**UNIT V**

Network security: Network security consists of the policies and practices adopted to prevent and monitor unauthorized access, misuse, modification, or denial of a computer network and network- accessible resources.

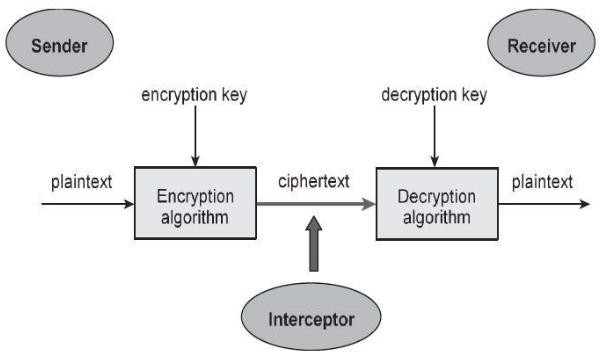
# Cryptography

**Cryptology**, the study of cryptosystems, can be subdivided into two branches.

* + - Cryptography
    - Cryptanalysis

**Cryptography** comes from the Greek words for ''secret writing”. It is the branch of science that deals with encryption and decryption of data to provide information security.

**Cryptanalysis** The art and science of breaking the cipher text is known as cryptanalysis.



**Cryptosystem or Cipher:** A cryptosystem is an implementation of cryptographic techniques and their accompanying infrastructure to provide information security services. A cryptosystem is also referred to as a cipher system.

## Components of a Cryptosystem:

The various components of a basic cryptosystem are as follows −

* + - **Plaintext.** It is the data to be protected during transmission.
    - **Encryption Algorithm**. It is a cryptographic algorithm that takes plaintext and an encryption key as input and produces a ciphertext.
    - **Ciphertext.** It is the scrambled version of the plaintext produced by the encryption algorithm using a specific the encryption key.
    - **Decryption Algorithm,** It is a cryptographic algorithm that takes a ciphertext and a decryption key as input, and outputs a plaintext. The decryption algorithm essentially reverses the encryption algorithm and is thus closely related to it.
    - **Encryption Key**. It is a value that is known to the sender. The sender inputs the encryption key into the encryption algorithm along with the plaintext in order to compute the ciphertext.
    - **Decryption Key.** It is a value that is known to the receiver. The decryption key is related to the encryption key, but is not always identical to it. The receiver inputs the decryption key into the decryption algorithm along with the ciphertext in order to compute the plaintext.

For a given cryptosystem, a collection of all possible decryption keys is called a **key space**.

**An interceptor or intruder or an attacker** is an unauthorized entity who attempts to determine the plaintext. He can see the ciphertext and may know the decryption algorithm. He, however, must never know the decryption key.

## Security Services of Cryptography:

The primary objective of using cryptography is to provide the following fundamental information security services.

* + - **Confidentiality**: Confidentiality is the fundamental security service provided by cryptography. It is a security service that keeps the information from an unauthorized person. It is sometimes referred to as privacy or secrecy. Confidentiality can be achieved through numerous means starting from physical securing to the use of mathematical algorithms for data encryption.
    - **Data Integrity:** It is security service that deals with identifying any alteration to the data. The data may get modified by an unauthorized entity intentionally or accidently. Integrity service confirms that whether data is intact or not since it was last created, transmitted, or stored by an authorized user.
    - **Authentication:** Authentication provides the identification of the originator. It confirms to the receiver that the data received has been sent only by an identified and verified sender.
    - **Non-repudiation:** It is a security service that ensures that an entity cannot refuse the ownership of a previous commitment or an action. It is an assurance that the original creator of the data cannot deny the creation or transmission of the said data to a recipient or third party.
    - **Availability:** The data should be available when ever requested.

Encryption methods have historically been divided into two categories: substitution ciphers and transposition ciphers. All of these systems are based on symmetric key encryption scheme.

## Substitution Ciphers

In a substitution cipher each letter or group of letters is replaced by another letter or group of letters to disguise it. One of the oldest known ciphers is the Caesar cipher For example, A.B…..Y.Z and Z.Y……B.A are two obvious permutation of all the letters in alphabet. Permutation is nothing but a jumbled up set of alphabets. With 26 letters in alphabet, the possible permutations are 26! (Factorial of 26) which is equal to 4x1026. The sender and the receiver may choose any one of these possible permutation as a ciphertext alphabet. This permutation is the secret key of the scheme.

Process of Simple Substitution Cipher

* + Write the alphabets A, B, C,...,Z in the natural order.
  + The sender and the receiver decide on a randomly selected permutation of the letters of the alphabet.
  + Underneath the natural order alphabets, write out the chosen permutation of the letters of the alphabet. For encryption, sender replaces each plaintext letters by substituting the permutation letter that is directly beneath it in the table. This process is shown in the following illustration. In this example, the chosen permutation is K,D, G, ..., O. The plaintext ‘point’ is encrypted to ‘MJBXZ’.

Here is a jumbled Ciphertext alphabet, where the order of the ciphertext letters is a key.



* + On receiving the ciphertext, the receiver, who also knows the randomly chosen permutation, replaces each ciphertext letter on the bottom row with the corresponding plaintext letter in the top row. The ciphertext ‘MJBXZ’ is decrypted to ‘point’.

**Monoalphabetic and Polyalphabetic Cipher**

Monoalphabetic cipher is a substitution cipher in which for a given key, the cipher alphabet for each plain alphabet is fixed throughout the encryption process. For example, if ‘A’ is encrypted as ‘D’, for any number of occurrence in that plaintext, ‘A’ will always get encrypted to ‘D’.

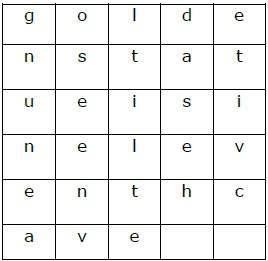
Polyalphabetic Cipher is a substitution cipher in which the cipher alphabet for the plain alphabet may be different at different places during the encryption process.

## Transposition Ciphers

It is another type of cipher where the order of the alphabets in the plaintext is rearranged to create the ciphertext. The actual plaintext alphabets are not replaced.

An example is a ‘simple columnar transposition’ cipher where the plaintext is written horizontally with a certain alphabet width. Then the ciphertext is read vertically as shown.

For example, the plaintext is “golden statue is in eleventh cave” and the secret random key chosen is “five”. We arrange this text horizontally in table with number of column equal to key value. The resulting text is shown below.



The ciphertext is obtained by reading column vertically downward from first to last column. The ciphertext is ‘gnuneaoseenvltiltedasehetivc’.

To decrypt, the receiver prepares similar table. The number of columns is equal to key number. The number of rows is obtained by dividing number of total ciphertext alphabets by key value and rounding of the quotient to next integer value.

The receiver then writes the received ciphertext vertically down and from left to right column. To obtain the text, he reads horizontally left to right and from top to bottom row.

## One-Time Pads

The one-time pad is a long sequence of random letters. These letters are combined with the plaintext message to produce the ciphertext. To decipher the message, a person must have a copy of the one- time pad to reverse the process. A one-time pad should be used only once (hence the name) and then destroyed. This is the first and only encryption algorithm that has been proven to be unbreakable.

To encipher a message, you take the first letter in the plaintext message and add it to the first random letter from the one-time pad. For example, suppose you are enciphering the letter S (the 19th letter of the alphabet) and the one-time pad gives you C (3rd letter of the alphabet). You add the two letters and subtract 1. When you add S and C and subtract 1, you get 21 which is U. Each letter is enciphered in this method, with the alphabet wrapping around to the begining if the addition results in a number beyond 26 (Z).

To decipher a message, you take the first letter of the ciphertext and subtract the first random letter from the one-time pad. If the number is negative you wrap around to the end of the alphabet.

Example

plaintext : SECRETMESSAGE one-time pad: CIJTHUUHMLFRU ciphertext : UMLKLNGLEDFXY

### *Quantum Cryptography*

Quantum cryptography is an emerging technology in which two parties can secure network communications by applying the phenomena of quantum physics. The security of these transmissions is based on the inviolability of the laws of quantum mechanics. The quantum cryptography relies on two important elements of quantum mechanics the Heisenberg Uncertainty principle and the principle of photon polarization.

Polarization can be used to polarize (pass through a filter) a photon so that it has a particular spin, vertical or horizontal or diagonal. Polarization of a photon is performed using polarization filters.

Heisenberg’s Uncertainty Principle, which states that it is impossible to measure together the speed and position of a particle with highest accuracy, and its state will change when measured. In other words, if an eavesdropper intercepts the transmitted photons and passes it through its polarizer, if it is wrong it will make the receiver get the wrong photon. Hence the interception of communication will get detected.

## Two Fundamental Cryptographic Principles

1. **Redundancy:** The first principle is that all encrypted messages must contain some redundancy, that is, information not needed to understand the message.

Cryptographic principle 1: “Messages must contain some redundancy.”

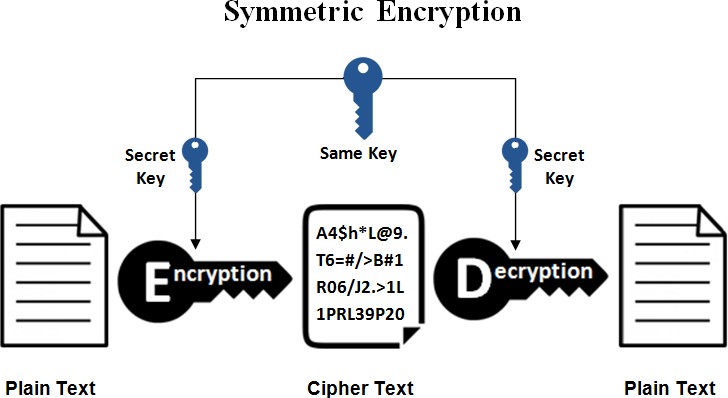
In other words, upon decrypting a message, the recipient must be able to tell whether it is valid by simply inspecting it and perhaps performing a simple computation. This redundancy is needed to prevent active intruders from sending garbage and tricking the receiver into decrypting the garbage and acting on the ''plaintext.'' However, this same redundancy makes it much easier for passive intruders to break the system.

1. **Freshness:** The second cryptographic principle is that some measures must be taken to ensure that each message received can be verified as being fresh, that is, sent very recently. This measure is needed to prevent active intruders from playing back old messages.

Cryptographic principle 2: “Some method is needed to foil replay attacks.”

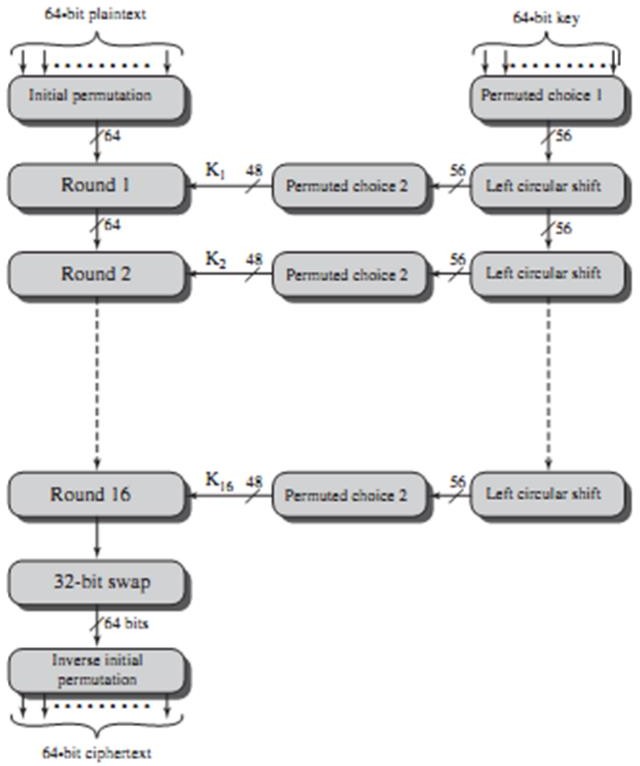
One such measure is including in every message a timestamp valid only for, say, 10 seconds. The receiver can then just keep messages around for 10 seconds, to compare newly arrived messages to previous ones to filter out duplicates. Messages older than 10 seconds can be thrown out, since any replays sent more than 10 seconds later will be rejected as too old.

# Symmetric Key Algorithms



This is the simplest kind of encryption that involves only one secret key to cipher and decipher information. Symmetrical encryption is an old and best-known technique. It uses a secret key that can either be a number, a word or a string of random letters. It is a blended with the plain text of a message to change the content in a particular way. The sender and the recipient should know the secret key that is used to encrypt and decrypt all the messages. Blowfish, AES, RC4, DES, RC5, and RC6 are examples of symmetric encryption. The most widely used symmetric algorithm is AES-128, AES-192, and AES-256. The main disadvantage of the symmetric key encryption is that all parties involved have to exchange the key used to encrypt the data before they can decrypt it. Kerckhoff stated that a cryptographic system should be secure even if everything about the system, except the key, is public knowledge.

## DES Algorithm



There are two inputs to the encryption function: the plaintext to be encrypted and the key. In this case, the plaintext must be 64 bits in length and the key is 56 bits in length (Actually, the function expects a 64-bit key as input. However, only 56 of these bits are ever used; the other 8 bits can be used as parity bits or simply set arbitrarily). DES follows Fiestel Structure and has 16 rounds of operation. The operation in each round can be seen in the below figure.

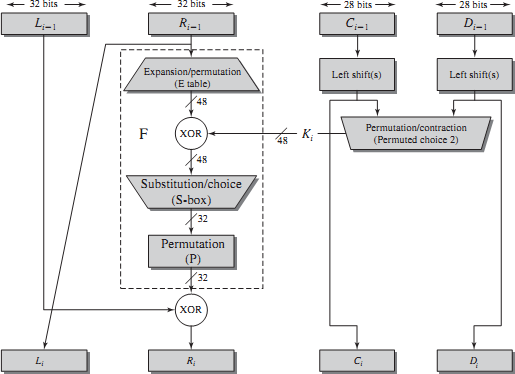


Figure: Operation in a round

## Triple DES

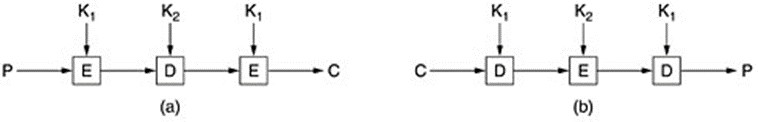


Figure: (a) Triple encryption using DES. (b) Decryption.

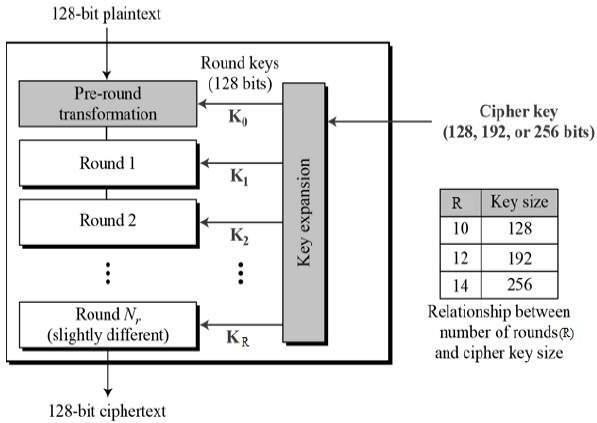
The encryption-decryption process is as follows −

* + Encrypt the plaintext blocks using single DES with key K1.
  + Now decrypt the output of step 1 using single DES with key K2.
  + Finally, encrypt the output of step 2 using single DES with key K3.
  + The output of step 3 is the ciphertext.
  + Decryption of a ciphertext is a reverse process. User first decrypt using K3, then encrypt with K2, and finally decrypt with K1.

Due to this design of Triple DES as an encrypt–decrypt–encrypt process, it is possible to use a 3TDES (hardware) implementation for single DES by setting K1, K2, and K3 to be the same value. This provides backwards compatibility with DES. Triple DES systems are significantly more secure than single DES, but these are clearly a much slower process than encryption using single DES.

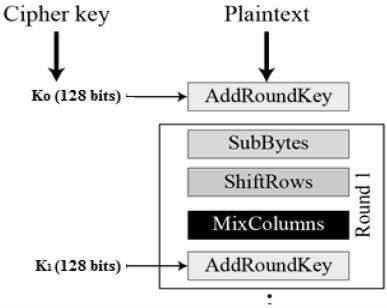
## AES

The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES). The features of AES are as follows −

* + Symmetric key symmetric block cipher
  + 128-bit data, 128/192/256-bit keys
  + Stronger and faster than Triple-DES
  + Provide full specification and design details
  + Software implementable in C and Java Operation of AES

AES is an iterative rather than Feistel cipher. It is based on ‘substitution–permutation network’. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations). Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix. Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

**Encryption Process:**



Here, we restrict to description of a typical round of AES encryption. Each round comprise of four sub-processes.

## Byte Substitution (SubBytes)

The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

## Shiftrows

Each of the four rows of the matrix is shifted to the left. Any entries that ‘fall off’ are re-inserted on the right side of row. Shift is carried out as follows −

* First row is not shifted.
* Second row is shifted one (byte) position to the left.
* Third row is shifted two positions to the left.
* Fourth row is shifted three positions to the left.
* The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

## MixColumns

Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

## Addroundkey

The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

**Decryption Process:**

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order −

* Add round key
* Mix columns
* Shift rows
* Byte substitution

Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms needs to be separately implemented, although they are very closely related.

## Cipher Modes

1. **Cryptanalysis**

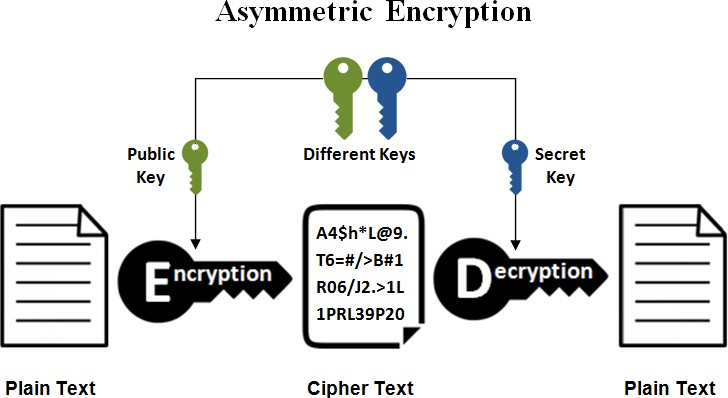
**Differential cryptanalysis**: This technique can be used to attack any block cipher. It works by beginning with a pair of plaintext blocks that differ in only a small number of bits and watching carefully what happens on each internal iteration as the encryption proceeds. In many cases, some bit patterns are much more common than other patterns, and this observation leads to a probabilistic attack.

**Linear cryptanalysis**: It can break DES with only 243 known plaintexts. It works by XORing certain bits in the plaintext and ciphertext together and examining the result for patterns. When this is done repeatedly, half the bits should be 0s and half should be 1s.

**Using analysis of the electrical power consumption to find secret keys**: Computers typically use 3 volts to represent a 1 bit and 0 volts to represent a 0 bit. Thus, processing a 1 takes more electrical energy than processing a 0. If a cryptographic algorithm consists of a loop in which the key bits are processed in order, an attacker who replaces the main n-GHz clock with a slow (e.g., 100-Hz) clock and puts alligator clips on the CPU's power and ground pins, can precisely monitor the power consumed by each machine instruction. From this data, deducing the key is surprisingly easy.

**Timing analysis:** Cryptographic algorithms are full of if statements that test bits in the round keys. If the then and else parts take different amounts of time, by slowing down the clock and seeing how long various steps take, it may also be possible to deduce the round keys. Once all the round keys are known, the original key can usually be computed.

# Public Key Algorithms



Asymmetrical encryption is also known as public key cryptography. Asymmetric encryption uses two keys to encrypt a plain text. Secret keys are exchanged over the Internet or a large network. It ensures that malicious persons do not misuse the keys. It is important to note that anyone with a secret key can decrypt the message and this is why asymmetrical encryption uses two related keys to boosting security. A public key is made freely available to anyone who might want to send you a message. The second private key is kept a secret so that you can only know.

A message that is encrypted using a public key can only be decrypted using a private key, while also, a message encrypted using a private key can be decrypted using a public key. Security of the public key is not required because it is publicly available and can be passed over the internet. Asymmetric key has a far better power in ensuring the security of information transmitted during communication.

Asymmetric encryption is mostly used in day-to-day communication channels, especially over the Internet. Popular asymmetric key encryption algorithm includes EIGamal, RSA, DSA, Elliptic curve techniques, PKCS.

## i. RSA

The system was invented by three scholars Ron Rivest, Adi Shamir, and Len Adleman and hence, it is termed as RSA cryptosystem.

The RSA method is based on some principles from number theory. We will now summarize how to use the method; for details, consult the paper.

1. Choose two large primes, p and q (typically 1024 bits).
2. Compute n = p x q and z = (p - 1) x (q - 1).
3. Choose a number relatively prime to z and call it d.
4. Find e such that e x d = 1 mod z.

To encrypt a message, P, compute C = Pe (mod n). To decrypt C, compute P = Cd (mod n).

The security of RSA depends on the strengths of two separate functions. The RSA cryptosystem is most popular public-key cryptosystem strength of which is based on the practical difficulty of factoring the very large numbers.

**Encryption Function** − It is considered as a one-way function of converting plaintext into ciphertext and it can be reversed only with the knowledge of private key d.

**Key Generation** − The difficulty of determining a private key from an RSA public key is equivalent to factoring the modulus n. An attacker thus cannot use knowledge of an RSA public key to determine an RSA private key unless he can factor n. It is also a one way function, going from p & q values to modulus n is easy but reverse is not possible.

# Digital Signatures

It is a system by which one party can send a signed message to another party in such a way that the following conditions hold:

1. The receiver can verify the claimed identity of the sender.
2. The sender cannot later repudiate the contents of the message.
3. The receiver cannot possibly have concocted the message himself.

## Symmetric-Key Signatures

One approach to digital signatures is to have a central authority that knows everything and whom everyone trusts, say Big Brother (*BB*)*.* Each user then chooses a secret key and carries it by hand to *BB*'s office. Thus, only Alice and *BB* know Alice's secret key, *KA*, and so on.

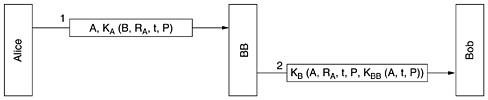


Figure: Digital signatures with Big Brother.

## Public-Key Signatures

A structural problem with using symmetric-key cryptography for digital signatures is that everyone has to agree to trust Big Brother. Furthermore, Big Brother gets to read all signed messages.

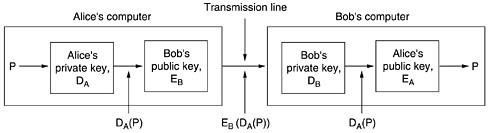


Figure: Digital signatures using public-key cryptography.

## Message Digests

This scheme is based on the idea of a one-way hash function that takes an arbitrarily long piece of plaintext and from it computes a fixed-length bit string. This hash function, *MD*, often called a **message digest**, has four important properties:

* 1. Given *P*, it is easy to compute *MD*(*P*).
  2. Given *MD*(*P*), it is effectively impossible to find *P*.
  3. Given *P* no one can find *P*' such that *MD* (*P*') = *MD*(*P*).
  4. A change to the input of even 1 bit produces a very different output.

To meet criterion 3, the hash should be at least 128 bits long, preferably more. To meet criterion 4, the hash must mangle the bits very thoroughly, not unlike the symmetric-key encryption algorithms we have seen. Computing a message digest from a piece of plaintext is much faster than encrypting that plaintext with a public-key algorithm, so message digests can be used to speed up digital signature algorithms.

### *MD5:*

**MD5** is the fifth in a series of message digests designed by Ronald Rivest. It operates by mangling bits in a sufficiently complicated way that every output bit is affected by every input bit. Very briefly, it starts out by padding the message to a length of 448 bits (modulo 512). Then the original length of the message is appended as a 64-bit integer to give a total input whose length is a multiple of 512 bits. The last precomputation step is initializing a 128-bit buffer to a fixed value.

Each round takes a 512-bit block of input and mixes it thoroughly with the 128-bit buffer. For good measure, a table constructed from the sine function is also thrown in. Four rounds are performed per input block. This process continues until all the input blocks have been consumed. The contents of the 128bit buffer form the message digest.

### *SHA-1:*

SHA1 algorithm consists of 6 tasks:

Task 1. Appending Padding Bits. The original message is "padded" (extended) so that its length (in bits) is congruent to 448, modulo 512. The padding rules are:

* The original message is always padded with one bit "1" first.
* Then zero or more bits "0" are padded to bring the length of the message up to 64 bits fewer than a multiple of 512.
* Task 2. Appending Length. 64 bits are appended to the end of the padded message to indicate the length of the original message in bytes. The rules of appending length are:
* The length of the original message in bytes is converted to its binary format of 64 bits. If overflow happens, only the low-order 64 bits are used.
* Break the 64-bit length into 2 words (32 bits each).
* The low-order word is appended first and followed by the high-order word.

Task 3. Preparing Processing Functions. SHA1 requires 80 processing functions defined as:

f(t;B,C,D) = B XOR C XOR D (20 <= t <= 39)

f(t;B,C,D) = (B AND C) OR (B AND D) OR (C AND D) (40 <= t <= 59) f(t;B,C,D) = B XOR C XOR D (60 <= t <= 79)

( 0 <= t <= 19)

f(t;B,C,D) = (B AND C) OR ((NOT B) AND D)

Task 4. Preparing Processing Constants. SHA1 requires 80 processing constant words defined as:

K(t) = 0x5A827999 ( 0 <= t <= 19) K(t) = 0x6ED9EBA1 (20 <= t <= 39) K(t) = 0x8F1BBCDC (40 <= t <= 59)

K(t) = 0xCA62C1D6 (60 <= t <= 79)

Task 5. Initializing Buffers. SHA1 algorithm requires 5 word buffers with the following initial values:

H0 = 0x67452301 H1 = 0xEFCDAB89 H2 = 0x98BADCFE H3 = 0x10325476

H4 = 0xC3D2E1F0

Task 6. Processing Message in 512-bit Blocks. This is the main task of SHA1 algorithm, which loops through the padded and appended message in blocks of 512 bits each. For each input block, a number of operations are performed.

## The Birthday Attack

One might think that it would take on the order of 2*m* operations to subvert an *m*-bit message digest. In fact, 2*m/*2 operations will often do using the **birthday attack,** an approach published by Yuval (1979) If there is some mapping between inputs and outputs with *n* inputs (people, messages, etc.) and *k* possible outputs (birthdays, message digests, etc.), there are *n*(*n* - 1)*/*2 input pairs. If *n*(*n* - 1)*/*2 *>k*, the chance of having at least one match is pretty good. Thus, approximately, a match is likely for . This result means that a 64-bit message digest can probably be broken by generating about 232 messages and looking for two with the same message digest.

**5.5 Management of Public keys**

## Certificates

An attachment to an electronic message used for security purposes. The most common use of a digital certificate is to verify that a user sending a message is who he or she claims to be, and to provide the receiver with the means to encode a reply.

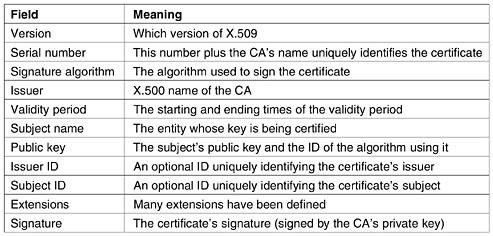
An individual wishing to send an encrypted message applies for a digital certificate from a Certificate Authority (CA). The CA issues an encrypted digital certificate containing the applicant's public key and a variety of other identification information. The CA makes its own public key readily available through print publicity or perhaps on the Internet.

The recipient of an encrypted message uses the CA's public key to decode the digital certificate attached to the message, verifies it as issued by the CA and then obtains the sender's public key and identification information held within the certificate. With this information, the recipient can send an encrypted reply.

The most widely used standard for digital certificates is X.509.

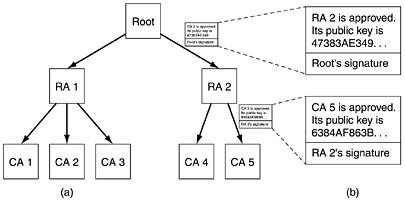
## X.509

X.509 is a way to describe certificates. The primary fields in a certificate are listed in Fig.



1. **PKI** (**Public Key Infrastructure**). A PKI has multiple components, including users, CAs, certificates, and directories. What the PKI does is provide a way of structuring these components and define standards for the various documents and protocols. A particularly simple form of PKI is a hierarchy of Cas. The top-level CA, the root, certifies second-level CAs, which we call **RA**s (**Regional Authorities**) because they might cover some geographic region, such as a country or continent. These in turn certify the real CAs, which issue the X.509 certificates to organizations and individuals. When the root authorizes a new RA, it generates an X.509 certificate stating that it has approved the RA, includes the new RA's public key in it, signs it, and hands it to the RA. Similarly, when an RA approves a new CA, it produces and signs a certificate stating its approval and containing the CA's public key.

Figure: (a) A hierarchical PKI. (b) A chain of certificates.



### *Directories*

Directory servers job is managing X.509 certificates. Such directories could provide lookup services by using properties of the X.500 names.

### *Revocation*

CA may periodically issue a **CRL** (**Certificate Revocation List**) giving the serial numbers of all certificates that it has revoked. Since certificates contain expiry times, the CRL need only contain the serial numbers of certificates that have not yet expired. Once its expiry time has passed, a certificate is automatically invalid, so no distinction is needed between those that just timed out and those that were actually revoked. In both cases, they cannot be used any more.

# 5.6 Communication Security

1. **IPSec**

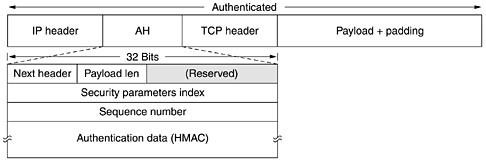
Internet Protocol Security (IPsec) is a network protocol suite that authenticates and encrypts the packets of data sent over a network. IPsec supports network-level peer authentication, data-origin authentication, data integrity, data confidentiality (encryption), and replay protection.

IPsec is an end-to-end security scheme operating in the Internet Layer of the Internet Protocol Suite, while some other Internet security systems in widespread use, such as Transport Layer Security (TLS) and Secure Shell (SSH), operate in the upper layers at the Transport Layer (TLS) and the Application layer (SSH). IPsec can automatically secure applications at the IP layer.

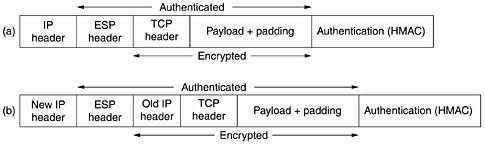
# Security architecture

The IPsec suite is an open standard. IPsec uses the following protocols to perform various functions:

* 1. Authentication Headers (AH) provides connectionless data integrity and data origin authentication for IP datagrams and provides protection against replay attacks.



* 1. Encapsulating Security Payloads (ESP) provides confidentiality, data-origin authentication, connectionless integrity, an anti-replay service



## Figure: (a) ESP in transport mode. (b) ESP in tunnel mode.

* 1. Security Associations (SA) provides the bundle of algorithms and data that provide the parameters necessary for AH and/or ESP operations. The Internet Security Association and Key Management Protocol (ISAKMP) provides a framework for authentication and key exchange, with actual authenticated keying material.

## Firewalls

A firewall is a network security system designed to prevent unauthorized access to or from a private network. Firewalls can be implemented as both hardware and software, or a combination of both. Network firewalls are frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet, especially intranets. All messages entering or leaving the intranet pass through the firewall, which examines each message and blocks those that do not meet the specified security criteria.

There are several types of firewall techniques that will prevent potentially harmful information from getting through:

* Packet Filter: Looks at each packet entering or leaving the network and accepts or rejects it based on user-defined rules. Packet filtering is fairly effective and transparent to users, but it is difficult to configure. In addition, it is susceptible to IP spoofing.
* Application Gateway: Applies security mechanisms to specific applications, such as FTP and Telnet servers. This is very effective, but can impose a performance degradation.
* Circuit-level Gateway: Applies security mechanisms when a TCP or UDP connection is established. Once the connection has been made, packets can flow between the hosts without further checking.
* Proxy Server: Intercepts all messages entering and leaving the network. The proxy server effectively hides the true network addresses.

## VPN

VPN technology was developed to allow remote users and branch offices to securely access corporate applications and other resources. To ensure security, data would travel through secure tunnels and VPN users would use authentication methods - including passwords, tokens and other unique identification methods - to gain access to the VPN.

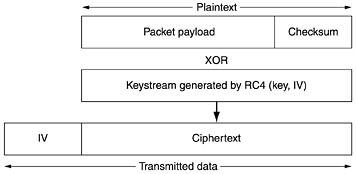
A VPN is created by establishing a virtual point-to-point connection through the use of dedicated connections, virtual tunneling protocols, or traffic encryption. A VPN available from the public Internet can provide some of the benefits of a wide area network (WAN). From a user perspective, the resources available within the private network can be accessed remotely.

## Wireless Security

### *802.11 Security*

The 802.11 standard prescribes a data link-level security protocol called **WEP** (**Wired Equivalent Privacy**), which is designed to make the security of a wireless LAN as good as that of a wired LAN. When 802.11 security is enabled, each station has a secret key shared with the base station. How the keys are distributed is not specified by the standard. They could be preloaded by the manufacturer. They could be exchanged in advance over the wired network. Finally, either the base station or user machine could pick a random key and send it to the other one over the air encrypted with the other one's public key. Once established, keys generally remain stable for months or years.

## Figure: Packet encryption using WEP.



### *Bluetooth Security*

Bluetooth has three security modes, ranging from nothing at all to full data encryption and integrity control. As with 802.11, if security is disabled (the default), there is no security. Most users have security turned off until a serious breach has occurred; then they turn it on.

Bluetooth provides security in multiple layers. In the physical layer, frequency hopping provides a tiny bit of security, but since any Bluetooth device that moves into a piconet has to be told the frequency hopping sequence, this sequence is obviously not a secret. The real security starts when the newly- arrived slave asks for a channel with the master. The two devices are assumed to share a secret key set up in advance. In some cases, both are hardwired by the manufacturer. In other cases, one device has a hardwired key and the user has to enter that key into the other device as a decimal number. These shared keys are called **passkeys**.

### *WAP 2.0 Security*

WAP 2.0 standard protocols in all layers. Security is no exception. Since it is IP based, it supports full use of IPsec in the network layer. In the transport layer, TCP connections can be protected by TLS, an IETF standard we will study later in this chapter. Higher still, it uses HTTP client authentication, as defined in RFC 2617. Application-layer crypto libraries provide for integrity control and nonrepudiation.

# Authentication Protocols

## Authentication Based on a Shared Secret Key

This protocol is based on a principle found in many authentication protocols: one party sends a random number to the other, who then transforms it in a special way and then returns the result. Such protocols are called challenge-response protocols.

A, B are the identities of Alice and Bob.

Ri's are the challenges, where the subscript identifies the challenger. Ki are keys, where i indicates the owner.

KS is the session key.

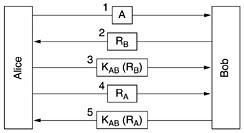
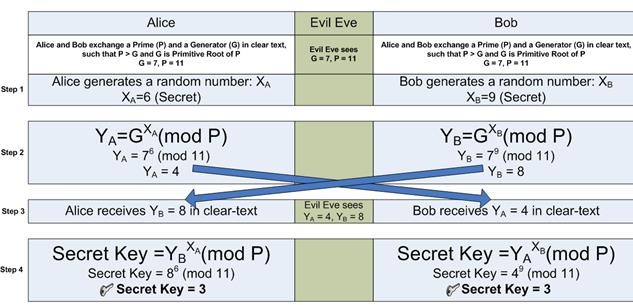
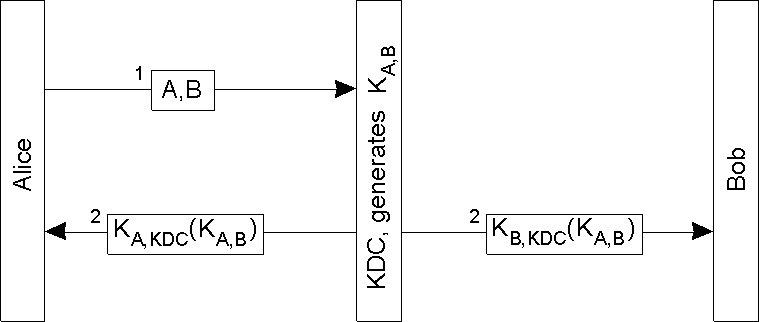


Figure: Two-way authentication using a challenge-response protocol.

## Establishing a Shared Key: The Diffie-Hellman Key Exchange



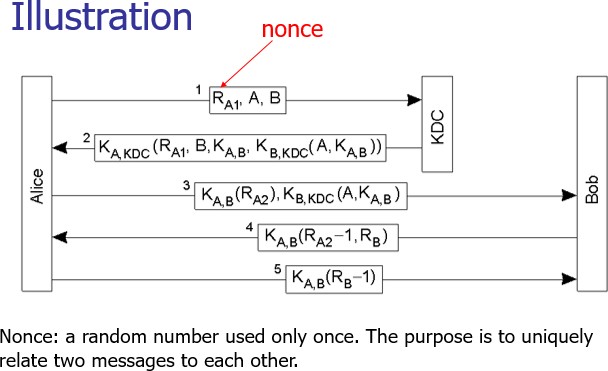
1. Authentication Using a Key Distribution Center



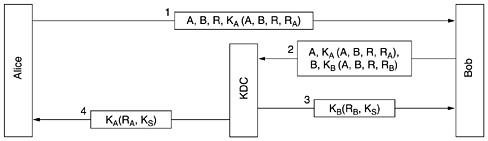
Needham-Schroeder Authentication Protocol

* + In early distributed systems (1974-84) it was difficult to protect the servers
  + Needham and Schroeder therefore developed an authentication and key-distribution protocol for use in a local network

◼ An early example of the care required to design a safe security protocol Introduced several design ideas including the use of *nonces*



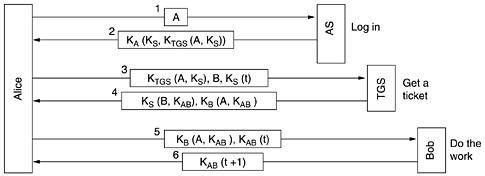
## Figure: The Otway-Rees authentication protocol (slightly simplified).



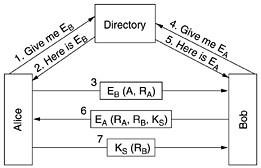
1. **Authentication Using Kerberos**

Kerberos involves three servers in addition to Alice (a client workstation):

* Authentication Server (AS): verifies users during login
* Ticket-Granting Server (TGS): issues ''proof of identity tickets''
* Bob the server: actually does the work Alice wants performed



## Authentication Using Public-Key Cryptography



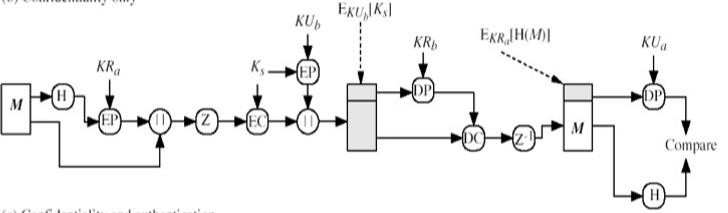
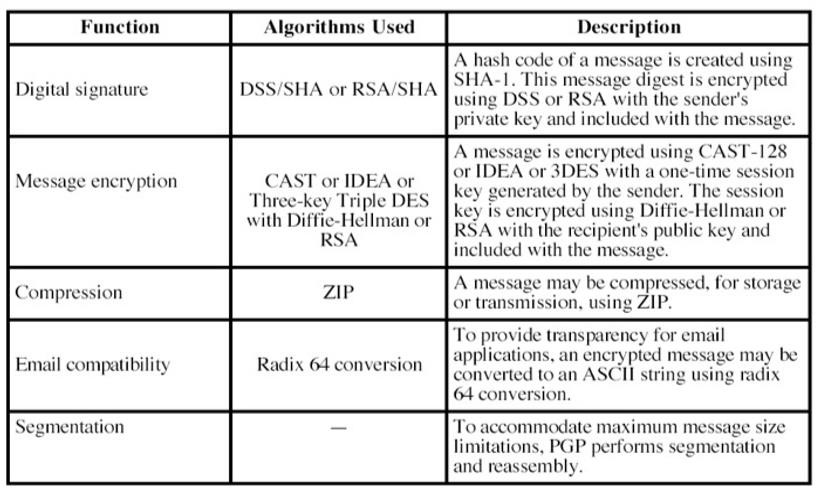
* 1. **E-mail Security**

1. **PGP**

Pretty Good Privacy (PGP) is an encryption program that provides cryptographic privacy and authentication for data communication. PGP is used for signing, encrypting, and decrypting texts, e- mails, files, directories, and whole disk partitions and to increase the security of e-mail communications.

PGP consists of the following ﬁve services:

* 1. Authentication
  2. Conﬁdentiality
  3. Compression
  4. E-mail compatibility
  5. Segmentation



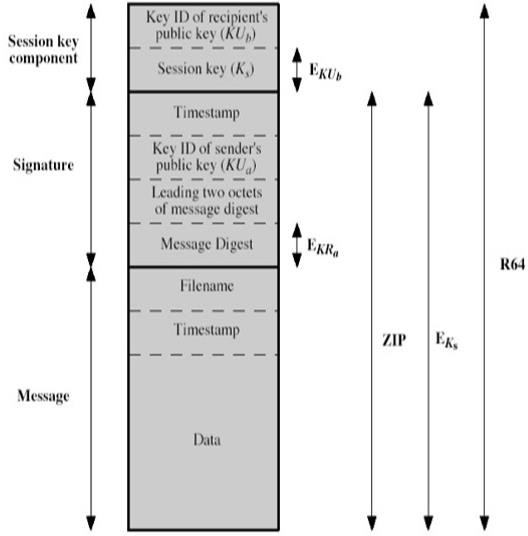
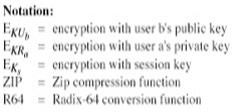
 

Figure: PGP Message Format

## PEM—Privacy Enhanced Mail

**Privacy Enhanced Mail** developed in the late 1980s, is an official Internet standard. PEM covers privacy and authentication for e-mail systems. Messages sent using PEM are first converted to a canonical form so they all have the same conventions about white space. Next a message hash is computed using MD2 or MD5. Then the concatenation of the hash and the message is encrypted using DES. The encrypted message can then be encoded with base64 coding and transmitted to the recipient.

As in PGP, each message is encrypted with a one-time key that is enclosed along with the message. The key can be protected either with RSA or with triple DES using EDE. Key management is more structured than in PGP. Keys are certified by X.509 certificates issued by CAs, which are arranged in a rigid hierarchy starting at a single root. The advantage of this scheme is that certificate revocation is possible by having the root issue CRLs periodically.

## S/MIME

IETF's next venture into e-mail security, called **S/MIME** (**Secure/MIME**), is described in RFCs It provides authentication, data integrity, secrecy, and nonrepudiation. It also is quite flexible, supporting a variety of cryptographic algorithms. Not surprisingly, given the name, S/MIME integrates well with MIME, allowing all kinds of messages to be protected. A variety of new MIME headers are defined, for example, for holding digital signatures.

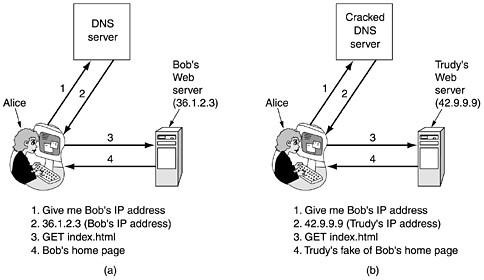
# Web security

## Secure Naming

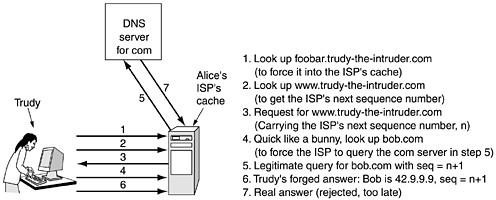
### *DNS Spoofing*

Tricking a DNS server into installing a false IP address is called **DNS spoofing**. A cache that holds an intentionally false IP address like this is called a **poisoned cache**.

## Figure: (a) Normal situation. (b) An attack based on breaking into DNS and modifying Bob's record.



**Figure: How Trudy spoofs Alice's ISP.**



### *Secure DNS*

DNSsec is conceptually extremely simple. It is based on public-key cryptography. Every DNS zone has a public/private key pair. All information sent by a DNS server is signed with the originating zone's private key, so the receiver can verify its authenticity.

DNSsec offers three fundamental services:

* 1. Proof of where the data originated.
  2. Public key distribution.
  3. Transaction and request authentication.

The main service is the first one, which verifies that the data being returned has been approved by the zone's owner. The second one is useful for storing and retrieving public keys securely. The third one is needed to guard against playback and spoofing attacks..

DNS records are grouped into sets called **RRSets** (**Resource Record Sets**), with all the records having the same name, class and type being lumped together in a set. An RRSet may contain multiple *A* records, for example, if a DNS name resolves to a primary IP address and a secondary IP address. Each

RRSet is cryptographically hashed (e.g., using MD5 or SHA-1). The hash is signed by the zone's private key (e.g., using RSA)..

### *Self-Certifying Names*

In this project, the authors designed a secure, scalable, worldwide file system, without modifying (standard) DNS and without using certificates or assuming the existence of a PKI.

## Figure: A self-certifying URL containing a hash of server's name and public key.



The hash is computed by concatenating the DNS name of the server with the server's public key and running the result through the SHA-1 function to get a 160-bit hash. In this scheme, the hash is represented as a sequence of 32 digits and lower-case letters, with the exception of the letters ''l'' and ''o'' and the digits ''1'' and ''0'', to avoid confusion. This leaves 32 digits and letters over. With 32 characters available, each one can encode a 5-bit string. A string of 32 characters can hold the 160-bit SHA-1 hash. Actually, it is not necessary to use a hash; the key itself could be used. The advantage of the hash is to reduce the length of the name.

## SSL—The Secure Sockets Layer

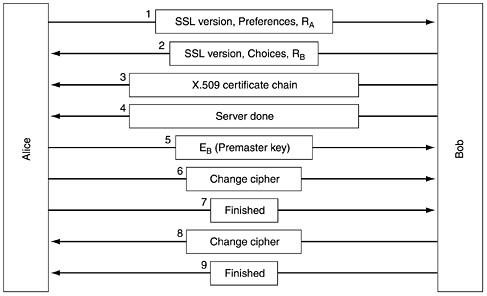
Secure Sockets Layer (SSL), developed by Netscape Communications, provides a secure method of communication for TCP connections, especially for HTTP connections.

SSL builds a secure connection between two sockets, including

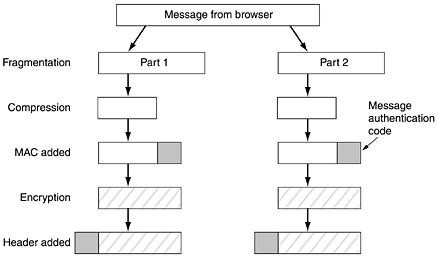
* 1. Parameter negotiation between client and server.
  2. Mutual authentication of client and server.
  3. Secret communication.
  4. Data integrity protection.

SSL consists of two subprotocols, one for establishing a secure connection and one for using it.

## Figure: A simplified version of the SSL connection establishment subprotocol.



**Figure: Data transmission using SSL.**



In 1996, Netscape Communications Corp. turned SSL over to IETF for standardization. The result was **TLS** (**Transport Layer Security**). The TLS version is also known as SSL version 3.1. The first implementations appeared in 1999.

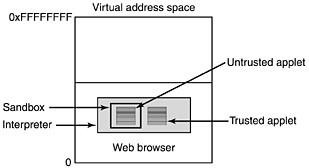
## Mobile Code Security

In the early days, when Web pages were just static HTML files, they did not contain executable code. Now they often contain small programs, including Java applets, ActiveX controls, and JavaScripts. Downloading and executing such **mobile code** is obviously a massive security risk, so various methods have been devised to minimize it.

### *Java Applet Security*

Java applets are small Java programs compiled to a stack-oriented machine language called **JVM** (**Java Virtual Machine**). They can be placed on a Web page for downloading along with the page. After the page is loaded, the applets are inserted into a JVM interpreter inside the browser, as illustrated in Fig. 8-53.

## Figure: Applets can be interpreted by a Web browser.



### *ActiveX*

ActiveX is a set of object-oriented programming technologies and tools that Microsoft developed for Internet Explorer to facilitate rich media playback. ActiveX is a set of object-oriented programming technologies and tools that Microsoft developed for Internet Explorer to facilitate rich media playback.

ActiveX controls are designed to be reusable. As such, a developer building a website did not necessarily have to create his own ActiveX controls. In many cases, it was possible for web developers to enable functionality by using pre-existing ActiveX controls, such as those used for playing multimedia files.

ActiveX controls are designed to be reusable. As such, a developer building a website did not necessarily have to create his own ActiveX controls. In many cases, it was possible for web developers to enable functionality by using pre-existing ActiveX controls, such as those used for playing multimedia files.

### *JavaScript*

JavaScript does not have any formal security model, but it does have a long history of leaky implementations. Each vendor handles security in a different way. For example, Netscape Navigator version 2 used something akin to the Java model, but by version 4 that had been abandoned for a code signing model. The fundamental problem is that letting foreign code run on your machine is asking for trouble.

### *Viruses*

Viruses are another form of mobile code. The difference between a virus and ordinary mobile code is that viruses are written to reproduce themselves. When a virus arrives, either via a Web page, an e- mail attachment, or some other way, it usually starts out by infecting executable programs on the disk. When one of these programs is run, control is transferred to the virus, which usually tries to spread itself to other machines, Some viruses infect the boot sector of the hard disk, so when the machine is booted, the virus gets to run. Viruses have become a huge problem on the Internet and have caused billions of dollars worth of damage.