Chapter 14

Cyber Security

With the Internet making inroads into almost all facets of everyday life, not only business information, but a large amount of personal information too is now digitised and stored on computers connected to the Internet. Information has its own value and can either be destroyed or used with malicious intent or for commercial interest.

Systems connected to the Internet are potential targets for eavesdropping and destruction/tampering of the data stored in them. A website offering services on the Internet is vulnerable to attacks, which render the site non-functional resulting in denial-of-service. Carrying out traffic analysis could reveal valuable information regarding the channels of communication from a server. Masqueraders pretending to be authorised users could gain access to privileged areas. Authorised e-commerce or e-governance transactions could be modified or replayed for commercial gain.

Attacks like the above, when carried out, could result in crisis situations due to service outages, unauthorised use of computing systems, compromise of data and direct financial losses. In the case of a successful attack on critical infrastructure such as power grid, the consequence could even be the loss or endangerment of human life. With the growing Internet economy, such incidents would result in loss of trust in computers and networks, and be detrimental to the growth of public confidence in the Internet.

Systems, networks and data have to be protected to guard against such attacks which could originate from within the organisation or from outside. It is extremely important to secure internal information systems from being attacked. With client machines on an organisation's internal LAN routinely accessing the Internet, they too become targets for attack by unscrupulous elements. Most surveys carried out worldwide have indicated that the threat to an organisation is much higher from within the organisation than from outside.

14.1 Cyber Attacks

Attacks can be classified as executable-based or network-based. In the case of the former, the attack happens only when a program is executed on the targeted computer system through either of the following:

- **Trojan**—a computer program that appears to have a useful function, but also has hidden and potentially malicious functions that evade security mechanisms, sometimes by exploiting legitimate authorisations of a system entity that invokes the program. The idea of modifying a normal program to do nasty things in addition to its usual function and arranging for the victim to use the modified version is known as a Trojan horse attack.
- **Virus**—a program fragment that is attached to a legitimate program with the intention of infecting other programs. It is hidden, self-replicating computer software, usually malicious logic, that propagates by infecting, i.e., inserting a copy of itself into and becoming part of another program. A virus cannot run by itself; it requires its host program to be run to make the virus active.
- **Worm**—a computer program that can run independently, can propagate a complete working version of itself onto other hosts on a network, and may consume computer resources destructively. It differs from a virus only in that

a virus piggybacks on an existing program, whereas a worm is a complete program itself. Viruses and worms both attempt to spread themselves and both can do severe damage.

- **Spam**—is also a major source of cyber attacks. There are some estimates according to which 70 percent of all e-mail is junk mail, or spam. While spam by itself has nuisance value in that it clogs most of the Internet highways around the world causing losses by way of improper utilisation of bandwidth, it is used to propagate viruses and worms. Junk mail appears to be promotional material, similar to advertisements and catalogues in the physical world. Unsuspecting users become victims as soon as they click on attachments. Trojans and spy-ware get installed on their systems. Information and data on all activities of interest thus gets reported from users' computers to sites whose forwarding addresses have been installed as part of spy-ware. This is a form of information espionage, which may be used by competitors. At the national level however, intelligence agencies may collect useful data from important systems that may have been compromised by sending spam.

In order to protect systems from executable-based attacks, anti-virus measures must be deployed on desktops and servers and on the corporate gateway for data coming in from external sources.

14.2 Hacking

Externally accessible systems are targets of *hacking*. Hackers can deface websites and steal valuable data from systems resulting in a significant loss of revenue if it is a financial institution or an e-commerce site. In the case of corporate and government systems, loss of important data may actually result in the launch of information espionage or information warfare. Using *IP spoofing*, attackers often hide the identity of machines used to carry out an attack by falsifying the source address of the network communication. This makes it more difficult to identity the sources of attack traffic and sometimes shifts attention onto innocent third parties.

14.2.1 Phishing

Phishing is the creation of e-mail messages referencing web pages that are replicas of existing sites to make users believe that these are authentic sites. Unsuspecting users are made to submit personal, financial, or password data to such sites from where the data get directed to fraudsters' chosen sites. By hijacking the trusted brands of well-known banks, online retailers and credit card companies, phishers are able to convince up to 5 percent of the recipients to respond to them. According to a tech-security company MessageLabs, the number of phishing attacks increased ten-fold during the year 2004 as compared to 2003. In the month of November, 2004, phishing attacks rose to 4.5 million. Phishing has indeed emerged as a major threat to any organisation or individual conducting business online. Yet another trend associated with these attacks is the singling out of certain companies, especially financial institutions, to be the victim of phishing attacks. This signals the beginning of a wider trend. From a random, scattergun approach there emerge customised attacks designed to take advantage of weakness of some businesses.

14.2.2 IP Spoofing

IP Spoofing is used by intruders to gain unauthorized access to computers. Messages are sent to the computer with the sender IP address of a trusted system. Packet headers of the message are modified to make it appear that the message is coming from a trusted system.

For externally accessible systems such as web, e-mail and FTP servers, protection can be accorded in the following ways.

- Use **Scanning** tools to scan systems connected over an IP network and report on the systems they encounter, the ports available, and other information, such as OS types.
- Put them behind an appropriate Firewall (defined in section 14.3), —preferably in a demilitarised zone (DMZ).
- Disable all services except those absolutely needed.
- Filter all except port-specific traffic to systems (e.g., FTP servers should only receive *ftp* requests and nothing else).
- Turn on system and firewall logs.
- Review the logs on a daily basis.
- Implement Intrusion Detection Systems.
- Establish proxy servers, so that internal client requests for accessing external services are routed through the proxy server. This ensures that the client and the external server are not in direct communication with each other.
- Establish an additional network as a buffer between the internal and external networks

14.3 Firewalls

A *Firewall* is a system or group of systems that enforces an access control policy. Firewalls operate on the basis of a set of user defined rules. These rules govern the flow of data into and out of the firewall. The rule base is created to enforce a specific security policy on the firewall. Rules could decide, for example, which packets of data, depending on the originating IP address, should be allowed to pass into the organisation's network.

14.4 Intrusion Detection Systems

Intrusion Detection Systems (IDS) complement the firewall to detect whether or not those communication channels through the firewall are being exploited. Firewalls can filter incoming and outgoing traffic from the Internet; however, there are ways to circumvent the firewall. For example, external users can connect to the internal intranet via an unauthorised modem that does not pass through the firewall. If the threat comes from within the organisation, the firewall does not recognise those threats because it monitors only traffic between the internal and external network.

There are two types of Intrusion Detection Systems—Host-based (HIDS) Intrusion Detection works based on a reactionary approach in which the Intrusion Detection software monitors system log files. When log activity matches a pre-determined attack signature, an alert is generated and Network-based (NIDS) IDS which works by monitoring real-time network traffic similar to the way a network sniffer functions. Malicious activity is identified by matching network traffic to predefined attack signatures.

14.5 Secure Sockets Layer

Web communications also require additional levels of security to protect against situations such as compromise of credit card numbers when transmitted across the network. Client-server authentication mechanisms (dealt with later in this chapter) must also be installed to guard against business malpractices.

The Secure Sockets Layer (SSL) protocol was developed by Netscape Communications to provide security during a communications session. SSL operates above the TCP layer and provides protection to applications such as FTP, TELNET and HTTP. This includes services such as client and server authentication, data integrity and confidentiality.

Secure HTTP (SHTTP) was developed for CommerceNet, a consortium of companies promoting the establishment of electronic commerce on the Internet. SHTTP provides security to individual transactions.

14.6 Authentication and Assurance of Data Integrity

The electronic environment ushered in by e-commerce and e-governance, while allowing transactions to be conducted with the click of a mouse, also opened up new risks that were not inherent in the paper environment. Copies cannot be distinguished from originals. Information can be modified without leaving any trace.

In cyberspace, two transacting partners, Bob and Alice, need to be assured of each other's identities. When sending a confidential document, Bob must be sure that the document will only be available to Alice and no one else.

In an environment where business transactions take place on the basis of paper documents, a Purchase Order cannot be modified without leaving evidence. The payment amount on a cheque too cannot be modified without leaving a trace.

However, when the transacting parties operate on electronic documents, not only can changes be made in documents without leaving any visible signs, documents can also be 're-played' and made to appear as bona fide transactions. In providing security in the electronic environment, there should be therefore an integration of manual and technical controls appropriate to the risks that a business believes it is exposed to. With the introduction of e-commerce, the new system should offer at least the same reliability as the paper system that it replaces.

Whatever the environment, paper or electronic, securing it necessarily implies the prevention of (a) destruction of information, and (b) unauthorised availability of information through mechanisms for guaranteeing confidentiality, integrity, authenticity, and non-repudiability of business documents and transactions. These are listed as follows:

- *Confidentiality*: Information should be protected from prying eyes of unauthorised internal users and external hackers.

and from being intercepted during transmission on communication networks. The content should be made unintelligible to the attacker so that it is not decipherable by anyone who does not know the transformation algorithm.

- *Integrity*: On retrieval of a stored document or on receipt at the other end of a communication network, the information should appear exactly as was stored or sent. It should be possible to detect any modification, addition or deletion to the original content. Integrity also precludes information 're-play', i.e generation/re-transmission of a fresh copy of the data using the authorisation features of the earlier authentication.
- *Authenticity*: No entity should be able to masquerade as another entity. When information is retrieved or received it should be possible to verify whether it has indeed been sent by the entity claiming to be the originator. Similarly, it should also be possible to ensure that the message is delivered to the intended recipient.
- *Non-repudiability*: After sending/authorising a message, the sender should not be able to, at a later date, deny having done so. Similarly, the recipient of a message should not be able to deny receipt at a later date. Messages and message acknowledgments must be bound to their originators.

Implementing a security solution in an e-commerce environment therefore, necessitates an analysis of the risks the business is exposed to so that information infrastructure is protected from intentional and accidental destruction. In the case of some transactions, confidentiality might be a critical requirement whereas in others it may only be data integrity that is of paramount importance.

14.7 Cryptopgraphy-based Solutions

Implementation of technology solutions for all the security services listed above is based on cryptographic techniques.

Cryptography comprises encryption, i.e. the process of making information unintelligible to the unauthorised reader, and decryption, i.e. reversing encryption to make the information intelligible once again. Conventional cryptography uses a secret code or key to encrypt information. The same secret key is used to decrypt the encrypted information.

A simple encryption scheme could be one in which all alphabetic and numerical characters are shifted by a fixed number of positions in the encrypted text. If the characters are to be shifted by, say five places, then the result would be as follows:

Character	*Represented as*
A	F
B	G
C	H
V	A
W	B
X	C
y	D
Z	E

Using this encryption scheme, where the key is a 5-character shift, the plaintext message THIS IS A BOOK would be encrypted to read YMNX NX F GTTP, which would not be very easily decipherable to the casual reader.

Over time, many encryption systems have developed and with the increased availability and advancement of computing resources, the level of sophistication of these systems has also increased. While the most obvious application of these processes is in ensuring confidentiality of information, advances in the science of cryptography have also made possible the provision of other security services such as integrity, authentication and non-repudiation.

Cryptographic systems or cryptosystems are symmetric or asymmetric. The symmetric system is based on a single secret key which is shared by the parties engaging in secure communication. The asymmetric system hinges on the possession by these parties of a pair of keys—one private and the other public.

14.7.1 Symmetric Cryptosystems

Major commercial use of symmetric cryptosystems began in 1977 when the Data Encryption Standard (DES) was adopted as a United States Federal standard. DES and other symmetric cryptosystems work on the concept of a single key being shared between two communicating entities. Essentially, therefore, for every pair of partners engaging in secure communications, a new key has to be generated and securely maintained.

Since, in the symmetric system, the secret key is shared between two persons or entities, it is very important to be able to ensure the secure exchange of the secret key. However, if indeed such a secure channel existed, it would not be necessary to encrypt data in the first place. How to circumvent this will be discussed in the section on Asymmetric or Public Key Cryptosystems. Figure 14.1 illustrates the use of symmetric keys.

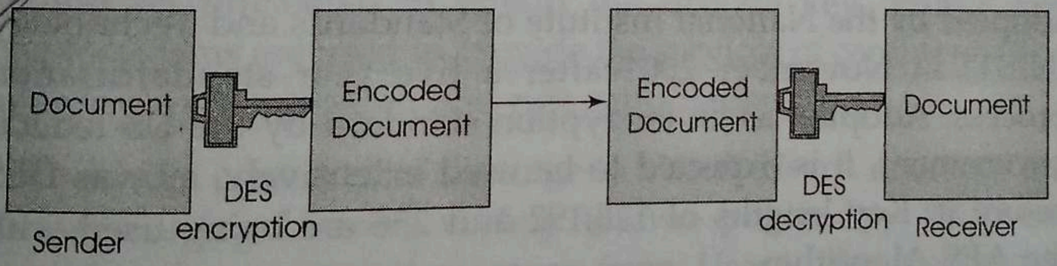

Fig. 14.1 *Symmetric keys*

Symmetric systems operate either in the block cipher or in the stream cipher mode. In the block cipher mode, the data to be encrypted is broken up into fixed size blocks. Each of these fixed size blocks is encrypted and on decryption is again presented back with data in blocks of the same size. The stream cipher mode can operate on data of any size and on encryption results in encrypted data of the same size as the plaintext data.

The DES cryptosystem operates in the block cipher mode. Data is encrypted in 64-bit blocks using a 56-bit key. After an initial permutation of the data bits, the result is passed through 16 rounds of processing using the 56-bit key. A final permutation generates the encrypted 64-bit data block. The decryption process is similar except that it is followed in the reverse order.

The strength of the encryption key is directly proportional to the key length, since a brute force attack using all possible combinations within the key length would yield the secret key. Increasing key length increases the strength but there is consequently a trade-off with the processing overhead and consequently the cost of key usage.

Triple-DES follows the same algorithm as DES, using three 56-bit keys. 64-bit data blocks are first encrypted using key1. The result is encrypted using key2 and again encrypted using key3.

Another popular cryptographic algorithm is the International Data Encryption Algorithm (IDEA) which uses 128-bit key for encryption. The Advanced Encryption Standard (AES), also known as Rijndael, operates in a block cipher mode. It was adopted by the National Institute of Standards and Technology (NIST) in November 2001 after a five year standardisation process. Adopted as an encryption standard by the US federal government, it is expected to be used extensively, as was DES before it. Key lengths of 128,192 and 256 are being used with the AES Algorithm.

While the solutions presented above only provide data confidentiality, symmetric cryptosystems can also be used to support the requirements of message integrity and data authentication. This is done through the secret key based generation of a checksum from the contents of the original data. The checksum is sent along with the data. Any modifications made to the data en route will become known to the receiver since the new checksum created from the received data using the shared secret key will not match with the checksum which has been sent by the originator. The Message Authentication Code (MAC) is an integrity checksum standardised in 1986 for use by the banking and financial sector. MAC uses DES algorithm for generating the integrity checksum.

In order to counter the problems of lost or duplicate messages, unique Message Serial Numbers are incorporated cryptographically into the message so that there is no message 're-play' or dropped messages.

14.7.2 Asymmetric Cryptosystems

Asymmetric or public key cryptosystems are built around the possession of a pair of keys—a public key and a private key—by each entity wishing to engage in secure communication. While, as its name suggests, everyone knows the public key, only the owner knows the private key. The algorithm used to generate these keys is such that if either of the keys is used to encrypt a message, only the other corresponding key in the key pair will be able to decrypt it. Although these keys would then have to be related to one another, knowing the public key, it should be infeasible to obtain the private key. Public key cryptosystems are used to provide the services of confidentiality, integrity, authentication and non-repudiation.

To send a confidential message to UserB, UserA encrypts the message using UserB's widely known public key PKB. On receiving the encrypted message from UserA, the message is decrypted using UserB's private key SKB. Confidentiality is assured since the private key would have been carefully protected by UserB. Any third party, without knowledge of UserB's private key would not be able to decipher the encrypted message. This is explained through Fig. 14.2(a).

For UserB to receive an authenticated message from UserA, the message is encrypted using UserA's private key SKA. At the recipient end, the encrypted message is decrypted by UserA' public key PKA which is widely known. On validation, the message is assured to have been sent by UserA since the corresponding private key is held securely by UserA. Any third party would also be able to verify the authenticity since the public key is known to everyone. Authentication is explained through Fig. 14.2(b).

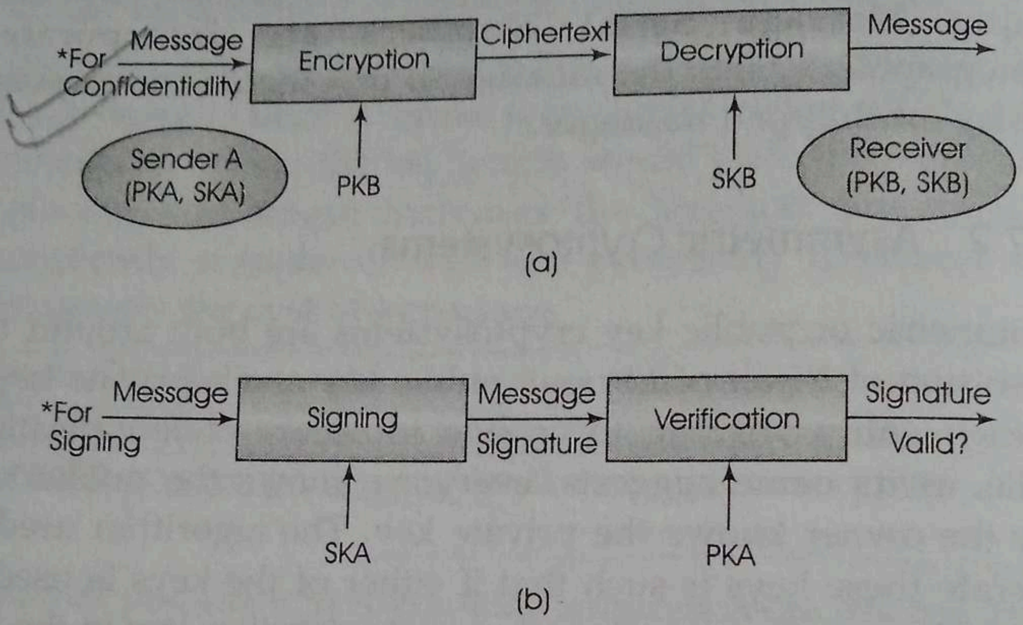

Fig. 14.2 *Asymmetric algorithm*

In order to achieve both confidentiality and authenticity, the message can first be authenticated using the originators private key and the authenticated message can then be made confidential by encrypting with the recipients public key.

Since the secret key does not have to be shared between communicating entities in public key cryptosystems, the chances of the secret key being compromised is reduced. Although theoretically, the public key can be used to determine the corresponding private key, its infeasibility (within time and cost constraints) is built over the 'difficulty' of solving certain mathematical problems.

Digital signatures are created and verified by using public key cryptography. An algorithm generates the two different and related keys: a private key and a public key. The private key is used to sign the document. The result of this encryption process, which depends on the private key and the contents of the document, is the digital signature. The corresponding public key is used to verify the digital signature, because it alone can decrypt the message.

The application of the signer's private key on a message generates the digital signature. For every new message, the digital signature of the same person will be different. Therefore, if the contents of a message are altered, the digital signature will not match when verified by the recipient with the signer's public key. The integrity of the message is thus ensured. Since the verification of the digital signature can take place only through the public key of the signer, the identity of the signer is established. However, the identity of the signer must be bound to his or her public key by some entity in the physical space. The onus of verification of the identity of the individual rests with Certifying Authorities.

As noted earlier, knowing the public key, one cannot compute the corresponding private key belonging to the owner of the key pair. This is because it is computationally infeasible to derive a user's, private key from his public key. 'Computational infeasibility' is a relative concept based on the value of the data protected, the computing overhead required to protect it, the length of time it needs to be protected, and the cost and time required to attack the data, with such factors assessed both currently and in the light of future technological advance. The user can keep the private key on a smart card for access through a PIN or biometric identification such as a fingerprint or a retinal print. Digital signatures are unforgeable as long as the private key is not compromised.

14.7.3 The RSA Algorithm

One of the most popular and widely used public key cryptosystems is the RSA algorithm, developed in 1978 by Ron Rivest, Adi Shamir and Len Adleman of the Massachusetts Institute of Technology (MIT).

Two large prime numbers p and q are randomly chosen and their product $N = p \times q$ is computed. From the product $(p - 1) \times (q - 1)$, a number, e, is chosen such that e is relatively prime to $(p - 1)(q - 1)$ i.e both $(p - 1)$ and $(q - 1)$ do not have any common factors with e. Similarly d is chosen such that d satisfies

$$de = 1 \bmod (p - 1)(q - 1)$$

i.e. $de - 1$ is divisible by $(p - 1)\ (q - 1)$

The public key is then (N, e) while the private key is (N, d)

In order to encrypt a message M using the public key (N, e), the value of M^e mod N is calculated to produce the encrypted message E. For decrypting, calculation of E^d mod N yields the original message M.

14.8 Digital Signatures

Digital signatures are used not only to verify the authenticity of the message and the claimed identity of the sender, but also to verify message integrity. The recipient, however, should not be able to use the received digital signature to falsely 'sign' messages on behalf of the original sender.

Using the RSA cryptosystem, a message is encrypted with the sender's private key to generate the 'signature'. The message is then sent to the destination along with this signature. The recipient decrypts the signature using the sender's public key, and if the result matches with the copy of the message received, the recipient can be sure that the message was sent by the claimed originator and that the message has not been modified during transmission, since only the originator is in possession of the corresponding encryption key. Figure 14.3 illustrates the implementation of Digital Signatures.

Although this is a highly secure way of digitally signing messages, it generates a large processing overhead. The size of the signature is the same as that of the original message, thereby resulting in a 100 percent increase in the data that is to be

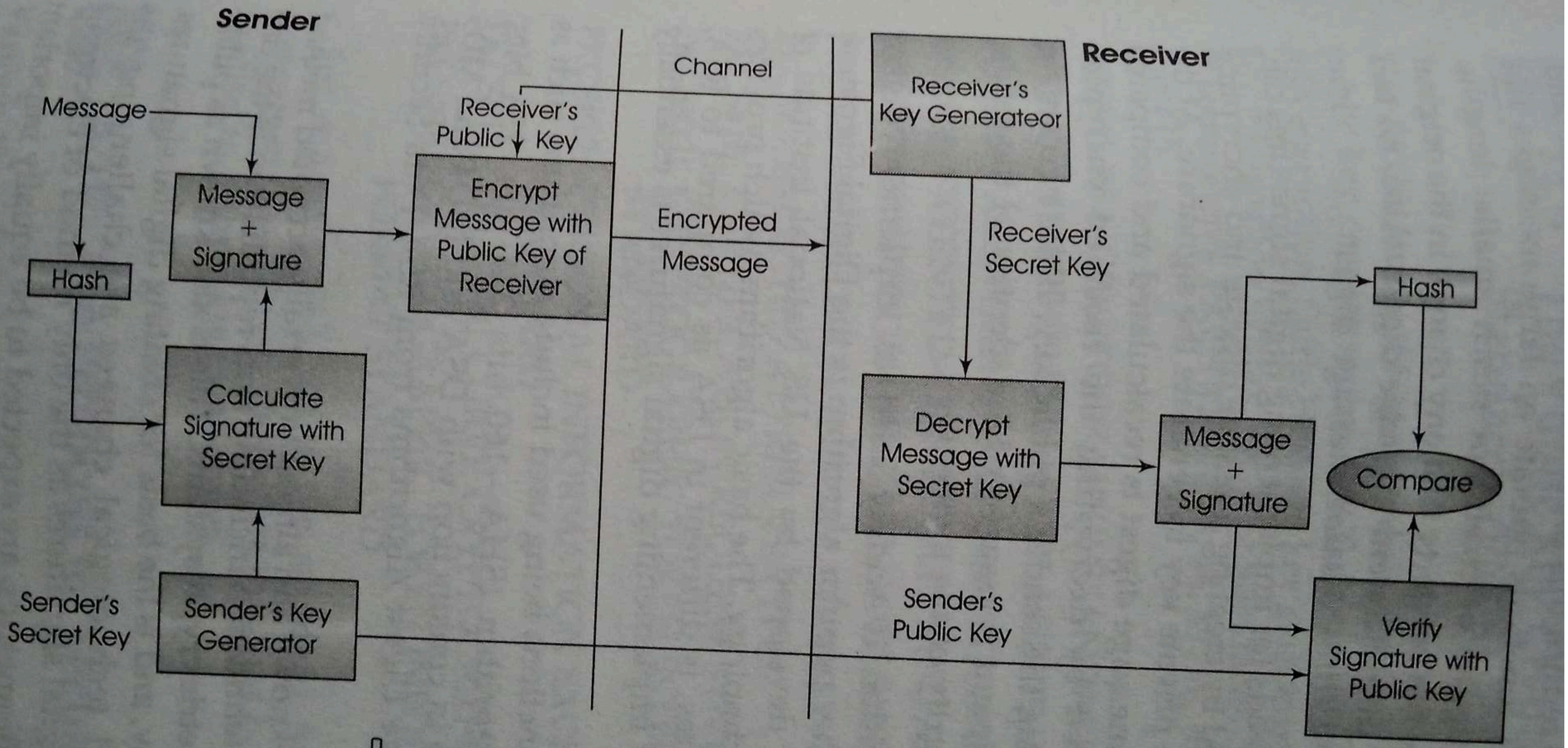

Fig. 14.3 *Implementation of digital signatures*

handled. In order to reduce this processing load, hash functions are employed. Hash functions operate on large messages and generate message digests of fixed but much smaller lengths. These functions have the property that any change in the original message will result in a different message digest. Also, no two messages would result in the same message digest.

Processing overheads for implementing digital signatures can, therefore, be reduced by creating message digests and encrypting this digest with the private key to generate the signature. At the receiving end, the message digest is recalculated and compared with that generated by decrypting the received encrypted message digest using the sender's public key. If the two are the same, then the recipient is assured of the identity of the sender as well as the integrity of the message.

The RSA algorithm is widely used to implement digital signatures. The other popular algorithm is the Digital Signature Algorıthm (DSA) developed by the US National Institute of Standards and Technology. The basic algorithm, which provides the security features, is different in DSA as compared to RSA, but the method of implementing digital signatures is essentially the same.

The hashing functions being used include algorithms such as Secure Hash Algorithm (SHA—160 bits 224 bits, 256 bits, 384 bits, 512 bits) in conjunction with DSA and MD4 and MD5 (128 bits) Message Digest Algorithms from RSA Data Security Inc.

The most well-known and almost universally accepted method of electronic authentication including non-repudiation is the one based on asymmetric cryptosystems. This is also known as public key cryptography, and is the basis for creating digital signatures. However, rapid technological changes are challenging the supremacy of digital signatures as the only method of electronic authentication. Biometrics, are expected to be equally important in authentication for access control in the years to come.