PANIMALAR INSTITUTE OF TECHNOLOGY

DEPARTMENT OF IT

CCS354 NETWORK SECURITY

UNIT I INTRODUCTION

Basics of cryptography, conventional and public-key cryptography, hash functions, authentication, and digital signatures.

BASICS OF CRYPTOGRAPHY

Cryptography

✓ Cryptography is derived from the Greek words. Crypto means Secret and Graphy means Writing.

✓ Cryptography is the art and science of keeping messages secure.

✓ Cryptography is the science of using mathematics to encrypt and decrypt data.

✓ **Computer Security** - generic name for the collection of tools designed to protect data and to prevent hackers

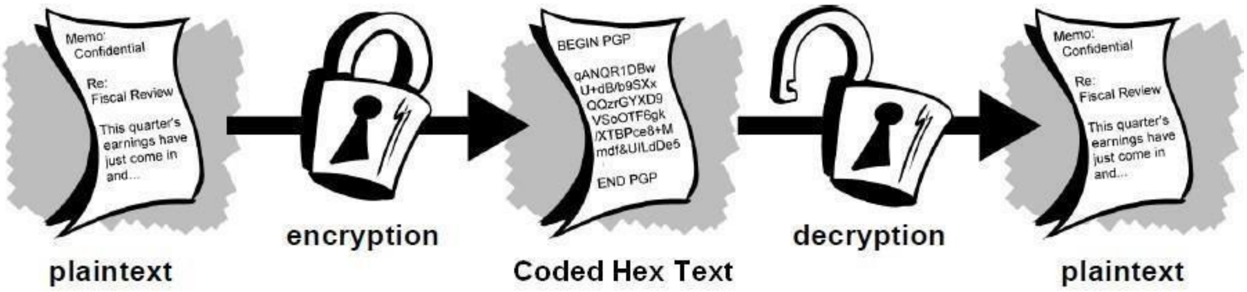
✓ **Network Security** - measures to protect data during their transmission

✓ **Internet Security** - measures to protect data during their transmission over a collection of interconnected networks

Terminologies

- A message is **plaintext** (sometimes called clear text).
- The process of converting plaintext to cipher text using a key is known as **encryption**.
- An encrypted message is **cipher text**.
- The process of turning cipher text back into plaintext is **decryption**.
- **Cryptanalysis** The study of principles and methods of transforming an unintelligible message back into an intelligible message *without* knowledge of the key. Also called **code breaking.**

- **Cryptanalysts** are also called attackers.
- **Cryptology** holds both cryptography and cryptanalysis.



<u>Security Trends</u>

CIA Triad

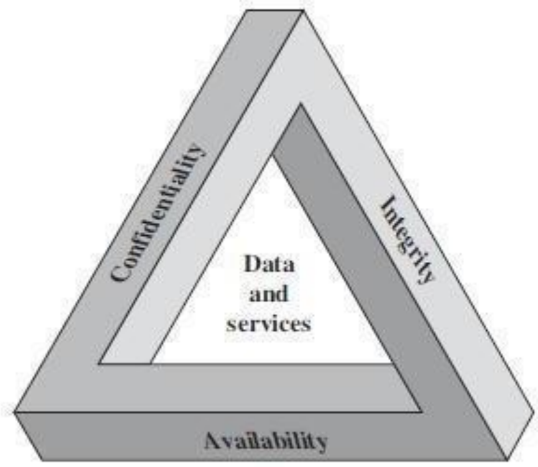


Figure 1.1 The Security Requirements Triad

- **Data confidentiality**

Assures that private or confidential information is not made available or disclosed to unauthorized user.

- **Integrity**

'integrity' ensures that the information is not changed or altered while in transmission.

Availability

- Availability refers to the information resources should be available to authorized users.

Model for Network Security

- A message is to be transferred from one party to another across some sort of Internet service.
- A logical information channel is established by defining a route through the Internet from source to destination.
- **All the techniques for providing security have two components:**
- A security-related transformation on the information to be sent.
- Examples: **encryption of the message**
- Example: **encryption key** used in conjunction with the transformation
- **A trusted third party may be needed to achieve secure transmission.**

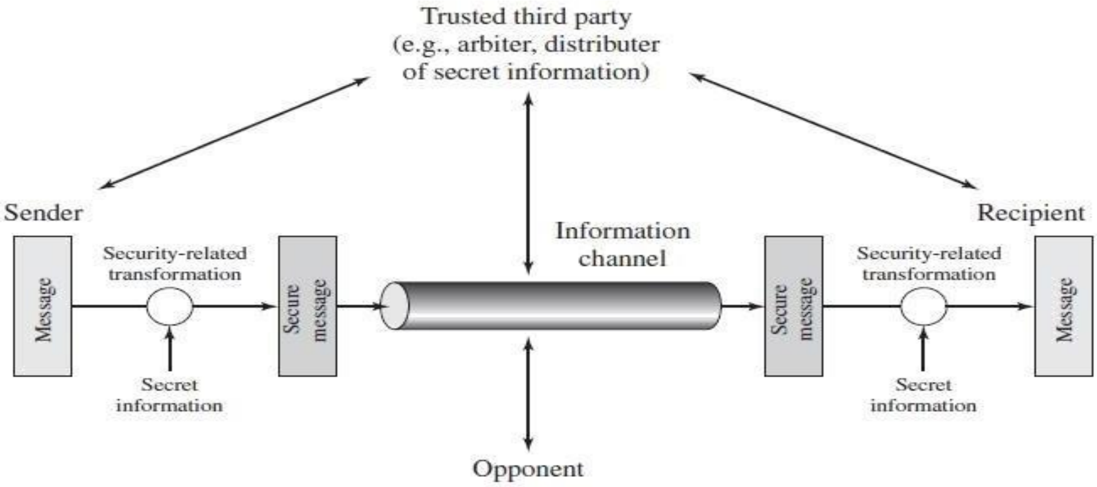


Figure: Network security model

CONVENTIONAL CRYPTOGRAPHY

- **Symmetric encryption is a form of cryptosystem in which encryption and decryption are performed using the same key. It**

is also known as conventional encryption.

- Symmetric encryption transforms plaintext into ciphertext using a secret key and an encryption algorithm. Using the same key and a decryption algorithm, the plaintext is recovered from the ciphertext.

Symmetric Cipher Model

A symmetric encryption scheme has five ingredients

- **Plaintext:** This is the original intelligible message or data that is fed into the algorithm as input.
- **Encryption algorithm:** The encryption algorithm performs various substitutions and transformationson the plaintext.
- **Secret key:** The secret key is also input to the encryption algorithm. The key is a value independent ofthe plaintext and of the algorithm.
- **Ciphertext:** This is the scrambled message produced as output.
- **Decryption algorithm:** This is essentially the encryption algorithm run in reverse. It takes the ciphertext and the secret key and produces the original plaintext

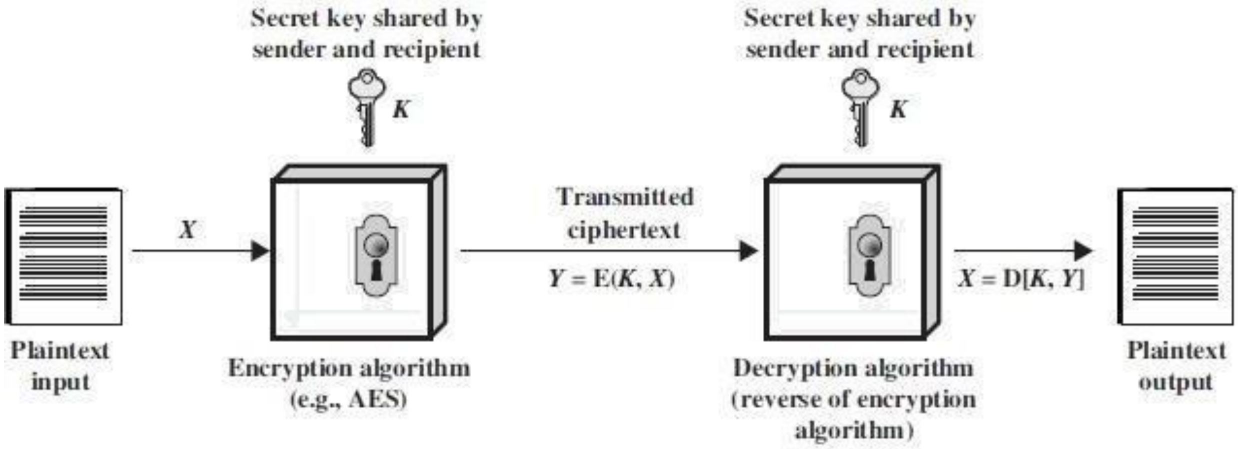

Secret key shared by sender and recipient
K
Secret key shared by sender and recipient
K
X
Transmitted ciphertext
Y = E(K, X)
X = D[K, Y]
Plaintext input
Encryption algorithm (e.g., AES)
Decryption algorithm (reverse of encryption algorithm)
Plaintext output

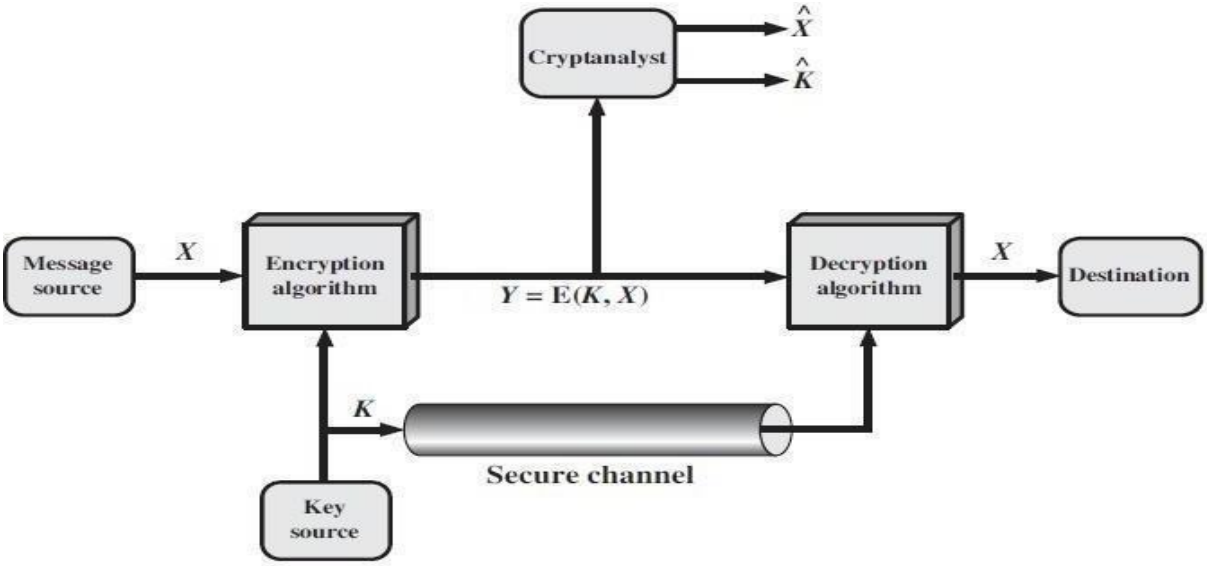
Fig: Simplified Model of Symmetric Encryption

Two requirements for secure use of conventional / symmetric encryption

- We need a strong encryption algorithm
- Sender and receiver must have obtained copies of the secret key in a secure fashion andmust keep the key secure.

Model of Conventional Cryptosystem

- A source produces a message in plaintext, X = [X1, X2, ..., XM]. The M elements of X are letters in some finite alphabet.
- With the message X and the encryption key K as input, the encryption algorithm forms the ciphertext Y = [Y1, Y2, ..., YN].
- We can write this as **Y = E(K, X)**
- The intended receiver, is able to invert the transformation: **X = D(K, Y)**



PUBLIC KEY CRYPTOGRAPHY

Public key encryption algorithm:

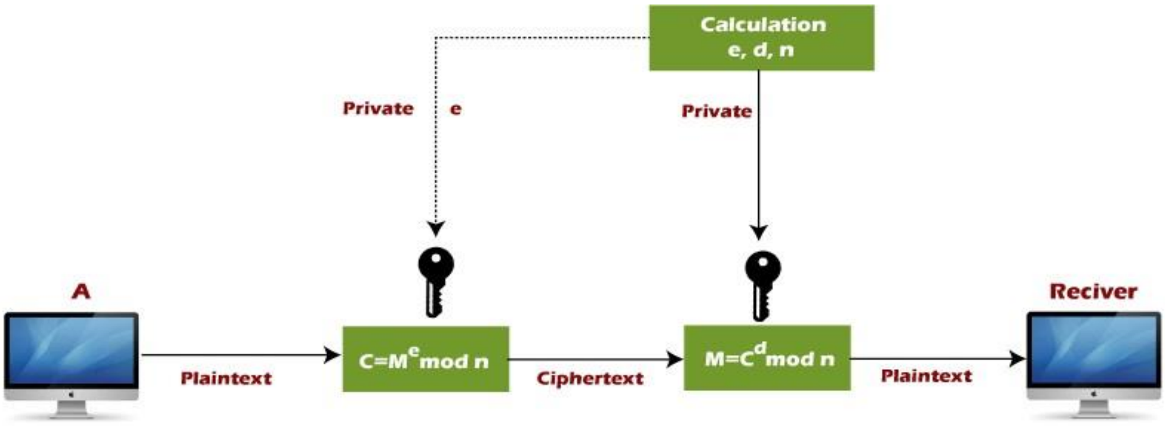
- **Public Key encryption algorithm is also called the Asymmetric algorithm.**

- **Asymmetric algorithms are those algorithms in which sender and receiver use different keys for encryption and decryption.**

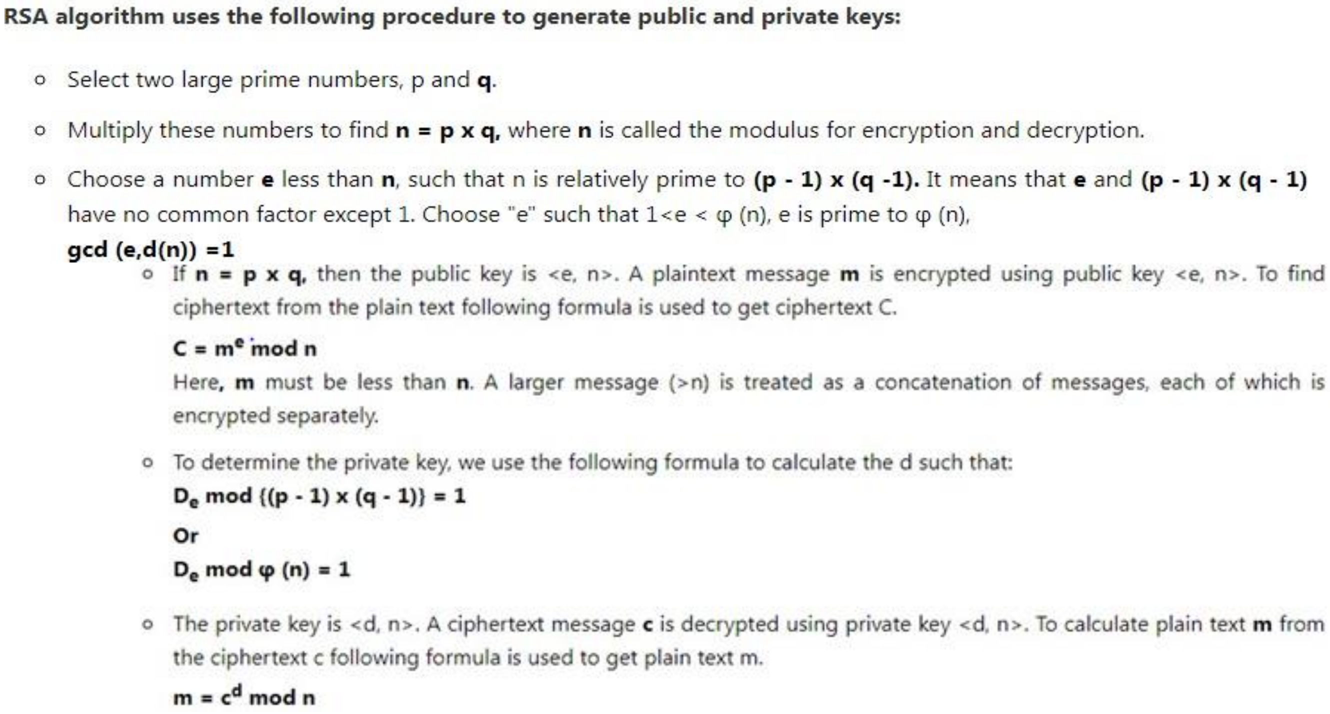
Each sender is assigned a pair of keys:

- **Public key**
- **Private key**

The Public key is used for encryption, and the Private Key is used for decryption.



RSA



RSA algorithm uses the following procedure to generate public and private keys:

- Select two large prime numbers, p and **q**.
- Multiply these numbers to find **n = p x q,** where **n** is called the modulus for encryption and decryption.
- Choose a number **e** less than **n**, such that n is relatively prime to **(p - 1) x (q -1).** It means that **e** and **(p - 1) x (q - 1)** have no common factor except 1. Choose "e" such that 1<e < φ (n), e is prime to φ (n),
  gcd (e,d(n)) =1
  - If **n = p x q,** then the public key is <e, n>. A plaintext message **m** is encrypted using public key <e, n>. To find ciphertext from the plain text following formula is used to get ciphertext C.

    $C = m^e \bmod n$

    Here, **m** must be less than **n**. A larger message (>n) is treated as a concatenation of messages, each of which is encrypted separately.
  - To determine the private key, we use the following formula to calculate the d such that:

    $D_e \bmod \{(p - 1) \times (q - 1)\} = 1$

    Or

    $D_e \bmod \varphi(n) = 1$
  - The private key is <d, n>. A ciphertext message **c** is decrypted using private key <d, n>. To calculate plain text **m** from the ciphertext c following formula is used to get plain text m.

    $m = c^d \bmod n$

The Security of RSA

Brute force attacks: This involves trying all possible private keys.

Mathematical attacks: There are several approaches, all equivalent in effort to factoring the product of two primes.

Timing attacks: These depend on the running time of the decryption algorithm.

SECURITY ATTACK

- **Any action that compromises the security of information owned by an organization.**
- Security attacks are of two types:

1. **Passive attacks**
2. **Active attacks**

Passive Attacks

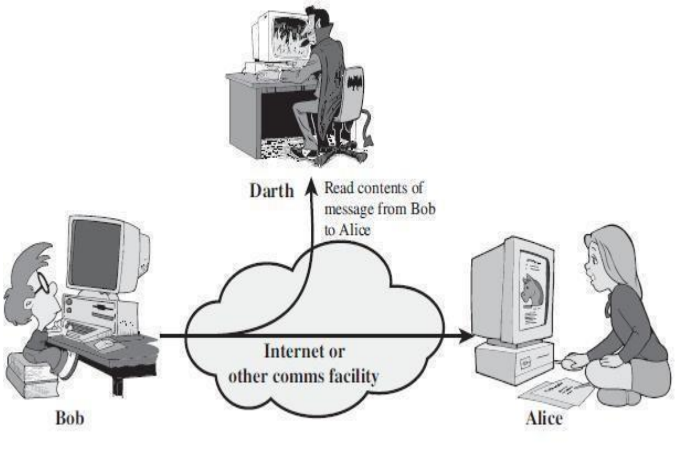
- It is the nature of eavesdropping on, or monitoring of, transmissions.
- The goal is to obtain information that is being transmitted.

Two types of passive attacks

- Release of message contents
- Traffic analysis.

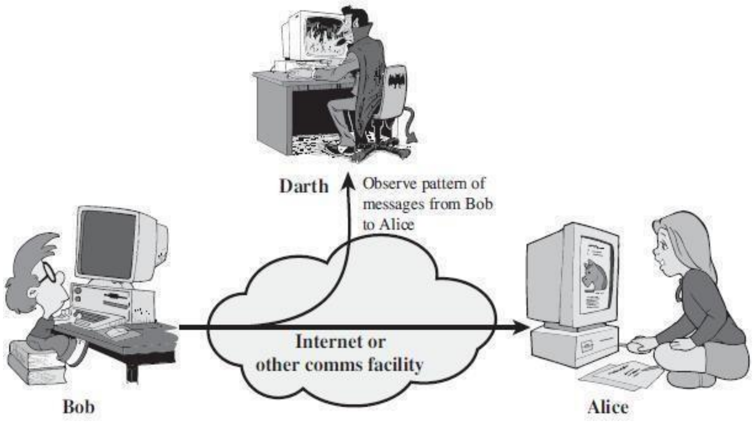
1. Release of Message Contents

A telephone conversation, an electronic mail message, and a transferred file may containsensitive or confidential information. This information is read by the attacker.


Darth
Read contents of message from Bob to Alice
Internet or other comms facility
Bob
Alice

2. Traffic Analysis

- The opponent could determine the location and identity of communicating hosts and couldobserve the frequency and length of messages being exchanged.


Darth
Observe pattern of messages from Bob to Alice
Internet or other comms facility
Bob
Alice

Active Attacks

- Active attacks involve some modification of the data stream or the creation of a falsestream.

- Active attacks can be subdivided into four categories:

➢ masquerade,

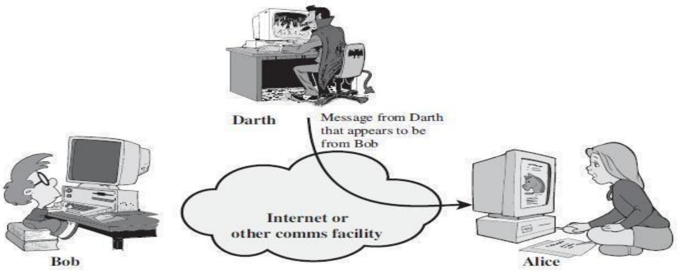
➢ replay,

➢ modification of messages

➢ denial of service

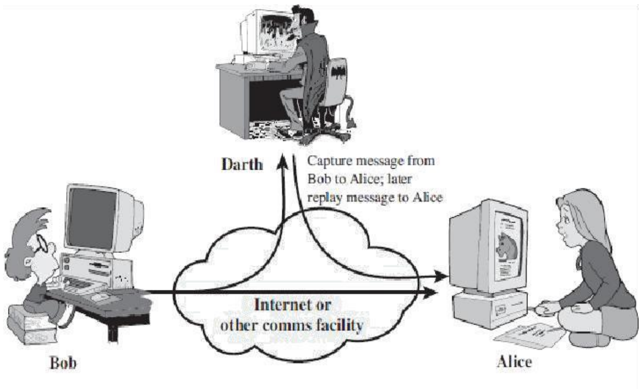
1. Masquerade

 one entity pretends to be a different entity

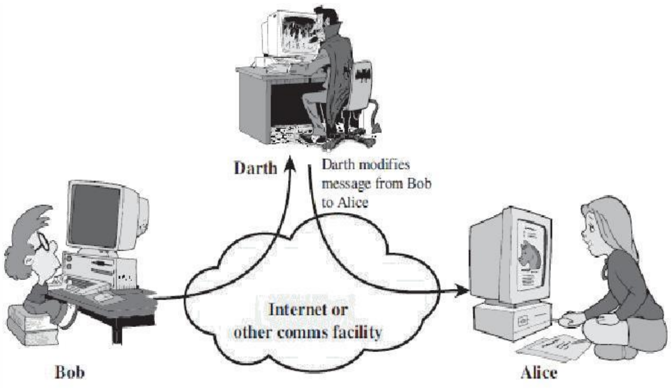

Darth
Message from Darth that appears to be from Bob
Internet or other comms facility
Bob
Alice

2. Replay

 involves passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect.


Darth
Capture message from Bob to Alice; later replay message to Alice
Bob
Internet or other comms facility
Alice

(b) Replay


Darth
Darth modifies message from Bob to Alice
Bob
Internet or other comms facility
Alice

(c) Modification of messages

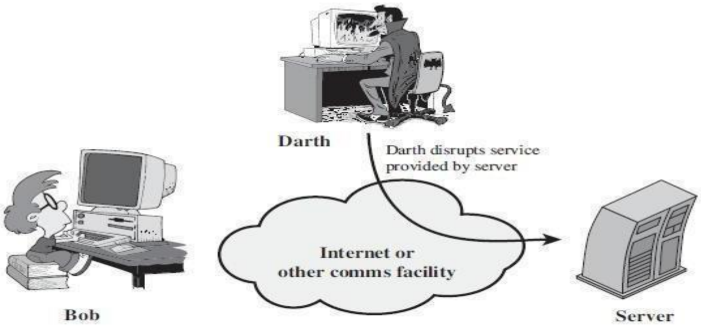
3. Modification of Messages

- Some portion of a legitimate message is altered, or that messages are delayed or reordered,to produce an unauthorized effect

Example

- A message meaning “Allow John Smith to read confidential file accounts” is modified to mean“Allow Fred Brown to read confidential file accounts.”

4. Denial of Service


Darth
Darth disrupts service provided by server
Bob
Internet or other comms facility
Server

(d) Denial of service

- Disruption of an entire network, either by disabling the network or by overloading it withmessages so as to degrade performance

Man In The Middle Attack

- Man In The Middle Attack (MITM) is an attack in which an attacker is able to read, insert and modify messages between two parties without either party knowing that the link between them has been compromised.

HASH FUNCTIONS

- A hash function accepts a variable size message M as input and produces a fixed-size output,referred to as hash code H(M).
- A hash value h is generated by a function H of the form h = H(M)
  where M is a variable-length message and H(M) is the fixed-length hash value.
- The hash value is appended to the message at the source at a time when the message is assumedor known to be correct.
- The receiver authenticates that message by recomputing the hash value.

Requirements for a Hash Function

1. H can be applied to a block of data of any size.
2. H produces a fixed-length output.
3. H(x) is relatively easy to compute for any given x, making both hardware and softwareimplementations practical.
4. For any given value h, it is computationally infeasible to find x such that H(x) = h. This issometimes referred to in the literature as the one-way property.
5. For any given block x, it is computationally infeasible to find y, x such that H(y) = H(x). Thisis sometimes referred to as weak collision resistance. (Given one msg y, it is difficult to find another msg x such that H(y)=H(x),We cannot two different messages with same hash value)
6. It is computationally infeasible to find any pair (x, y) such that H(x) = H(y). This is sometimes referred to as strong collision resistance.

Simple Hash Functions

- All hash functions operate using the following general principles. The input (message, file, etc.) is viewed as a sequence of n-bit blocks.
- The input is processed one block at a time in an iterative fashion to produce an n-bit hash function.
- One of the simplest hash functions is the bit-by-bit exclusive-OR (XOR) of every block.
- This can be expressed as follows: $C_i = b_{i1} + b_{i1} ... + b_{im}$ where
- C_i = i^{th} bit of the hash code, $1 \leq i \leq n$ m = number of n-bit blocks inthe input b_{ij}= ith bit in jth block.
- An improvement over simple hash function is to do bit by bit circular rotation

Procedure:

1. Initially set the n-bit hash value to zero.
2. Process each successive n-bit block of data as follows:

a. Rotate the current hash value to the left by one bit.

b. XOR the block into the hash value.

SECURITY OF HASH FUNCTION AND MAC

- we can group attacks on hash functions and MACs into two categories:

1. **brute-force attacks**
2. **cryptanalysis.**

Brute-Force Attacks

The nature of brute-force attacks differs somewhat for hash functions and MACs.

Hash Functions

The strength of a hash function against brute-force attacks depends solely on the **length of thehash code produced by the algorithm.**

Requirements of Hash Function:

- **One-way**: For any given code h, it is computationally infeasible to find x such thatH(x) = h.
- **Weak collision resistance**: For any given block x, it is computationally infeasible tofind y x with H(y) = H(x).
- **Strong collision resistance**: It is computationally infeasible to find any pair (x, y) suchthat H(x) = H(y).

For a hash code of length n, the level of effort required, as we have seen is proportional to thefollowing:

One way	2^n
Weak collision resistance	2^n
Strong collision resistance	$2^{n/2}$

Message Authentication Codes

- A brute-force attack on a MAC is a more difficult undertaking because it requires knownmessage-MAC pairs.
- To attack a hash code, we can proceed in the following way. Given a fixed message x with n-bit hash code h = H(x), a brute-force method of finding a collision is topick a random bit string y and check if H(y) = H(x).
- The attacker can do this repeatedly off line. To proceed, we need to state the desired security property of a MAC algorithm, which can be expressed as follows:

Cryptanalysis

Hash Functions

- The hash function takes an input message and partitions it into L fixed-sized blocks of b bits each. If necessary, the final block is padded to b bits.
- The final block also includes the value of the total length of the input to the hash function. The inclusion of the length makes thejob of the opponent more difficult.
- Either the opponent must find two messages of equal length that hash to the same value or twomessages of differing lengths that, together with their length values, hash to the same value.

SHA (Secure Hash Algorithm)

The algorithm takes as input a message with a maximum length of less than 2128 bits and producesas output a 512-bit message digest. The input is processed in 1024-bit blocks.

Step 1

Append padding bits.

- The message is padded so that its length is congruent to 896 modulo 1024 [length K 896(mod 1024)].
- Padding is always added, even if the message is already of the desired length. Thus, the number of padding bits is in the range of 1 to 1024.
- The padding consists of a single 1 bit followedby the necessary number of 0 bits.

Step 2

Append length.

- A block of 128 bits is appended to the message.
- This block is treated as an unsigned 128-bit integer (most

significant byte first) and contains the length of the original message (before the padding).

Step 3

Initialize hash buffer.

- A 512-bit buffer is used to hold intermediate and final results of the hash function.
- The buffer canbe represented as eight 64-bit registers (a, b, c, d, e, f, g, h).

Step 4

Process message in 1024-bit (128-word) blocks.

- The heart of the algorithm is a module that consists of 80 rounds; this module is labeled F.
- Each round takes as input the 512-bit buffer value, abcdefgh, and updates the contents of the buffer.
- At input to the first round, the buffer has the value of the intermediate hash value, Hi-1.
- Each round t makes use of a 64-bit value Wt, derived from the current 1024-bit block being processed (Mi). These values are derived using a message schedule described subsequently.

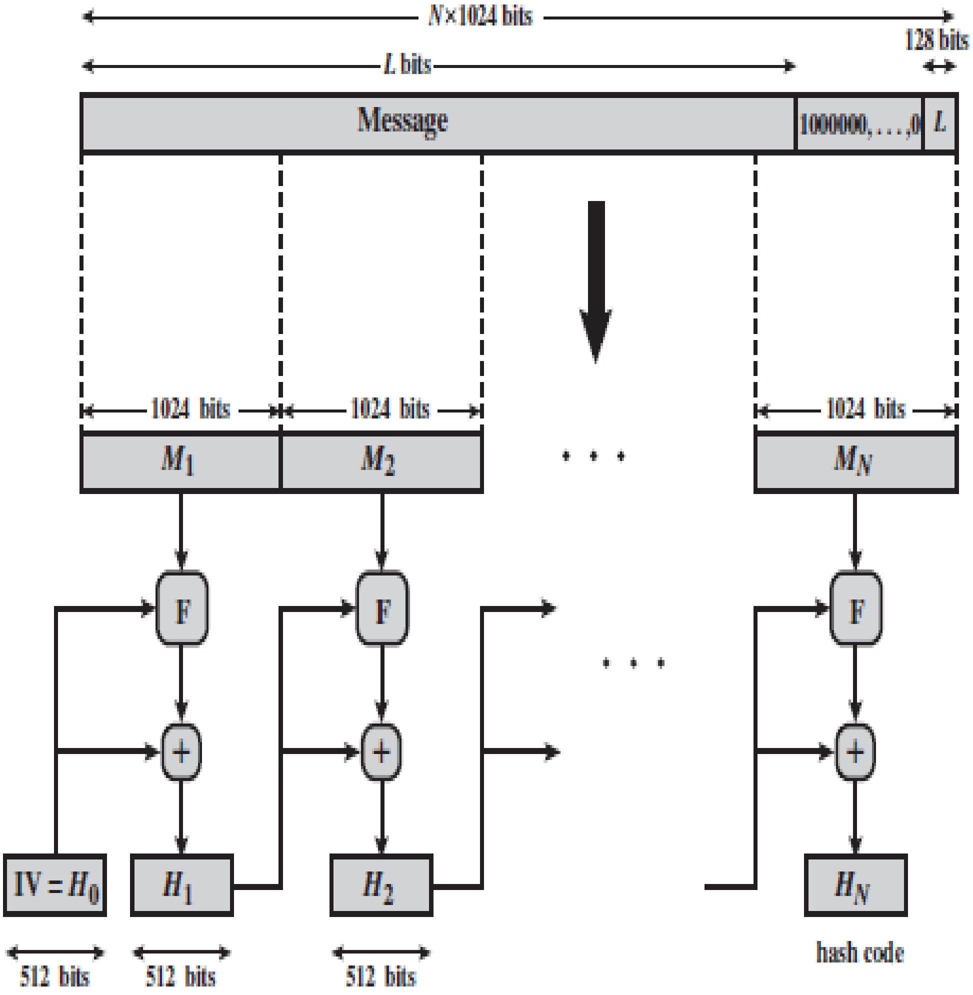


Fig: Message Digest Generation Using SHA-512

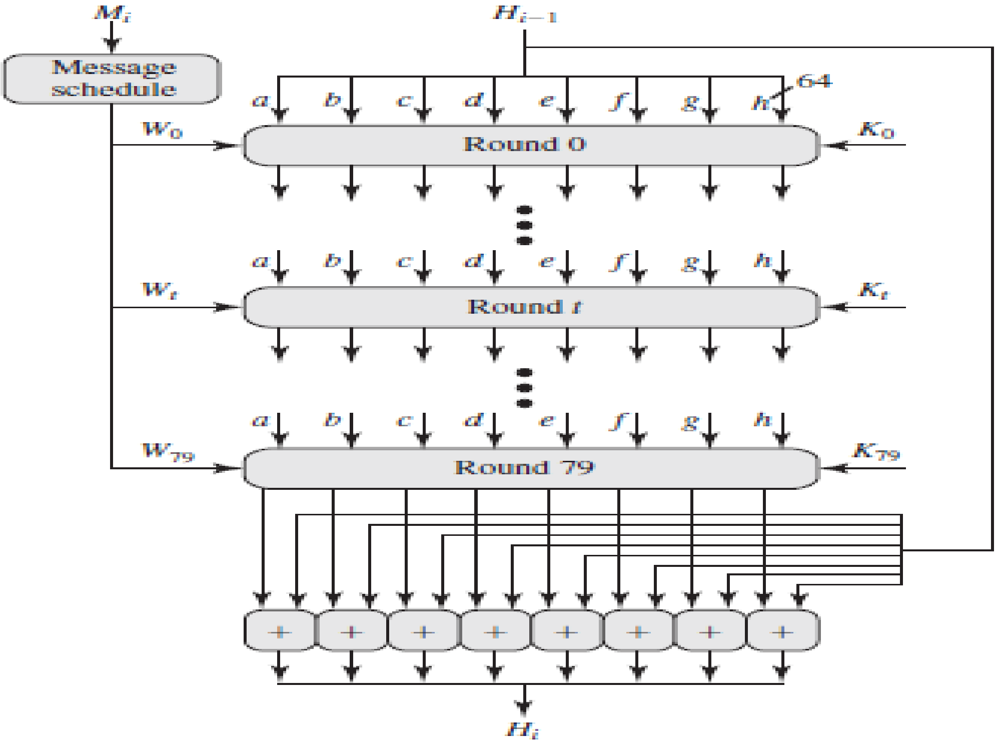


Fig: SHA-512 Processing of a Single 1024-Bit Block

- Each round also makes use of an additive constant Kt, where 0 ≤ t ≤ 79 indicates one of the 80 rounds.
- The output of the eightieth round is added to the input to the first round (Hi-1) to produce Hi. The addition is done independently for each of the eight words in the buffer with each of the corresponding words in Hi-1, using addition modulo 264.

Step 5 Output.

After all N 1024-bit blocks have been processed, the output from the Nth stage is the 512-bit message digest.

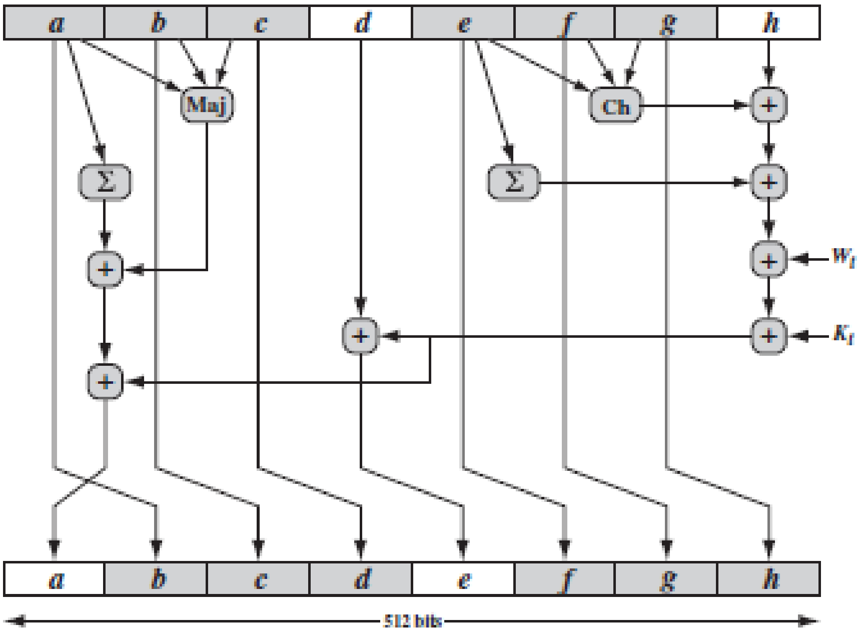


Fig: Elementary SHA Operation (single step)

Summary of SHA-512

H0 = IV

Hi = SUM64(Hi-1, abcdefghi)

MD = HN

where,

IV = initial value of the abcdefgh buffer, defined in step 3

abcdefghi = the output of the last round of processing of the ith messageblock

N = the number of blocks in the message (including padding and lengthfields)

SUM64 = addition modulo 264 performed separately on each word of thepair of inputs

MD = final message digest value

SHA-512 Round Function

$$T_1 = h + \mathrm{Ch}(e, f, g) + \left(\sum\nolimits_1^{512} e \right) + W_t + K_t$$

$$T_2 = \left(\sum\nolimits_0^{512} a \right) + \mathrm{Maj}(a, b, c)$$

h = g,

g = f,

f = e

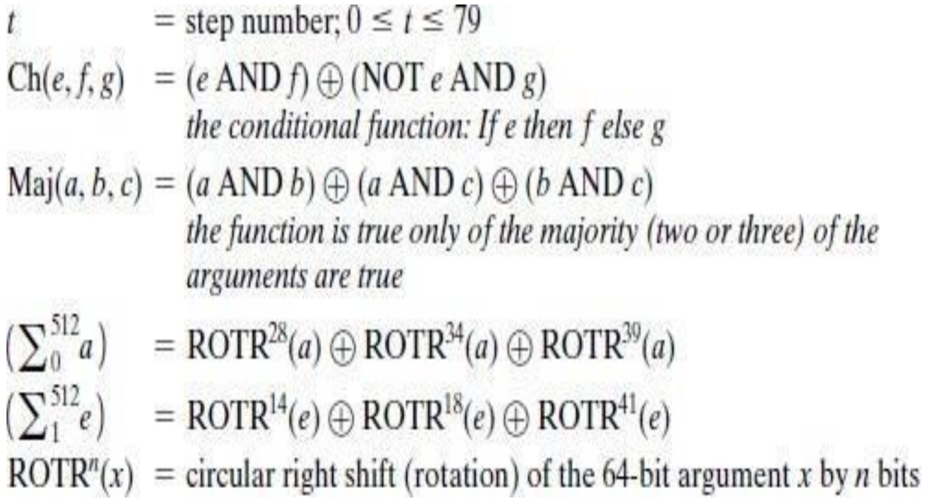
e = d + T1,

d = c

c = b,

b = a

a = T1 + T2



t = step number; $0 \le t \le 79$

$\mathrm{Ch}(e, f, g)$ = $(e \text{ AND } f) \oplus (\text{NOT } e \text{ AND } g)$

the conditional function: If e then f else g

$\mathrm{Maj}(a, b, c)$ = $(a \text{ AND } b) \oplus (a \text{ AND } c) \oplus (b \text{ AND } c)$

the function is true only of the majority (two or three) of the arguments are true

$\left(\sum_0^{512} a\right)$ = $\mathrm{ROTR}^{28}(a) \oplus \mathrm{ROTR}^{34}(a) \oplus \mathrm{ROTR}^{39}(a)$

$\left(\sum_1^{512} e\right)$ = $\mathrm{ROTR}^{14}(e) \oplus \mathrm{ROTR}^{18}(e) \oplus \mathrm{ROTR}^{41}(e)$

$\mathrm{ROTR}^n(x)$ = circular right shift (rotation) of the 64-bit argument x by n bits

Wt = a 64-bit word derived from the current 1024-bit input block

Kt = a 64-bit additive constant

+ = addition modulo 264

AUTHENTICATION

Authentication techniques are used to verify identity.

The authentication of authorized users prevents unauthorized users from gaining access to corporate information systems.

AUTHENTICATION REQUIREMENT

Communication across the network, the following attacks can be identified.

Disclosure – release of message contents to any person or process not possessing the appropriatecryptographic key.

Traffic analysis – discovery of the pattern of traffic between parties.

Masquerade – insertion of messages into the network from fraudulent source.

Content modification – changes to the contents of a message, including insertion, deletion,transposition, and modification.

Sequence modification – any modification to a sequence of messages between parties, includinginsertion, deletion, and reordering.

Timing modification – delay or replay of messages.

Source repudiation – denial of transmission of message by source.

Destination repudiation – denial of receipt of message by destination.

AUTHENTICATION FUNCTION

Any message authentication or digital signature mechanism can be viewed as havingfundamentally two levels.

At the lower level, there must be some sort of function that produces an authenticator, a value tobe used to authenticate a message.

At the higher-level, low-level function is then used as primitive in a higher-level authenticationprotocol that enables a receiver to verify the authenticity of a message.

The types of function that may be used to produce an authenticator are grouped into three classes.

Message Encryption – the ciphertext of the entire message serves as its authenticator.

Message Authentication Code (MAC) – a public function of the message and a secret key that produces a fixed length value that serves as the authenticator.

Hash Function – a public function that maps a message of any length into a fixed-length hashvalue, which serves as the authenticator.

1. Message Encryption:

Message encryption Message encryption by itself can provide a measure of authentication. The analysis differs from symmetric and public key encryption schemes.

(a) If symmetric encryption is used then:

- A message m, transmitted from source A to destination B is encrypted using a secret keyshared by A and B.
- Since only sender and receiver knows key used

- Receiver knows sender must have created it. Hence authentication is provided.
- Know content cannot have been altered. Hence confidentiality is also provided.
- If message has suitable structure, redundancy or a checksum to detect any changes
- Therefore Symmetric Encryption provides authentication and confidentiality.

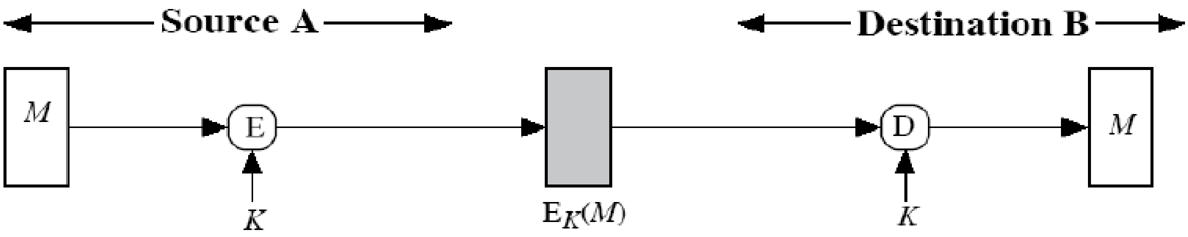
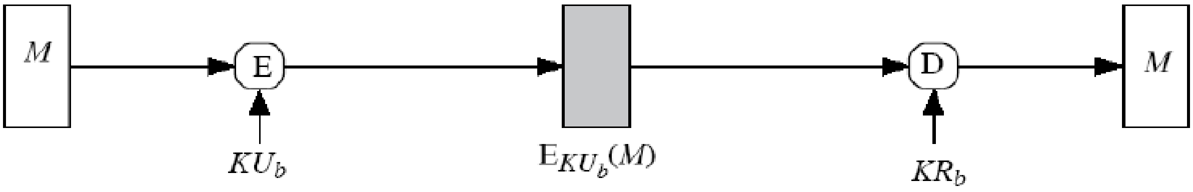


Fig: Symmetric key encryption confidentiality, authentication and signature

(b) If public-key encryption is used:

This method is the use of public key cryptography which provides confidentiality only. The sender A makes use of the public key of the receiver to encrypt the message. Here there is no authentication because any user can use B‟s public key to send a message and claim that only A has sent it.



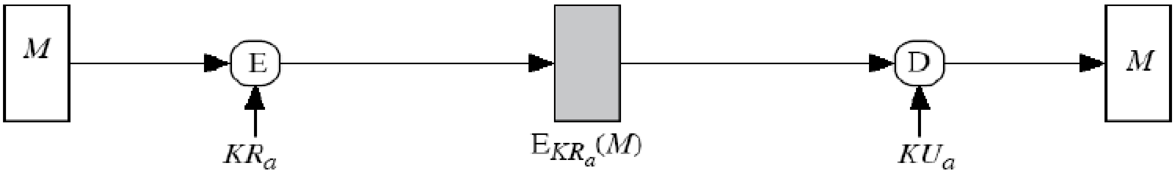
Public key encryption confidentiality

In this method to have only authentication, the message is encrypted with the sender‟s A‟s private key. The receiver B uses the sender‟s A‟s public key to decrypt the message.

Now A cannot deny that it has not transmitted since it only knows its private key. This is called as **authentication or Digital Signature**.

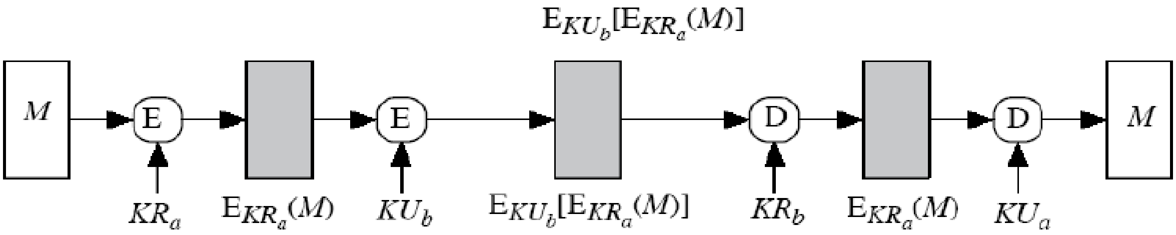
Hence the problem is the,

- ➢ Receiver cannot determine whether the packet decrypted contains some useful messageor random bits.
- ➢ The problem is that anyone can decrypt the message when they know the public key ofsender A.



Public key encryption authentication and signature

This method provides authentication, confidentiality and digital signature. But the problem with this method is the complex public key cryptography algorithm should be applied twice during encryption and twice during decryption.



Public key encryption confidentiality, authentication and signature

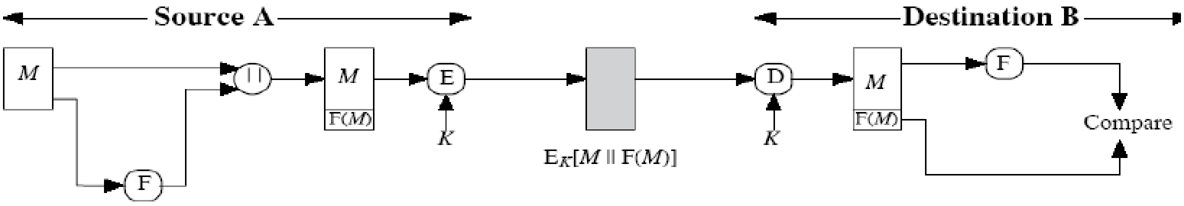
Suppose the message can be any arbitrary bit pattern, in that case, there is no way to determine automatically, at the destination whether an incoming message is the ciphertext of a legitimate message.

One solution to this problem is to force the plaintext to have some structure that is easily recognized but that cannot be replicated without recourse to the encryption function.

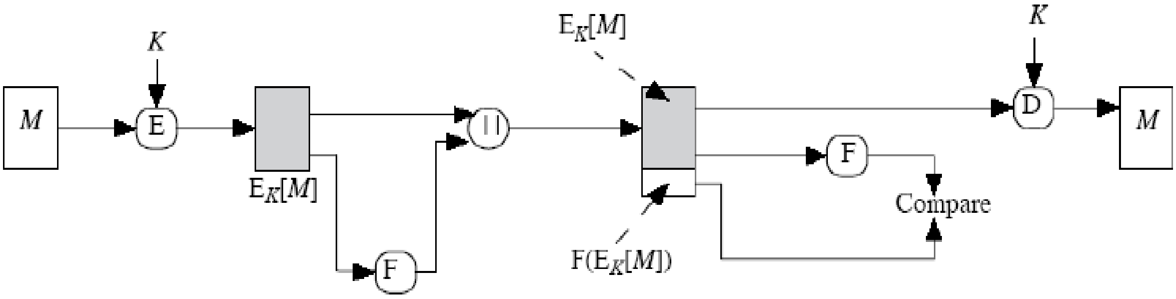
Append an error detecting code, also known as Frame Check Sequence (FCS) or checksumto each message before encryption „A‟ prepares a plaintext message M and then provides this as input to a function F that produces an FCS. The FCS is appended to M and the entire block is thenencrypted.

At the destination, B decrypts the incoming block and treats the result as a message with an appended FCS. B applies the same function F to attempt to reproduce the FCS.

If the calculated FCS is equal to the incoming FCS, then the message is considered authentic. In the internal error control, the function F is applied to the plaintext, whereas in externalerror control, F is applied to the ciphertext.



Internal error control



External error control

DIGITAL SIGNATURES

A digital signature is a cryptographic technique that serves as an electronic counterpart to a handwritten signature or a stamped seal. It provides a way to authenticate the origin, identity, and status of a digital message, document, or transaction. Digital signatures are used to ensure the integrity and authenticity of digital information, and they play a crucial role in securing electronic communication and transactions.

Properties of digital signature:

- It must verify the author and the date and time of the signature.
- It must to authenticate the contents at the time of the signature.
- It must be verifiable by third parties, to resolve disputes.

Requirements for a digital signature:

- The signature must be a bit pattern that depends on the message being signed.
- The signature must use some information unique to the sender, to prevent both forgery and denial.
- It must be relatively easy to produce the digital signature.
- It must be relatively easy to recognize and verify the digital signature.
- It must be computationally infeasible to forge a digital signature, either by constructing a newmessage for an existing digital signature or by constructing a fraudulent digital signature for agiven message.
- It must be practical to retain a copy of the digital signature in storage.

Direct Digital Signature

The term **direct digital signature** refers to a digital signature scheme that **involves only the communicating parties (source, destination).** It is assumed that the destination knows the publickey of the source.

Confidentiality can be provided by encrypting the entire message plus signature with ashared secret key (symmetric encryption).

The validity of the scheme just described depends on the security of the sender"s private key.

Weakness of Direct Digital Signature:

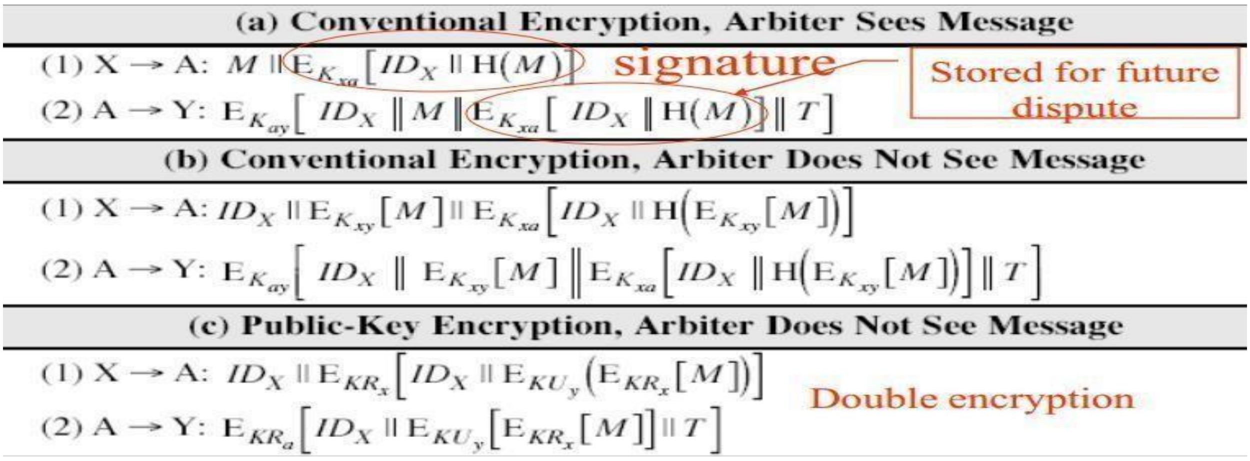
➢ If a sender later wishes to deny sending a particular message, the sender can claim that the private key was lost or stolen and that someone else forged his or her signature.

➢ Another threat is that some private key might actually be stolen from X at time T. The opponent can then send a message signed with X"s signature and stamped with a time before or equal to T.

1. **Arbitrated Digital Signatures**
   - The problem associated with the Direct digital signature can be overcome by using arbitratedschemes.
   - In the arbitrated scheme, the entire signed message from the sender goes to the arbiter A.
   - The arbiter subjects the message and signature to a number of tests to check the origin and control. The date and time is attached to the message.
   - This indicates that the digital signature has been verified and is satisfied. The message is then transmitted to the receiver.

Requirement of the arbiter:

➢ As the arbiter plays a sensitive and crucial role, it should be **a trusted third party.**



(a) Conventional Encryption, Arbiter Sees Message

(1) X → A: $M \| E_{K_{xa}}[ID_X \| H(M)]$

(2) A → Y: $E_{K_{ay}}[\,ID_X \| M \| E_{K_{xa}}[\,ID_X \| H(M)] \| T]$

signature
Stored for future dispute

(b) Conventional Encryption, Arbiter Does Not See Message

(1) X → A: $ID_X \| E_{K_{xy}}[M] \| E_{K_{xa}}\left[ID_X \| H\left(E_{K_{xy}}[M]\right)\right]$

(2) A → Y: $E_{K_{ay}}\left[\,ID_X \| E_{K_{xy}}[M] \| E_{K_{xa}}\left[ID_X \| H\left(E_{K_{xy}}[M]\right)\right] \| T\right]$

(c) Public-Key Encryption, Arbiter Does Not See Message

(1) X → A: $ID_X \| E_{KR_x}\left[ID_X \| E_{KU_y}\left(E_{KR_x}[M]\right)\right]$

(2) A → Y: $E_{KR_a}\left[ID_X \| E_{KU_y}\left[E_{KR_x}[M]\right] \| T\right]$

Notation:X = Sender Y = Recipient A = Arbiter M=Message T=Timestamp

Scheme 1: Conventional encryption, Arbiter sees the message:

The sender X and arbiter A share the master key Kax the receiver y and the arbiterA share the master key Kay

When X wants to send a message M to Y, construct a message computes the hash value H(M). This hash is encrypted using symmetric encryption with the key Kax which acts as signature. The message along with the signature is transmitted to A.

At A, it decrypts the signature and checks the hash value to validate the message. A transmit the message to Y, encrypted with Kay. Y decrypt to extract the message and signature.Disadvantage:

Eaves dropper can read the message as there is no confidentiality.

Scheme 2: Conventional encryption, Arbiter does not see the message:

- Kax and Kay are the master keys.
- Kxy is the key shared between the X and Y
- When x wants to transmit a message to Y, the packet goes to arbiter.
- The same procedure as that of I scheme is used X transmit an identifier, a copy of themessage encrypted with Kxy and a signature

to A.

- The signature is the hash of the message encrypted with K_{xa}
- A decrypt the signature, and checks the hash value to validate the message.
- A cannot read the message, A attaches to it the time stamps, encrypt with K_{xa} andtransmit to Y.

Attack: The arbiter can join with an attacker and deny a message with sender‟s signature.

Scheme 3: Public key encryption, Arbiter does not see the message:

This method uses the public key cryptography which gives authentication and digital signature.The doubly encrypted message is concatenated with ID_X and sent to arbiter.

- A can decrypt the outer encryption to ensure that the message has come from X.
- A then transmit the message with ID_X and time stamp. Advantages:
- No information is shared among parties before communication, hence fraud is avoided.
- No incorrectly dated message can besent. Disadvantages:

The complex public key algorithm is to be twice for encryption and twice for decryption.

DSS

The DSS Approach

The DSS uses an algorithm that is designed to provide only the digital signature function.

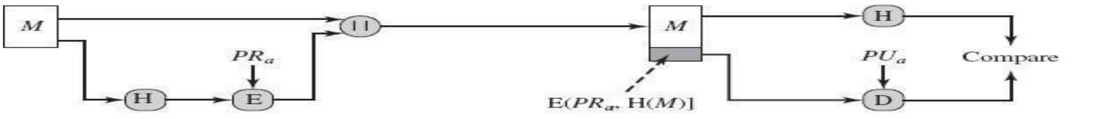
Two Approaches to Digital Signatures

RSA approach

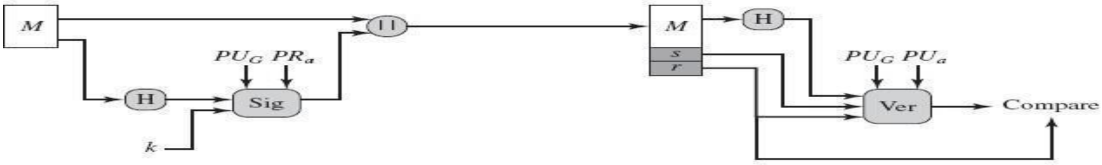
- The message to be signed is input to a hash function that produces a secure hash code of fixed length.
- This hash code is then encrypted using the sender‟s private key to form the signature.
- Both the message and the signature are then transmitted. The recipient takes the message and produces a hash code. The recipient also decrypts the signature using the sender‟s public key.
- If the calculated hash code matches the decrypted signature, the signature is accepted as valid. Because only the sender knows the private key, only the sender could have produced a valid signature.

DSS approach

- The DSS approach also makes use of a hash function. The hash code is provided as input to a signature function along with a random number generated for this particular signature.
- The signature function also depends on the sender‟s private key (*PRa*) and a global public key (*PUG*). The result is a signature consisting of two components, labeled *s* and *r*. At the receiving end, the hash code of the incoming message is generated. This plus the signature is input to a verification function. The verification function also depends on the global public key as well as the sender‟s public key, which is paired with the sender‟s private key.
- The output of the verification function is a value that is equal to the signature component if the signature is valid. The signature function is such that only the sender, with knowledge of the private key, could have produced the valid signature.



(a) RSA approach



(b) DSS approach

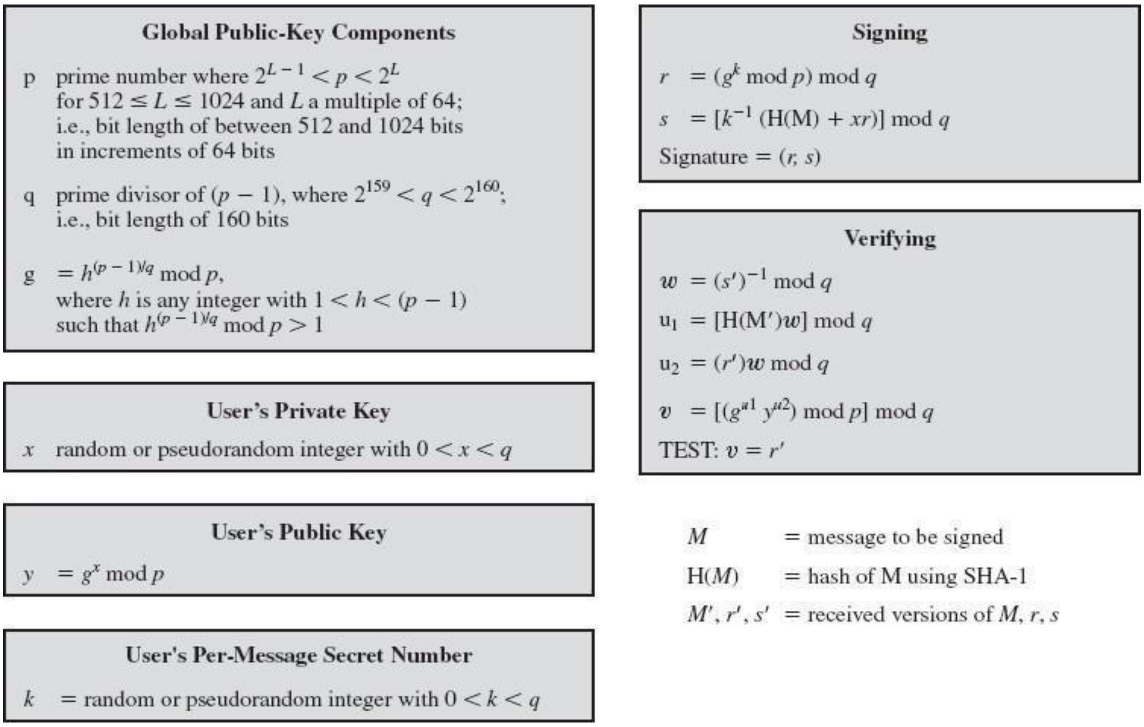
The Digital Signature Algorithm:

There are **three parameters** that are public and can be common to a group of users.

- A 160-bit **prime number q** is chosen.
- Next, a **prime number p** is selected with a length between 512 and 1024 bits such that q divides (*p* - 1).
- Finally, ***g* is chosen** to be of the form $h^{(p-1)/q}$mod *p*, where *h* is an integer between 1 and (p-1).

With these numbers in hand, each user selects a private key and generates a public key. The private key x must be a number from 1 to (q-1) and should be chosen randomly. T

- The public key is calculated from the private key as y = g^x mod p .
- To create a signature, a user calculates two quantities, *r* and *s*, that are functions of the public key components (*p, q, g*), the user's private key (*x*), the hash code of the message, H(*M*), and an additional integer *k* that should be generated randomly and be unique for each signing.
- At the receiving end, verification is performed using the formulas.
- The receiver generates a quantity *v* that is a function of the public key components, the sender's public key, and the hash code of the incoming message.
- If this quantity matches the *r* component of the signature, then the signature is validated.



Global Public-Key Components

p prime number where $2^{L-1} < p < 2^{L}$
for $512 \leq L \leq 1024$ and L a multiple of 64;
i.e., bit length of between 512 and 1024 bits
in increments of 64 bits

q prime divisor of $(p - 1)$, where $2^{159} < q < 2^{160}$;
i.e., bit length of 160 bits

g $= h^{(p-1)/q} \bmod p$,
where h is any integer with $1 < h < (p - 1)$
such that $h^{(p-1)/q} \bmod p > 1$

User's Private Key

x random or pseudorandom integer with $0 < x < q$

User's Public Key

$y \; = g^{x} \bmod p$

User's Per-Message Secret Number

k = random or pseudorandom integer with $0 < k < q$

Signing

$r \; = (g^{k} \bmod p) \bmod q$

$s \; = [k^{-1} (\mathrm{H}(\mathrm{M}) + xr)] \bmod q$

Signature = (r, s)

Verifying

$w \; = (s')^{-1} \bmod q$

$\mathrm{u}_1 \; = [\mathrm{H}(\mathrm{M}')w] \bmod q$

$\mathrm{u}_2 \; = (r')w \bmod q$

$v \; = [(g^{u1} y^{u2}) \bmod p] \bmod q$

TEST: $v = r'$

M	= message to be signed
$\mathrm{H}(M)$	= hash of M using SHA-1
M', r', s'	= received versions of M, r, s

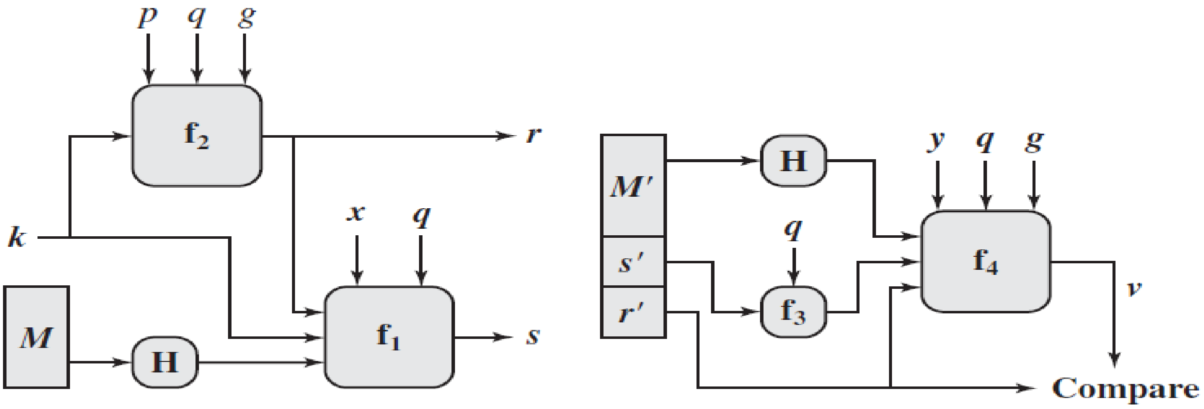


Fig: DSS Signing and Verifying