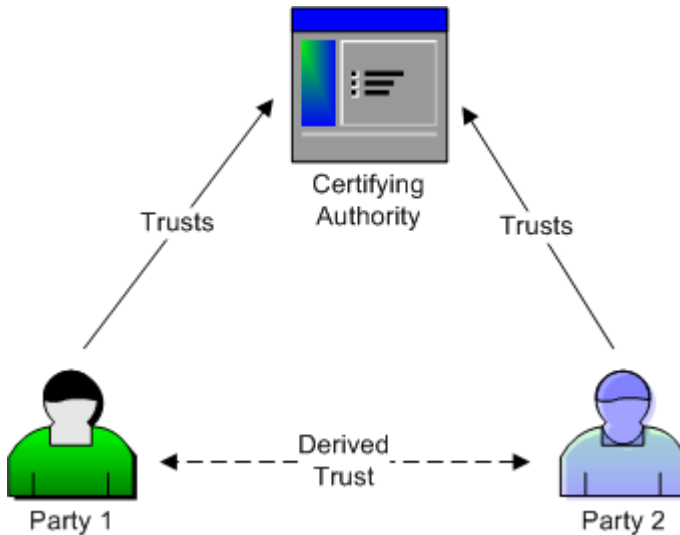


Figure 4.4: Trusts between parties in a PKI environment are derived from their trust of the certifying authority that vouches for the identity of parties.

A certifying authority creates and manages digital certificates, which contain identifying information about parties in the PKI environment. The digital certificate essentially states that the CA certifies that the party possessing this certificate is in fact the party that it claims to be. A CA can either be the organization that makes use of the certificates or it can be a trusted third party.

Large organizations might want to create and manage their own certificates when they are used primarily for internal purposes; for example, authenticating servers and users within the corporate wide area network (WAN). In other cases, it is more appropriate to have a third party issue certificates. For example, when an e-commerce site wants to assure potential customers that their site is trustworthy while minimizing the chances of hackers hijacking their Web site, the business would apply for a certificate from a third party. Some well-known CAs include:

- Entrust
- Equifax Secure
- GTE Corporation
- RSA Security
- VeriSign

After the identity of an applicant has been verified, the CA will issue a digital certificate. The certificate itself must be tamper-proof to ensure that no one can change the information in that certificate once it is issued. For manageability reasons, certificates will have limited lifetimes. Figure 4.5, shows an example certificate for a VeriSign server.



Figure 4.5: A typical digital certificate for a server.

Certificates must interoperate with multiple applications, OSs and security protocols, so they are standardized to all contain the same types of information. The key attributes:

- Version of the certificate
- Unique serial number of certificate
- Algorithm used to calculate digital signature of the certificate
- Name of the CA issuing the certificate
- Start and end dates of the certificate
- Name of the subject or owner of the certificate
- Public key of the owner
- Unique identifier of the CA issuing the certificate
- Unique identifier of the owner of the certificate

In addition, optional extensions may be included.

RAs

CAs typically work with RAs to perform the initial identity verification tasks. For example, if an individual wanted to acquire a digital certificate, he or she would contact an RA who would verify his or her identity. The level of effort going into verifying an identity will vary with the purpose of a certificate. For example, an individual applying for a certificate for personal use will have less documentation required of them than a bank applying for a certificate to assign to a server for online banking. Of course, the cost of acquiring a certificate will vary as well.

The RA works with the CA after identity is established to create a certificate for the applying party. After the certificate is issued, the RA may also provide other basic services, such as issuing new certificates for expired ones.

Certificate Repository and Key Revocation

A certificate repository is a database of certificates and associated management information. In addition to certificates, the repository can contain revocation lists, also known as certificate revocation lists (CRLs), which are lists of certificates that have been revoked even though the certificate may still contain valid dates. For example, if a new certificate is issued to a person because the person's private key has been compromised, the certificate with the public key associated with the compromised private key would be revoked.

Key Backup and Recovery System

As the title implies, the function of the key backup and recovery system is to allow users to recover keys that have been lost or damaged. It also provides administrators with an application-specific method to back up key information for recovery and continuity operations. For example, if a business is disrupted due to natural disaster in one site, operations can continue from a backup site, assuming all data, including PKI key and related data, can be replicated to the backup site.

Client Applications

Client applications are the applications that make use of digital certificates. Applications can use digital certificates in custom applications or can use protocols, such as Secure Sockets Layer (SSL) that make use of certificates.

When a Web application user accesses a secure site using SSL, a digital certificate is sent from the server to the browser. Rather than have users review every certificate as it is downloaded, browsers are typically configured with a list of trusted CAs. Figure 4.6 shows a partial list of the CAs included in the Firefox browser.

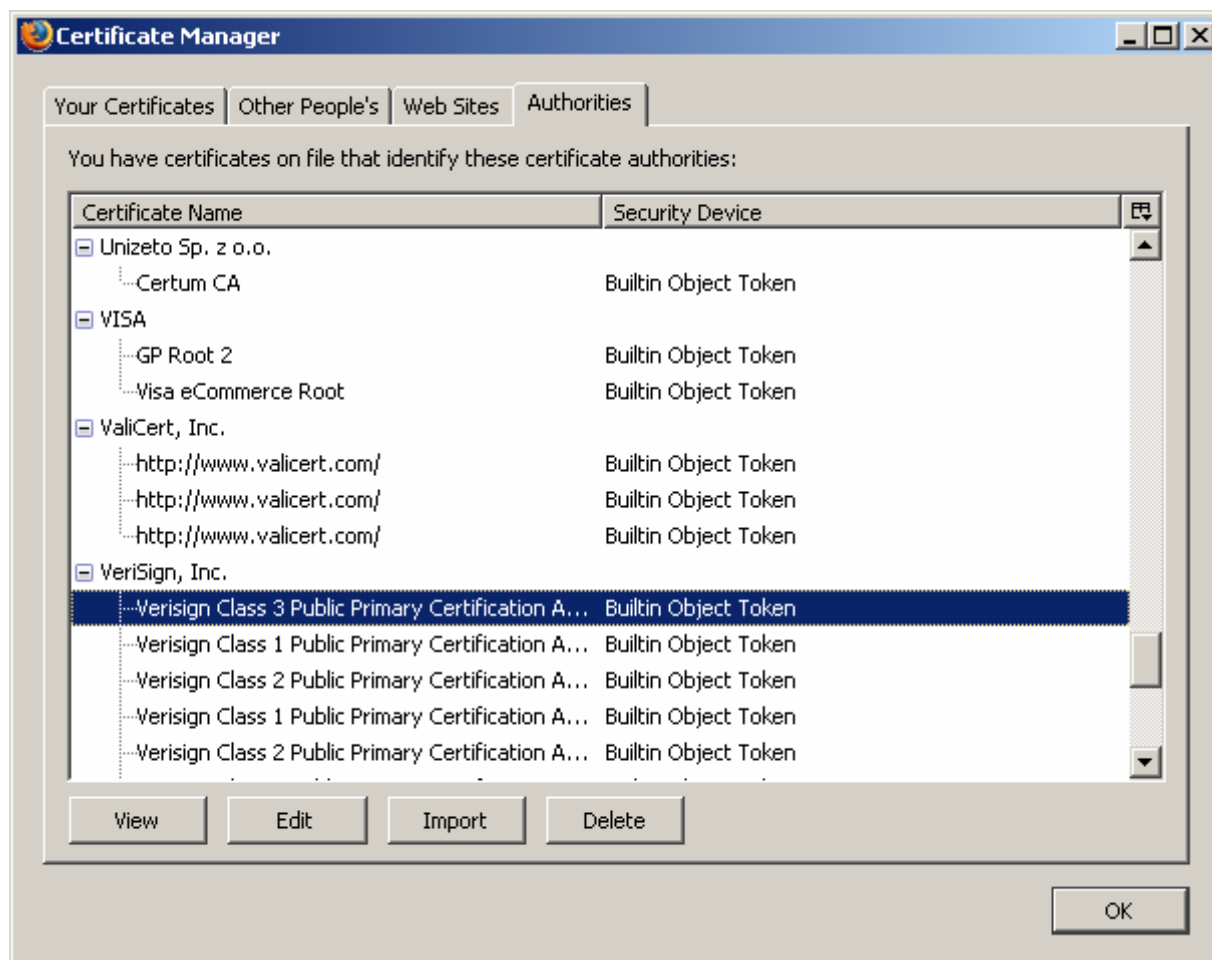


Figure 4.6: Client applications, such as Web browsers that make use of certificates to authenticate SSL sessions, manage certificates to ease the task on end users.

PKI is used to implement cryptographic services in an organization. Planning a PKI is large undertaking and deserves a lengthy discussion beyond the scope of this chapter. However, there are a few high-level points that should be considered when deploying a PKI (see Figure 4.7):

- Determine the number of CAs that will be needed based on the number of certificates issued, the frequency of issuing, and the geographical distribution of users.
- Determine the trust model for outside parties; for example, will business partners be included? Will their CAs be trusted?
- Will users and servers be enrolled automatically?
- Will templates for a defined set of roles be used?
- Will the CA integrate with a Lightweight Directory Access Protocol (LDAP) directory?
- What is the approval process for certificate requests?
- What policies will be enforced with respect to certificates?
- How will applications use certificates?
- Will certificates be used outside the organization's network?

For introductory material on PKIs, see “PKI Basics—A Technical Perspective” at http://www.pkiforum.org/pdfs/PKI_Basics-A_technical_perspective.pdf, and for general PKI information, see the PKI Forum at <http://www.pkiforum.org/>.

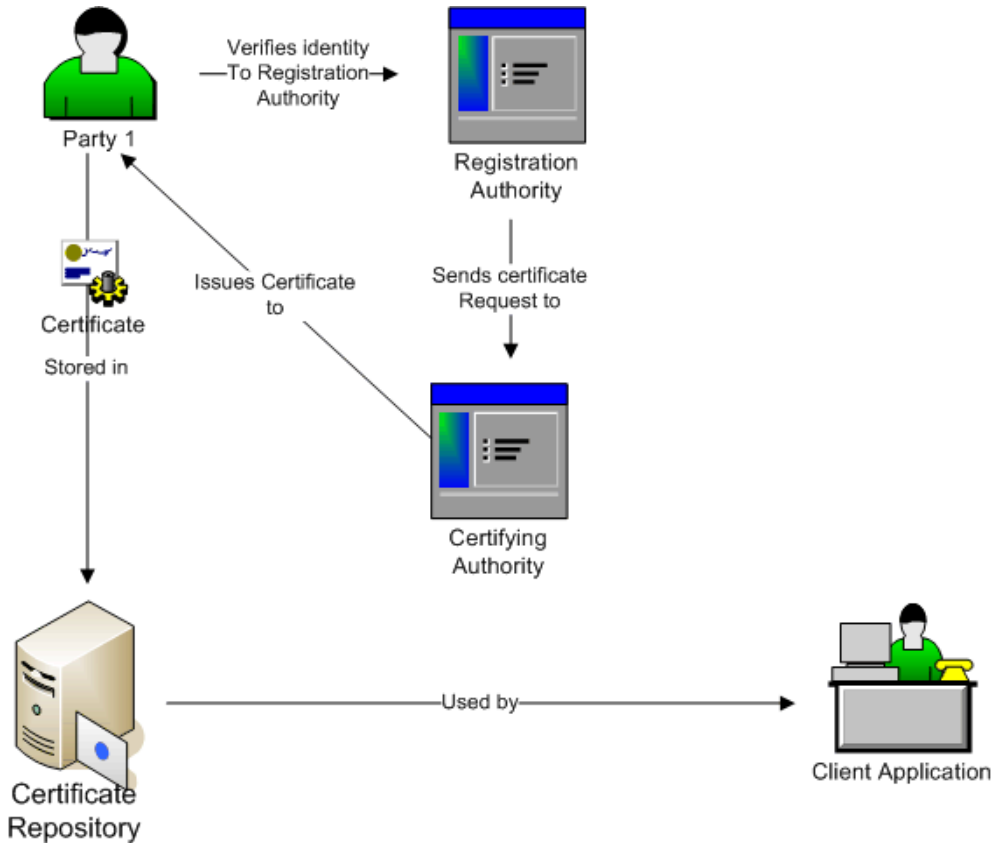


Figure 4.7: Components and information flow of a PKI.

Authenticating Users, Programs, and Devices


Certificates are a fundamental method for authenticating users, programs, and devices. We are all familiar with authenticating to an OS by providing a username and password, but there are other types of authentication.

Authenticating Users

In general, one can authenticate using:

- Something one knows, such as a username and password
- Something one is, that is, by biometric identification such as iris scan or fingerprint scan
- Something one has, such as a smart card or a digital certificate

Each of these authentication methods has its pros and cons. Usernames and passwords are easy to implement and are inexpensive; their chief disadvantages are that they are easy to crack, passwords are often forgotten, and may be written down and therefore easily compromised.

 If you are concerned about weak passwords on your server, check them with password crackers, which are freely available on the Web. Like other vulnerability scanning tools, these can be used by systems administrators as well as hackers; as always, *caveat emptor*. For more information about password security and links to resources, see Passwordportal.net.

Biometric devices are more secure than passwords. Users do not have to remember multiple passwords, they do not have to change passwords, and they cannot lose the biometric information. The disadvantages include cost and user resistance to biometric scanning devices.

Authenticating with something one has, especially digital certificates, can improve upon the security found with just passwords without some of the inconveniences of biometric devices. Digital certificates also have the advantage of working with devices, such as servers, as well as with human users.

Users can authenticate to applications using digital certificates, for example, using a PKI. Users would typically keep a certificate in a secure database or cache and provide it to the application on demand. Of course, the application would have to be designed to prompt for, receive, and accept the information in the certificate.

Authenticating Servers

A common use of digital certificates is to authenticate servers. Authentication is especially important when conducting transactions over the Web. A shopper, for example, will want to be sure he is communicating with a legitimate server and not a bogus look-a-like. SSL is commonly used for this. The SSL protocol works between a client and a server as follows:

1. A Web user browses to a secure Web page on a site.
2. The Web server sends a message to the client indicating a secure session is needed to continue.
3. The client sends information about the type of security protocols it supports.
4. The server finds a matching protocol that it also supports.
5. The server sends its digital certificate to the client.
6. The client either accepts the certificate, that is, it decides to trust the certifying authority, and the session continues or the client rejects the certificate and the session terminates.

Usually, accepting the certificate is done automatically if the CA of the certificate is listed as a trusted authority within the browser. Occasionally, a certificate will be expired and the user will be prompted to accept the certificate just for this session, accept it permanently, or reject the certificate. When the user is prompted in these cases, the user also has the chance to review the certificate before making a decision about how to proceed.

In addition to using SSL for single sessions between clients and Web sites, SSL has been adopted to extend the reach of corporate networks through the use of VPNs.

VPNs

With an increasingly mobile and geographically dispersed workforce, the need for secure remote access to corporate networks is increasing. Regardless of whether it is a single “road warrior” dialing in from a hotel room or a satellite office with a need for near constant access to the corporate headquarters LAN, remote access must be secure. VPNs are the family of technologies that are used to provide remote and secure access to the network infrastructure.

Figure 4.8 depicts a simple example. Remote sites or users connect through the public Internet but use secure connections to preserve the confidentiality and integrity of communications.

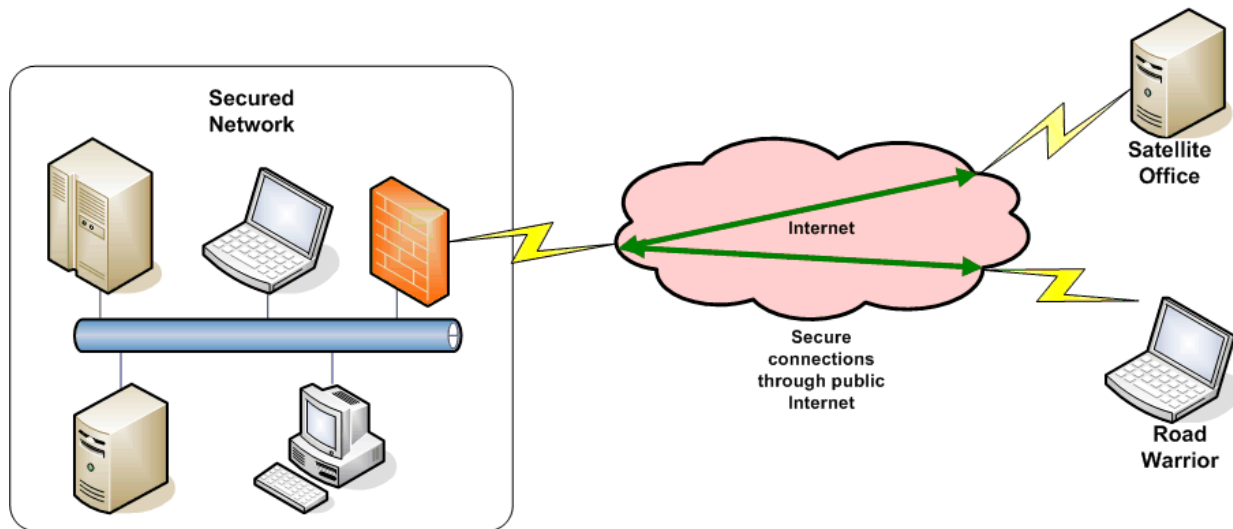


Figure 4.8: VPNs use secure connections over the public Internet.

VPNs are built on two underlying technologies: encryption and tunneling. Encryption has been discussed at length; tunneling will be addressed in the following section.

Network Tunneling

Tunneling is the process of encapsulating network data that is formatted for one network protocol into another protocol, transmitting the encapsulated data, then stripping off the encapsulating protocol data when the packet arrives at the destination. There are many uses for tunneling.

Consider two networks that need to communicate over the Internet. Both networks use NetBEUI and IPX protocols; the Internet, however, is based on the Internet Protocol (IP). To transport NetBEUI and IPX packets, they are first packaged into IP packets and then routed to their destinations. At the destination site, tunneling software accepts the packets, strips off the IP overhead information, and transmits the encapsulated packet in its native format.

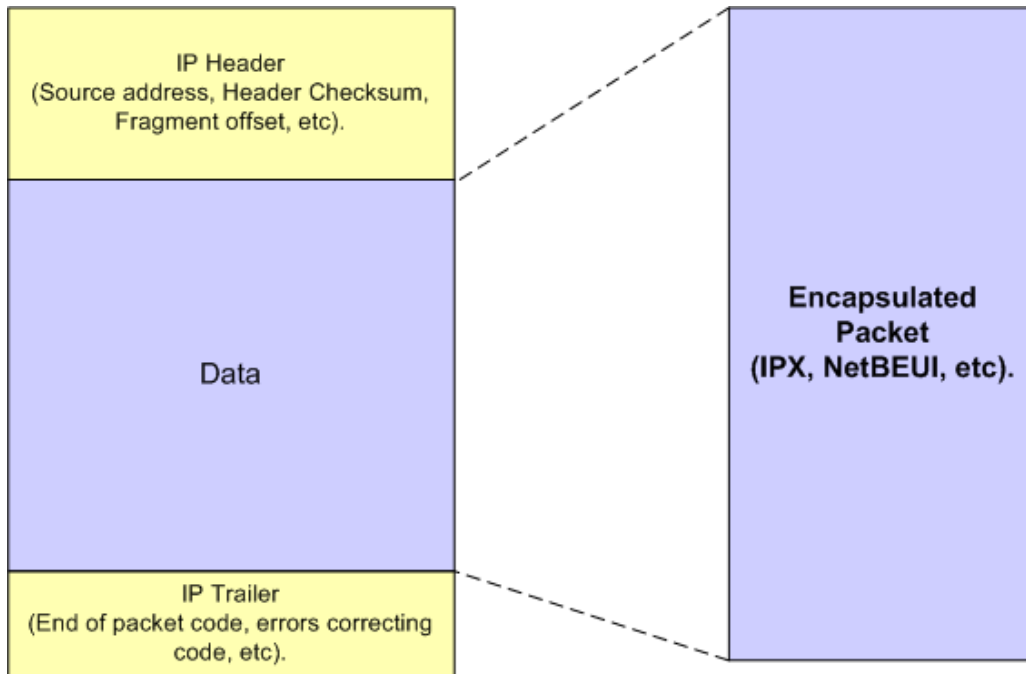


Figure 4.9: Data in one protocol can be encapsulated as the data or payload of another protocol's packet.

A number of protocols have been used to implement tunneling including:

- Serial Line Internet Protocol (SLIP) an older protocol designed for use over serial lines
- Point-to-Point Protocol (PPP) a newer protocol for serial line connections with more features than SLIP, including compression, error correction, and multiple authentication protocols
- Point-to-Point Tunneling Protocol (PPTP) a Microsoft protocol for VPNs
- L2TP combines PPTP with Layer 2 Forwarding (L2F) a Cisco protocol that supports point-to-point communication over multiple protocols, not just IP, and provides enhanced security through IPsec
- IPsec, a framework for secure communication that provides flexibility in setting up secure connections by providing for a variety of algorithms and authentication mechanisms.

These protocols have been used to implement VPNs but there are several drawbacks.

Challenges of Tunneling Protocols

Tunneling protocols offer a great deal of flexibility with respect to using multiple protocols across geographically distributed networks, but there are challenges to implementing them. Some vendors have developed proprietary implementations of VPNs that require client applications installed on all users' devices. This development resulted in the all-too-common support overhead needs to configure and maintain the proprietary applications.

In other cases, the complexity of setting up and managing the framework, IPsec in particular, deterred some would-be adopters. IPsec can be configured to different needs, but that flexibility also leads to additional planning and administration.

The IPsec framework uses two security protocols, the Authentication Header (AH) protocol and the Encapsulating Security Protocol (ESP). AH is used for authenticating and ESP is used for both authentication and encryption.

In addition to two protocols, IPsec supports two modes of operation. In transport mode, the data or payload is encrypted; in tunnel mode, the payload as well as the header is encrypted.


Because IPsec is so flexible, there are a number of parameters that must be configured when a client and a server engage in an IPsec session. These are stored in a security association (SA). The SA includes:

- Lifetime of the SA
- IPsec mode, which is either tunnel or transport mode
- Encryption algorithm used
- Encryption key
- ESP authentication key
- AH authentication key


Because one SA is needed for inbound communications and one for outbound, every session requires two SAs. With multiple IPsec sessions running at any time, there is a need for an additional data structure, the Security Parameter Index (SPI) to keep track of them.

In addition to configuration and management challenges, the potential to open network resources to remote users on a continuous basis is an advantage to some and a security risk to others. Within an IPsec VPN, remote users are logically part of the network. Services available to users physically connected to the corporate network are also available to remote users. This setup is ideal for some applications; for example, if a remote user needs access to file servers, databases, and legacy applications. The flip-side is that internal file servers, databases, and legacy applications are now potentially vulnerable to outsiders.

Some systems administrators and security managers would cringe at the thought of remote users on unmanaged devices with full access to the corporate network. For example, some companies might want to restrict access to a remote user using a hotel business center computer or other public device to a small set of applications, such as email and calendar programs. This type of fine-grained access control requires additional policies to be defined and enforced based on users' locations, the type of application they are accessing, the time of day, and other parameters.

 This security concern is a problem common to remote access, not just to particular frameworks or protocols used for VPNs. It has led to the development of new security measures, known collectively as security on demand. Chapter 5 will provide more details.

The complexity of implementing proprietary VPNs or IPsec protocols has prompted the development of an alternative method of implementing VPNs using SSL.

 For more information about implementation and related topics see <http://www.goog.freeswan.org>; see <http://www.microsoft.com/technet/prodtechnol/windows2000serv/howto/ispstep.msp> for a Microsoft perspective on IPsec.

SSL VPNs

VPNs can be implemented using the SSL protocol. Two methods are available:

- SSL sessions
- SSL gateways

Figure 4.10 depicts both.

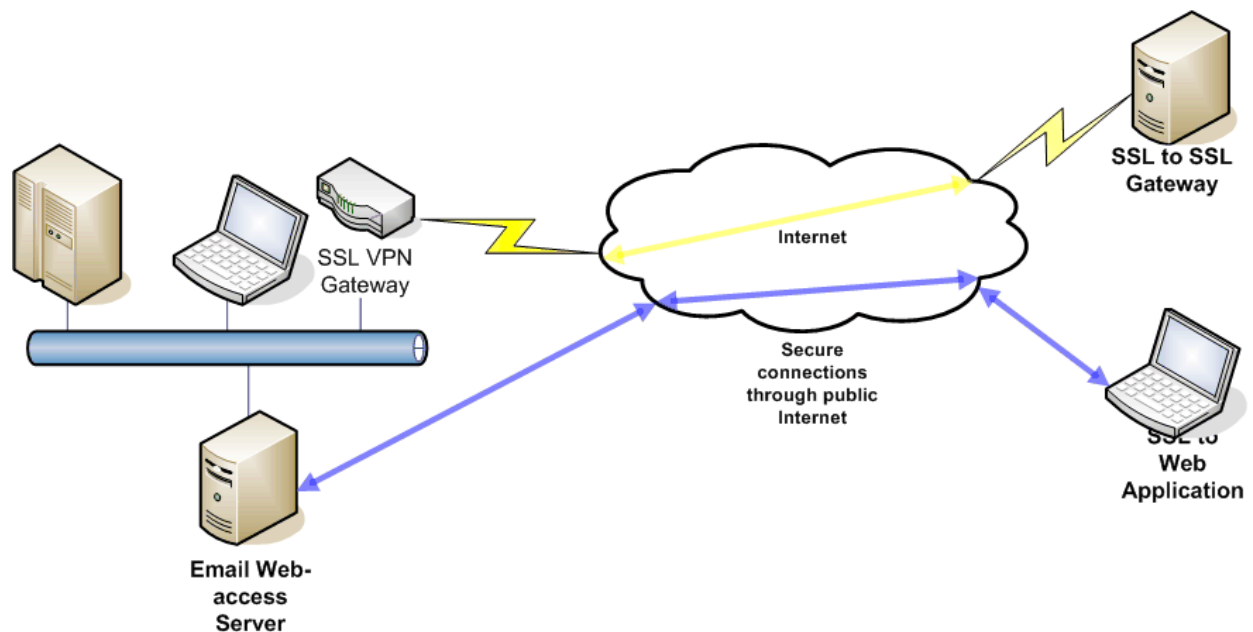


Figure 4.10: SSL can be used to connect securely to a single Web application, such as email, or to an SSL VPN gateway, which allows access to multiple network services.

SSL Sessions

SSL sessions operate at the application layer rather than the network layer, as IPsec VPNs do. Thus, it is the application that is responsible for understanding and managing the SSL services to perform encryption. This works well when remote users need access to a small number of applications, such as email. Another advantage is that this approach only maintains a connection to the remote device as long as the application is needed.

A drawback of this method is the responsibility for managing the SSL services placed on the application. If the user needed access to additional applications, the applications would have to support SSL as well. An alternative method for providing access to multiple applications without requiring each to support SSL is to use an SSL gateway.

SSL VPN Gateways

SSL VPN gateways are devices that provide network-to-network secure communication, much like IPsec VPNs. The major difference is that SSL VPN gateways do not require preinstalled software on the client. SSL VPN gateways provide access to remote applications through Web browsers and many will support common services such as:

- HTTP
- HTTPS
- FTP
- NFS
- Telnet
- SSH
- Email

In addition, some SSL VPN gateways use port forwarding, but this can require browser add-ons be pushed to the client during the session, which can cause compatibility problems with browsers and could violate some browser security policies.

Benefits of VPNs

Regardless of how a VPN is implemented, there are significant advantages to using VPNs. First, they allow remote access to network services without the need for dedicated network connections. Before the advent of VPNs, enterprises relied on costly T1 or other dedicated telecommunications lines to implement point-to-point networking solutions.

Second, VPNs offer flexibility. Without being hampered by the need for dedicated lines, any mobile user can have access to the corporate network from any remote device, assuming it is properly configured.

Another advantage of VPNs is that vendors now support policy enforcement through VPNs. For example, an organization may dictate that only email and calendaring applications are accessible from unmanaged devices. Auditing information can also be collected on remote sessions.

VPNs are relatively low-cost methods for extending the reach of an enterprise network. The security mechanism implemented within a network—such as identity management, access controls, auditing, and other measures—are now available to remote users as well. However, even with all the benefits of VPNs, they are no panacea when it comes to preventing information theft.

Limits of VPNs for Preventing Information Theft

Information theft can occur anywhere information is stored, transmitted, accessed, viewed, or documented. Information security practitioners have several methods for preventing information theft:

- Access controls on files and devices to limit who can retrieve and write data
- Firewalls to control network traffic in and out of a enterprise network
- Content-scanning systems to analyze the content of network traffic and prevent the inflow of malicious software or the outflow of confidential or sensitive information
- Antivirus applications to detect and remove malicious software
- Encryption to protect the confidentiality and integrity of communications
- Forensic techniques to analyze security breaches

Yet, even with this wealth of techniques, many systems administrators and application managers still face one more hurdle: unmanaged devices.

The security mechanisms described in this chapter and in the previous chapter will protect information up to a point. Consider the security measures in place to protect remote users who access email from the road.

- The user has to authenticate to the email application and once done, the user will have access only to email.
- As users can connect remotely to the corporate network, systems administrators have established firewall controls to limit the type of traffic allowed in and out of the network and they have implemented content scanners to prevent malware from slipping onto the email server.
- The SSL connection encrypts data as it is transmitted so that no one other than the intended recipient can read the message.
- Digital certificates are used to authenticate the email server so that the user can be sure he or she is working with the proper server.
- Email messages and file attachments are protected with file access controls.

From the source at the email server all the way through transmission to the client device, information is protected by a number of measures designed to prevent known vulnerabilities. Yet even with all these protections, information is still vulnerable if its destination is an unmanaged device.

Unmanaged devices, such as public kiosks or computers in hotel business centers, can present a number of security threats:

- Keyloggers that capture usernames and passwords
- Video frame grabbers that take snapshots of the display
- Browser cache monitors that copy data from browsers caches
- Viruses, worms, and other malware
- Spyware that tracks online activity

These same threats can compromise managed devices, but at least systems administrators and security managers can put countermeasures in place, define and enforce security policies, and monitor the security state of these devices.

Clearly, protecting information within the boundaries of the corporate network and in transmission is necessary but is not sufficient to prevent information theft. The next chapter examines in detail the countermeasures that can be implemented to protect information when it must be used on unmanaged devices.

Summary

Keeping information secure during transmission is an essential element of maintaining a secure information infrastructure. Encryption is a fundamental process that uses several technologies to ensure the confidentiality of information. In addition to the complex mathematical techniques used in cryptography, practitioners have addressed the practical needs of using encryption in the enterprise with frameworks such as PKI and VPNs. Like the network security mechanisms described in the previous chapter, the techniques outlined in this chapter are necessary but not sufficient to prevent information theft. The next chapter explores the final component, protecting information on unmanaged devices.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit <http://nexus.realtimepublishers.com>.