

Information

Security

By Bryan Hinton

1.0 Data Security

Four key issues arise in the transmission of digital data: tampering, impersonation, repudiation, and eavesdropping. When transmitting data over a medium, data tampering, identity impersonation, repudiation, and digital eavesdropping are inherent problems (Ford, 1994, p. 21). Confidentiality is a concern for both the United States government and the everyday, American citizen. When properly employed, Public Key cryptography provides a solution that addresses all of these problems (Schneier, 1995, p. 2).

Data tampering is an imminent and real threat. When data crosses a transmission medium, a potential for tampering exists. When the content of the transmission's information is modified, the data or contents of the transaction have been tampered with (Ford, 1994, p. 21). Ford says that eavesdropping also occurs frequently (Ford, 1994, p. 22). A third party can easily view the contents of a data transmission. In conjunction with eavesdropping and data tampering, identity impersonation imposes a dangerous hazard (Ford, 1994, p. 22). Identity impersonation enables a person to perform illegal actions on another person's behalf (Ford, 1994, p. 22). In the context of Internet transactions, illegal bank account transactions are executed on another person's behalf. The fourth type of threat, repudiation, closely relates to identity impersonation (Ford, 1994, p. 23). Repudiation occurs when a user performs an action and later denies that he performed this action (Ford, 1994, p. 23). Data tampering, impersonation, repudiation, and digital eavesdropping are discussed and defined in the context of cryptography.

Several key issues arise in the study of public key cryptography. Cryptography entails the detailed analysis of encryption algorithms (Schneier, 1995, p. 1). The RSA algorithm is a fundamental piece of public key cryptography (Pomerance, 1990, p. 9). While cryptography addresses the four

security concerns, public key cryptography specifically provides a safe means to prevent eavesdropping and data tampering. Several mechanisms, such as digital signatures and digital certificates, use public key cryptography to provide non-repudiation and identity verification.

For thousands of years, Cryptography has been used to secure information. Cryptography entails the study and analysis of cryptosystems (Schneier, 1995, p. 1). A cryptosystem encompasses the process of encryption and decryption. Encryption and decryption are the fundamental processes in a cryptosystem. There are two types of cryptosystems: symmetric and asymmetric. Asymmetric cryptosystems are also known as public key cryptosystems. Digital certificates and digital signatures utilize public key cryptosystems for identity verification and authentication. Internet commerce is using public key cryptography. Both email and web transmissions can be encrypted using public key cryptography.

2.0 Cryptography

Cryptography is the science of secure data transmission. Cryptography is the art and science of keeping messages secure (Schneier, 1995, p. 1). The goal of cryptography is to build systems that are hard to attack so that secure network communications and services can be provided (Kou, 1997, p. 13).

2.1 Key Issues

Four key issues arise in the study of cryptography. Tampering, eavesdropping, impersonation, and repudiation are discussed in the following subsections.

2.1.1 Tampering

Tampering with data transmission occurs when data is sent across the Internet and a third party views and subsequently modifies

the contents of the transmitted data (Ford, 1994, p. 28). This occurs without the permission of the sending and receiving parties. Data tampering usually occurs during Internet operations that involve financial information. Online, bank account transactions are vulnerable to this type of threat. When a bank customer submits an account transfer on a banker's website, a criminal can potentially alter the contents of the data transmission, thus modifying the amount of money transferred or the account identification number where the money is transferred. For example, a malicious user could modify the withdrawal amount message that is sent from the ATM to the bank by decreasing it from \$1000 to \$100 (Ford, 1994, p. 28). Subsequently, the user could modify the approval message that is sent from the bank back to the ATM, thus increasing the approved withdrawal amount from \$100 to \$1000. Internet bank account transactions are also susceptible to tampering.

2.1.2 Eavesdropping

The second type of security vulnerability is **eavesdropping**. Eavesdropping occurs, when a third party views the contents of an Internet data transmission without the permission of the sending and receiving parties (Ford, 1994, p. 28). Military institutions often transfer time critical data across data lines. The data is susceptible to eavesdropping by foreign government parties. During times of war, military institutions send time-critical commands to military posts. The confidentiality of the transmitted information is very important as lives are at stake.

Uninvited third parties can also record credit card numbers sent over the Internet. When an Internet user submits his credit card number over the Internet, a malicious user can easily watch the traffic and record the user's credit card number. Bank account numbers, social security numbers, and any type of personal data are all susceptible to eavesdropping. One of the most important areas of eavesdropping involves classified government information.

The essence of this problem lies in the fact that sensitive data can be watched in transit.

2.1.3 Impersonation

A third type of security vulnerability, often called identity **impersonation** is also prevalent. A user should not be able to masquerade as someone else and subsequently perform illegal actions on the other person's behalf (Schneier, 1995, p. 2). During an online account transaction, the identity of the source must be verified. For example, an unauthorized user could perform a bank account transfer into a foreign, off-shore account while logged into the bank as another user. Classified government documents are often sent over the internet. It is of utmost importance for the receiver of the document to be able to verify the sender. Identity impersonation is an ever-increasing problem.

2.1.4 Repudiation

The fourth type of security vulnerability, **repudiation**, occurs when a user sends a message and later denies that he sent the message (Schneier, 1995, p. 2). For instance, a legally binding signature may need to be provided with an electronically submitted document. Thus a non-repudiation mechanism is needed to present evidence of document submittal time.

Therefore, a **confidentiality** mechanism is needed that addresses impersonation, tampering, repudiation, and eavesdropping. Confidentiality is the protection of a document or message against disclosure to unauthorized parties (Ford, 1994, p. 26). In a broader context, confidentiality encompasses the four primary security vulnerabilities. Cryptography provides secure data transmission and confidentiality.

2.2 Cryptosystems

A cryptosystem is a specific system that is used to secure messages (Schneier, 1995, p. 2). For thousands of years, people have

devised schemes for securing messages. During the rule of Julius Caesar, Caesar would send messages to people via messengers. Caesar did not trust the messengers so he would shift all of the letters in his messages. A's would be replaced by D's and B's would be replaced by E's. In this case, n is the number of alphabetic characters that each character is shifted in the message. This is an example of a very simple encryption system, known as a Caesar cipher (Kurose, 2002, p. 610). A cipher is the mathematical algorithm used to scramble information (Schneier, 1995, p. 2). An algorithm is a step-by-step procedure that is used to carry out a task. The Caesar cipher is weak as there are only 25 possible combinations of each character in the scrambled text. Another type of primitive cipher, the mono-alphabetic cipher, presents an improvement over the Caesar cipher. Given a text message, the mono-alphabetic cipher substitutes every character in the alphabet with another character in the alphabet (Kurose, 2002, p. 610). Rather than shifting the characters in the text by an offset, the mono-alphabetic cipher uses pairings of alphabetic characters. Any given character can be substituted with another character. Thus, there are 26 factorial possible pairings of letters. However, a text message that has been scrambled via a mono-alphabetic cipher can easily be broken. If the intruder knows the potential contents of the message or simply recognizes frequently occurring letters or sequences of letters in normal English text, the intruder can perform a statistical analysis of the scrambled text. Hence, he can compute the frequency of potential words and deduce the alphabetic pairings of characters in those words (Kurose, 2002, p. 611). Both the Caesar and mono-alphabetic ciphers present the inherent weaknesses of early cryptography. In effect, the method of scrambling the text can easily be broken. In addition, the receiving party must know the method used to scramble the text so that he can descramble the text. Thus, a secure means of transmitting the method to the receiving party is required.

By the 20th century, cryptosystems began to formally develop. With the onset of World War I, technology was advancing at a rapid rate and overseas countries were an imposing threat. At the onset of World War I, the U.S. government began intercepting undecipherable messages from foreign entities. Subsequently, the U.S. government sought the outside help of William Friedman in deciphering these messages. Friedman coined the terms cryptology and cryptanalysis. Cryptography is practiced by cryptographers (Schneier, 1995, p. 1). Cryptanalysis is the science of breaking undecipherable messages (Schneier, 1995, p. 1). The field of cryptology encompasses both cryptanalysis and cryptography (Schneier, 1995, p. 1).

2.2.1 Symmetric and Asymmetric Systems

Two types of systems exist; asymmetric and symmetric cryptosystems (Ford, 1994, p. 65). Both systems provide a means to protect data. Data can be defined as any textual information that is transmitted or sent from one person to another. In the context of cryptography, data is confined to the digital spectrum. Thus, digital data or merely data is any form of textual or non-textual information that can be transmitted over the Internet. In its native form, data is referred to as plaintext. Generally speaking, data consists of human readable English text. Asymmetric cryptography is also known as public key cryptography. However, symmetric key cryptography superceded public key cryptography. In 1976, Whitfield Diffie and Martin Hellman introduced public key cryptography (Schneier, 1995, p. 513). Symmetric and asymmetric cryptosystems will be discussed in the context of encryption and decryption in the next section.

3.0 Encryption and Decryption

Encryption and decryption are fundamental processes in the study of cryptography. Encryption and decryption are the processes by which messages are rendered into decipherable and non-decipherable forms. A decipherable message is one that can be read easily by the human eye. A decipherable message usually consists of textual information that can be read by the human eye or processed by a computer program. Decipherable information or data is known as plaintext or clear text (Rhee, 1994, p. 2).

Encryption is defined as the process of hiding or disguising a data message, so that its contents are not decipherable (Schneier, 1995, p. 4). Encryption transforms plaintext into cipher text. On the other hand, decryption is the process of un-disguising a data message so that its contents are readable. Decryption transforms cipher text into plaintext. A data message is first encrypted, transmitted, and then decrypted.

A key is the fundamental mechanism in the encryption and decryption process. A data message is encrypted into cipher text using a key as the input to an encryption algorithm. Decryption is the reverse process. The data message is decrypted and converted to plaintext using the key. The forward and inverse transformation is known as a cryptosystem. A cryptosystem is “an algorithm, plus all possible plaintexts, cipher texts, and keys” (Schneier, 1995, p. 4).

If the same key is used for the forward and inverse transformation, then the cryptosystem is known as a symmetric key or secret key cryptosystem (Ford, 1994, P. 66). The underlying security of a symmetric key cryptosystem relies on the secrecy of the symmetric key. Given a data message, the symmetric key is used for both encryption and decryption. Thus, the key must be protected from outside parties. If a given message is encrypted with the secret key, anyone who possesses the secret key can

decrypt the message. This is the underlying problem with symmetric key cryptosystems. The secret key must be protected from outside parties. If two users are on different sides of the country and they wish to transmit a data message to each other using the symmetric key, both users must have a copy of the symmetric key. The first user uses the symmetric key to encrypt the message and the second user uses the symmetric key to decrypt the message. If the first user has the secret key and the second user does not have the secret key, then the first user must find a secure method of transmitting the key to the second user so that the second user can decrypt the message that the first user encrypted with the secret key. Oftentimes, a secure method of transmitting the secret key to the receiving user does not exist. Third parties can easily eavesdrop on phone or data lines and listen for the contents of the secret key. This is the fundamental problem with symmetric key cryptography.

In contrast, if a different key is used for the forward and inverse transformation then the system is known as an asymmetric cryptosystem (Schneier, 1995, p. 4). Asymmetric cryptosystems are also known as public key cryptosystems. Asymmetric cryptosystems address the secret key problem inherent in symmetric key cryptosystems. Because symmetric cryptosystems use a single key for encryption and decryption, transmission of the secret key to the receiver is a security concern. In order for two parties to send encrypted information to one another, they must exchange the secret key. Exchanging a secret key over the Internet or public transmission medium poses a security risk. Anyone who intercepted the secret key could use it to later decrypt encrypted data transmissions between the sender and receiver parties. Public key cryptography utilizes public and private key pairs to overcome the problem of a shared, symmetric key.

4.0 Public Key Cryptography

Public key cryptography enables remote parties to securely exchange data. Public and private key pairs are a fundamental piece of public key cryptography. Key pairs and key strength will first be discussed. Key generation methods will then be defined. Subsequently, RSA and El Gamal, which are the two primary public key encryption algorithms will be discussed in detail.

Public key cryptography utilizes two keys. The first key, known as the public key, is used in the forward process of encryption (Schneier, 1995, p. 4). The second key, known as the private key, is used in the inverse process of decryption.

4.1 Public and Private Key Pairs

The use of public and private key pairs is a fundamental aspect of public key cryptography. Key pairs overcome the inherent security risk of a single private key in symmetric key cryptosystems.

A public key is a unique number in a range of values called the key space (Schneier, 1995, p. 3). A private key is also a unique number in the key space. A public key algorithm uses the public key to encrypt a plaintext message. The private key is then used to decrypt the cipher text. A fundamental requirement of public key cryptosystems is that the private key cannot be deduced from the public key in a reasonable amount of time (Pomerance, 1990, p. 6). Given a plaintext message M , a public key K , and a public key encryption algorithm E , $E_k(M) = C$ is defined as a public key encryption function (Schneier, 1995, p. 5). The public key encryption algorithm takes the plaintext message M as its input and transforms M into cipher text using the public key K . The reverse process of decryption is denoted by $D_k(C) = M$ (Schneier, 1995, p. 5). A public key cryptosystem addresses the key secrecy problem inherent in symmetric key

cryptosystems. Since the private key cannot be deduced from the public key, the public key can be distributed to the public. Thus, anyone possessing the public key can encrypt a message. The cipher text can only be decrypted by the person holding the private key. The system is called a public key cryptosystem because the encryption key can be distributed to the public. Given two users, Alice and Bob, each person generates a public and private key. Alice sends her public key to Bob and Bob sends his public key to Alice. Subsequently, Alice encrypts a plaintext message using Bob's public key. Alice sends the cipher text to Bob, and Bob decrypts the cipher text using his private Key. Next, Bob encrypts a plaintext message using Alice's public key. Bob sends the cipher text to Alice and Alice decrypts the cipher text using her private key (Schneier, 1995, p. 32). This is the basic operation of a public key algorithm.

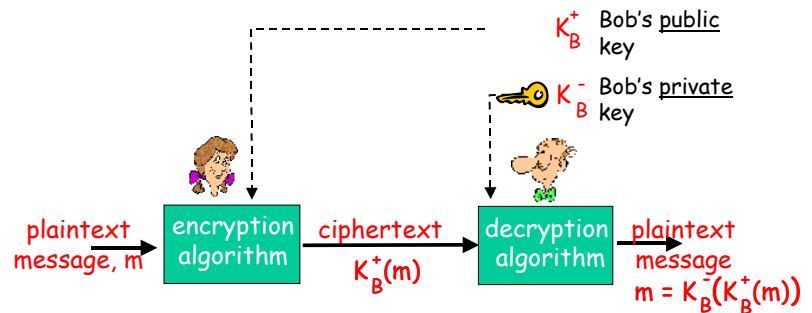


Figure 4.1 Public Key Encryption Scenario

4.1.1 Key Strength

One may wonder how secure the cipher text is. Encryption strength is dependent on the strength of the encryption algorithm and the length of the encryption key. Given a key of length n bits, there are 2^n possible keys. Generally speaking, if the length of the key is too small then the security of the system is compromised. On the other hand, a key whose length is too large may consume too much time when used by an encryption algorithm (Schneier, 1995, p. 166). It is theoretically possible for a cryptosystem to achieve perfect secrecy. Perfect secrecy occurs when the cipher text reveals no information at all about the plaintext.

However, most cryptographic algorithms try to minimize the amount of information that the cipher text reveals about the plaintext. Therefore, the job of a cryptanalyst is to try and figure out the key and the plaintext.

4.1.2 Key Generation Methods

A shared public key is a function of either the prime factoring problem or the discrete logarithm problem (Pomerance, 1990, p. 6). In the case of the prime factoring problem, a public key is created by generating two, very large prime numbers (Rhee, 1994, p. 10). The prime numbers are then multiplied together. One of the issues associated with generating large prime numbers is randomness. Large, random numbers are generated and then tested for primality. The other method used for public key cryptography is the discrete logarithms in a finite field problem. The problem lies in finding the discrete logarithm of a given number. This is the inverse problem of modular exponentiation. For example, given $a^x = b \pmod{n}$, one must find x , where x is an integer solution. The problem surfaces when 1024 bit numbers are used as it is very difficult to find x .

4.2 Public Key Encryption Algorithms

A number of algorithms efficiently implement public-key cryptosystems. RSA is the most popular public key encryption algorithm. RSA is based on the prime factoring problem. The second most widely used public key encryption algorithm is the El Gamal algorithm. El Gamal is based on the discrete logarithm problem. The following sections will discuss the RSA and El Gamal algorithms in detail.

4.2.1 RSA

RSA was invented by Rivest, Shamir, and Adleman. The RSA algorithm is based on the prime factoring problem. The RSA algorithm is a function of multiplying two, very large

prime numbers (Schneier, 1995, p. 467). The public and private keys are both created by multiplying two very large, random, prime numbers together. A prime number is a number that is only divisible by one and itself. The RSA algorithm works as follows. The public and private keys are created by randomly generating two very large prime numbers, say p and q of equal length. A large prime number is greater than 512 bits in length. The product of p and q is then computed: $n = p * q$. Next, an encryption key e , is randomly chosen, so that e and $(p - 1)(q - 1)$ are relatively prime. Thus, $(p - 1)$, e , and $(q - 1)$ and e have no common divisors. Two numbers are relatively prime if they share no factors in common except for one. The decryption key, d , is then produced using the extended Euclidian algorithm as follows:

$$ed = 1 \pmod{(p-1)(q-1)} \text{ or}$$

$$d = e^{-1} \pmod{((p-1)(q-1))}$$

Thus, we are effectively computing the inverse of a number modulo n where d and n are relatively prime. d is calculated so that $(de - 1)$ is divisible by $(p - 1)$ and $(q - 1)$. The numbers e and n form the public key and the number d forms the private key. P and Q must be hidden and thrown away as they are fundamental in reproducing the key.

The process of encryption works as follows. Given a message M , divide the message into blocks of size smaller than n . Next, we use the formula $C_i = M_i^e \pmod{n}$

Thus, for each message block m_i , we raise M_i to the e power modulo n where C_i denotes a cipher text block. To decrypt, we use the formula

$$M_i = C_i^d \pmod{n}.$$

For example:

$$\text{Let } P = 61, Q = 63, N = PQ = 3233$$

$$\text{Let } E = 17$$

$$\text{Where } D = 2753$$

Thus, given the plaintext value 123, the process of encryption works as follows:

$$\begin{aligned} \text{encrypt}(123) &= (123^{17}) \pmod{3233} \\ &= 33758791744665371559659295881767 \\ &= 9803 \pmod{3233} \\ &= 855 \end{aligned}$$

$$\begin{aligned} \text{decrypt}(855) &= (855^{2753}) \bmod 3233 \\ &= 123 \end{aligned}$$

The difficulty of cracking the RSA algorithm lies in finding the prime factors of a number. For 2048 and 4096 bit keys, the computational time to factor a number of this magnitude exceeds the lifetime and resources of the normal human being.

4.2.2 El Gamal

In contrast to the RSA algorithm, the El Gamal scheme utilizes the discrete logarithms in a finite field problem (Schneier, 1995, p. 477). The El Gamal algorithm works as follows. First, a prime p is chosen, and then two random numbers g and x are generated such that g and x are less than p . Next, $y = g^x \bmod(p)$ is calculated. The public, shared key is now y , g , and p and the private key is x . The encryption process works as follows. Given a plaintext message M , generate a random number k , which will be denoted as the key. k must be relatively prime to $p - 1$. Next, calculate $a = g^k \bmod(p)$ and $b = y^k \bmod(p)$. The tuple a, b is the cipher text. Decryption is performed by calculating $M = b/a^x \bmod(p)$ where M is the resulting plaintext. This problem becomes very difficult when p and n are chosen on the magnitude of 10^{300} . In public key cryptography, the discrete logarithm problem is often applied over elliptic curves.

Many public key algorithms exist, but few are suitable for both encryption and digital signatures. RSA and El Gamal work for both encryption and digital signatures (Schneier, 1995, p. 461).

5.0 Certificates and Signatures

Public key cryptography is used to validate digital Signatures. Digital certificates use public key cryptography for identity verification.

Digital signatures provide proof of a sender's identity when he sends a document.

Digital signatures address the problem of repudiation and impersonation. Once the sender transmits the document, he cannot deny that he sent the document. Digital signatures utilize public key cryptography. Given two users, Alice and Bob, Alice desires to send Bob an encrypted document. An additional requirement is imposed here though. Bob must verify that Alice sent the document and not someone else. The procedure works as follows. Alice produces a one way hash of the document. Next, Alice encrypts the hash with her private key. In effect, Alice is signing the document. The signed document consists of the document and the encrypted hash string. Next, Alice encrypts the signed message with Bob's public key. Alice then sends the message to Bob. Bob decrypts the message with his private key. Bob produces a one way hash of the document. Next, Bob decrypts the hash string included with the document using Alice's public key. Finally, Bob compares the included hash string with the hash value that he generated. Therefore, Bob authenticates Alice as the original sender of the document (Schneier, 1995, p. 37).

A key issue in public key cryptography is that of key distribution (Rhee, 1994, p. 461). Digital certificates address the problem of public key authenticity. A certificate is a public key that has been signed by a trustworthy source. Certification authorities maintain databases of public keys for individuals. The CA validates the integrity and identity of a public key by associating a name, address, and other information with a public key. First, the CA verifies the person or entity. The CA then creates a certificate for the person or entity. The certificate associates the public key of a person to his or her identity. After binding the public key with the identity, the CA digitally signs the certificate. The ISO X.509 protocol specifies the structure of a public key certificate (Schneier, 1995, p. 575). The CA creates a signed certificate that contains the user's name and his public key. A validity time frame is often associated with the key. For example, given two users, Alice and Bob, assume that Alice would like to send a

message to Bob. Alice retrieves Bob's certificate from the CA and verifies the signature. Every certificate is signed with the CA's private key. Alice then uses Bob's public key, which is held by the CA to encrypt a message and send it to Bob.

In conjunction with public key cryptography, digital signatures and digital certificates provide a means of securing internet commerce.

6.0 Securing Internet Commerce

Public key cryptography addresses several fundamental areas of secure data transmission. Internet email, web server transactions, and client/server communication can all be secured using public key cryptography.

6.1 Secure Email

Secure email transmission utilizes public key cryptography to ensure confidentiality between senders and receivers. Instead of encrypting data with the receiving party's public key, a random session key will be used. A random session key is used for both encryption and decryption but is thrown out after the data transaction is finished (Kurose, 2002, p. 625). The session key is thrown out after the transmission. The use of a one time, session key to encrypt the email message ensures the security of the email transmission. In this scenario, public key cryptography is used to encrypt the session key, thus providing a safe means of transporting the session key. The use of a session key provides confidentiality. There is an inherent amount of overhead associated with encrypting and decrypting data using public key cryptography. Thus, public key cryptography is usually used in conjunction with symmetric key cryptography (Kurose, 2002, p. 626). Symmetric key cryptography is faster and more efficient in encrypting and decrypting bulk text (Kurose, 2002, p. 626). In order to provide data

integrity and identity authentication, message digests and digital signatures are used.

In the following scenario, Alice desires to send an email message to Bob.

- 1) Alice hashes the email message m , thus obtaining a message digest. (SHA, MD5).
- 2) Alice signs the message digest with her private key to create a digital signature.
- 3) Alice concatenates the unencrypted message with the digital signature.
- 4) Alice selects a random session key K_s .
- 5) Alice encrypts the message M with the session key K_s (CAST, triple-DES, IDEA).
- 6) Alice encrypts the session key K_s with Bob's public key (RSA).
- 7) Alice concatenates the encrypted session key and the encrypted message.
- 8) Alice sends this message to Bob via email.

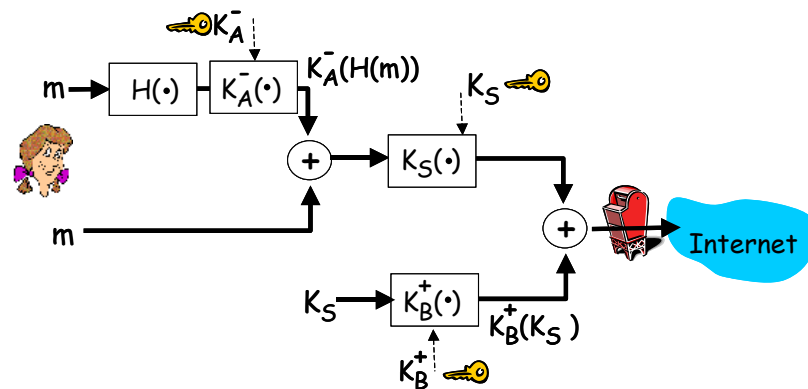


Figure 6.1 Public Key Encryption Email Scenario

- 1) Bob receives the email message from Alice.
- 2) Bob uses his private key to decrypt the encrypted session key K_s , thus obtaining the session key K_s .
- 3) Bob uses the symmetric session key K_s to decrypt the message m .

- 4) Bob applies Alice's public key to the digital signature to obtain the message digest.
- 5) Bob hashes the unencrypted data message with the same hash function that Alice used.
- 6) Bob compares the message digest that Alice sent with the message digest that he computed. If the two are the same, then Bob has verified that Alice is the sender and that the data in the message was not modified in transit.

Therefore, the described system of secure email provides confidentiality, authentication, and data integrity.

6.2 Secure Web Transactions

In addition to securing email, public key cryptography provides application layer security to web servers and related applications. The Secure Socket Layer (SSL), developed by Netscape Inc. utilizes public key encryption to encrypt network data that is sent over TCP sockets (Kurose, 2002, p. 635). When a web user connects to a web address with the https prefix, the web user is utilizing SSL and public key encryption.

- 1) Browser sends server cryptographic preferences and SSL version number. Cryptographic preferences include the symmetric key algorithm that the client and server will be using to encrypt transmitted data.
- 2) The server sends the browser an SSL version number, cryptographic preferences and a certificate. The

- certificate contains the server's public key (RSA) and is certified by a CA. Hence, the certificate has been signed by the CA's private key.
- 3) The browser holds an entrusted list of CA's and a public key for each CA on the list. When the browser receives the certificate from the server, it checks to see if the CA is on the list. If it is on the list, the browser uses the CA's public key to validate the certificate and obtain the server's public key.
 - 4) The browser then creates a random session key, encrypts it with the server's public key, and sends the encrypted session key to the server.
 - 5) The browser sends a message to the server informing it that future messages sent from the client will be encrypted with the session key. The browser then sends a separate encrypted message indicating that the browser portion of the handshake is finished.
 - 6) The server then sends a message to the browser that tells the browser that future messages sent from the server will be encrypted with the session key. The server then sends a separate encrypted message indicating that the server portion of the handshake is finished.
 - 7) The SSL handshake is finished. The browser and server use the session keys to encrypt and decrypt the data that they send to each other.

In the following diagram, a client and web server are communicating via the Secure Socket Layer.

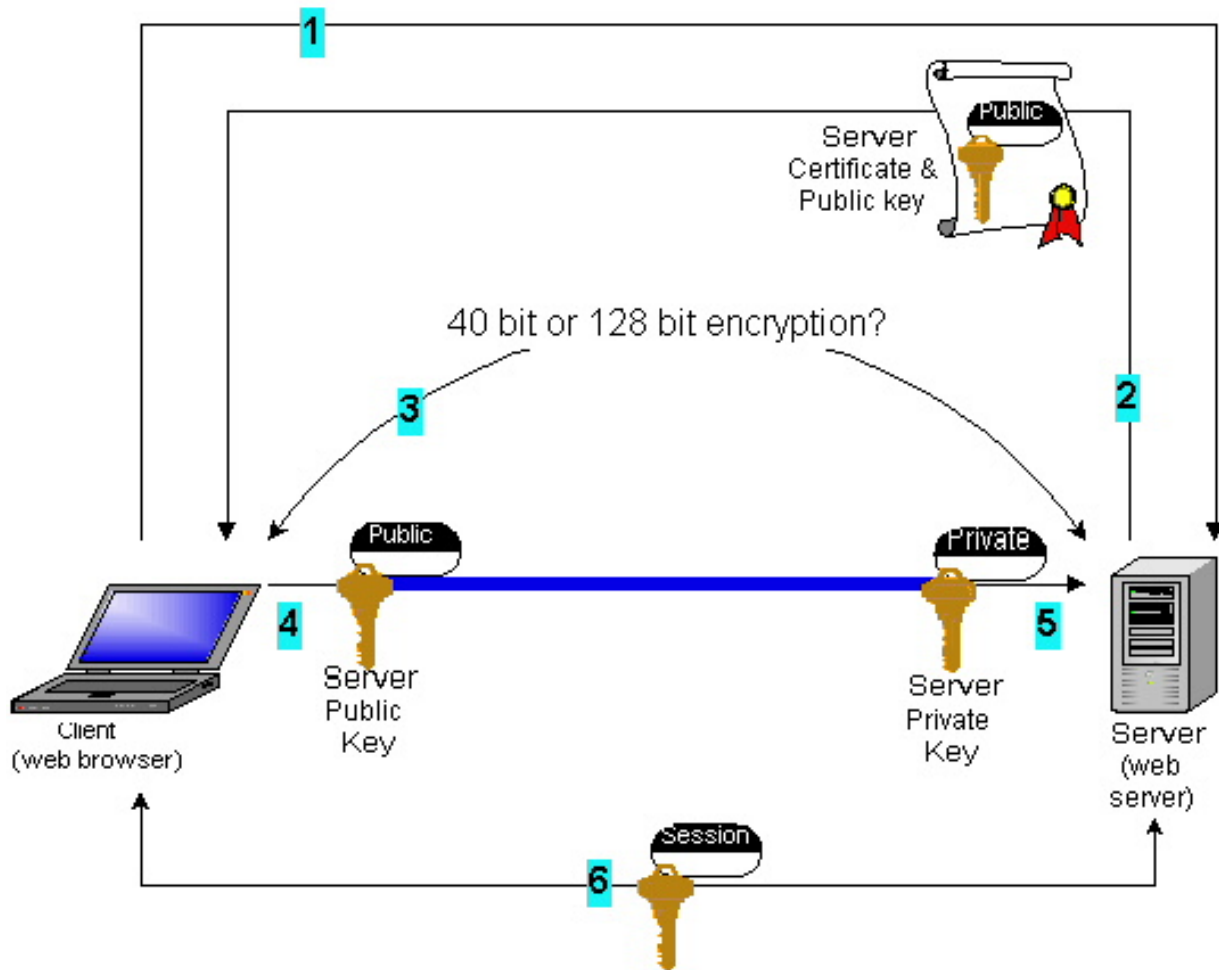


Figure 6.2 Secure Web Transaction

7.0 The Backbone of Security

Public key cryptography is the backbone of secure email, web, and client/server data transmission.

Cryptography has been used for hundreds of years to secure critical information. While cryptography entails the study and analysis of cryptosystems, cryptanalysis entails the detailed study of cryptosystems. Public key cryptosystems overcome the inherent problems in symmetric cryptosystems by utilizing public key pairs. Public key cryptography addresses

the four primary security concerns: tampering, eavesdropping, identity impersonation and repudiation.

Public key cryptography has a widespread distribution of applications. Internet commerce is secured using public key cryptography. Both email and web transmissions are encrypted using public key cryptography.

8.0 Bibliography

- Ford, Warwick. Computer Communications Security: Principles, Standards Protocol and Techniques. New Jersey: Prentice Hall; 1994.
- Kou, Weidong. Networking Security and Standards. Massachusetts: Kluwer Academic Publishers; 1997.
- Kurose, James F. Computer Networking: A Top Down Approach Feature the Internet. Addison Wesley; 2002. 605-666p.
- Kwangjo, Kim. Information Security and Cryptology – ICISC 2001. New York: Springer-Verlag Berlin Heidelberg; 2002. 1-15p.
- Pomerance, Carl. Cryptology and Computational Number Theory. Volume 42. American Mathematical Society; 1990.
- Rhee, Man Young. Cryptography and Secure Communications. Singapore: McGraw-Hill; 1994.
- Schneier, Bruce. Applied Cryptography. John Wiley & Son, Inc; 1995. 497p.
- Riverbank Labs at:
<http://www.geneva.il.us/riverbnk/riverpag.htm> (Accessed Oct. 3, 2002).
- U.S. Nuclear Regulatory Commission: Introduction to Cryptography at:
<http://www.nrc.gov/site-help/eie/intro-crypt.html> (Accessed Oct. 3, 2002).
- Introduction to Public Key Cryptography at:
<http://developer.netscape.com/docs/manuals/security/pkin/contents.htm>
(Accessed Oct. 18, 2002).
- RSA Security Frequently Asked Questions at:
<http://www.rsasecurity.com/rsalabs/faq/3-1.html> (Accessed Oct. 19, 2002).
- Jorgensen, Poul Henrik. Z39.5 and Cryptography. Zig July 13, 2000.
<http://lcweb.loc.gov/z3950/agency/zig/meetings/leuven/presentations/poul-crypt.ppt>