Information Security

CHP 1:

Computer Security

↳ measures and controls

↳ ensuring confidentiality, integrity, availability

of systems, resource or asset that

include hardware, software, data, communication line and networks

→ ak CIA Triad → defines security objectives

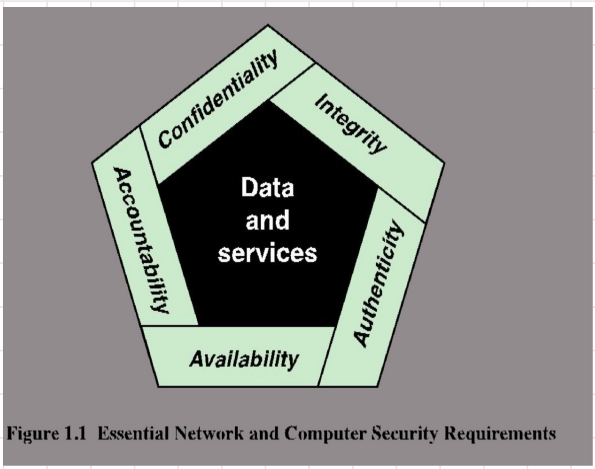


Figure 1.1 Essential Network and Computer Security Requirements

There are 2 more additional concepts

Authencity

↳ ensuring something is genuine, verifiable, trustworthy

↳ confidence in the validity of a transmission/msg

↳ So basically

↳ verifying users are who they claim to be

↳ verify all inputs come from trusted resources

Accountability

↳ ensures actions taken by an entity can be uniquely traced back to them

↳ This helps with

↳ non repudiation

↳ fault detection

↳ intrusion prevension

↳ legal actions

↳ Systems must keep record of their activities

to permit later for forensic analysis

to trace security breaches.

key security concepts

1. Confidentiality

↳ ensure confidential info isn't disclosed

↳ disclosure follow authorised restrictions

protecting privacy and proprietary data

↳ relating to ownership

Privacy

↳ ensures individuals have control over what personal info is collected

2. Integrity

↳ Guards info from unauthorised changes ensuring authenticity and non-repudiation

↳ Data integrity

↳ assures info and programs are modified only in authorised and approved ways

↳ System integrity

↳ ensures that a system functions as intended w/o being compromised by unauthorised changes

3. Availability

↳ assures system works promptly

↳ assures service isn't denied to authorised users

Levels of Impact

↳ These are the levels of impact used when there is a security breach → loss of confidentiality, integrity, availability

1. Low

↳ the loss is expected to have a limited negative impact on operations, assets and individuals

e.g.
(i) mission capability is reduced but the organization can still perform its primary functions, though less effectively
ii) minor damage occurs to organizational assets
(iii) minor financial loss happens
(iv) minor harm is caused to individuals.

2. Moderate

↳ the loss is expected to have a significant negative impact on operations, assets and individuals

e.g.
(i) mission capability is significantly reduced, but the organization can still perform its primary functions, though much less effectively;
(ii) significant damage is done to organizational assets;
(iii) significant financial loss occurs;
(iv) significant harm is caused to individuals, but without causing death or life-threatening injuries.

3. High

↳ the loss is expected to have a severe/catastrophic negative impact on operations, assets and individuals

e.g.
(i) mission capability is severely degraded or lost, preventing the organization from performing one or more of its primary functions;
(ii) major damage is done to organizational assets;
(iii) major financial loss occurs;
(iv) severe or catastrophic harm is caused to individuals, including loss of life or serious, life-threatening injuries.

Computer security challanges

1.Not as simple as it seems:
While the key security requirements like confidentiality, authentication, nonrepudiation, and integrity
may appear straightforward, the mechanisms to achieve them
are often intricate and require deep understanding.

2.Constant threat of attack:
Security designs must always account for potential attacks,
which often exploit unexpected weaknesses
by approaching problems from unconventional angles.

3.Counterintuitive solutions:
Due to the complexity of threats, security mechanisms are often elaborate
and not immediately obvious from the requirements alone.
Their necessity becomes clear only when threats are fully understood.

4.Placement of security mechanisms:
Security must be implemented at the right points, both physically (e.g., within a network)
and logically (e.g., at specific layers of the system architecture, such as TCP/IP).

5.Involves more than algorithms:
Security relies on secret information (e.g., encryption keys),
which brings challenges related to the creation, distribution, and protection of these secrets,
alongside the complications of protocols and networks.

6.Battle of wits:
Security is a continuous struggle between
attackers, who only need to find a single vulnerability, and
defenders, who must secure all potential weaknesses to maintain perfect security.

7.Perception of security investment:
Often, the value of investing in security measures is not appreciated
until after a security failure has occurred.

8.Ongoing monitoring:
Security requires constant vigilance,
which is difficult to maintain in today's fast-paced and overloaded work environments.

9.An afterthought:
Security is too often considered late in the design process,
rather than being integrated from the start.

10.Seen as a barrier:
Many users and administrators perceive strong security
as an obstacle to efficient system operation and ease of use.

Adversary → Threat Agent

↳ an organization, group, individual intending to cause harm

Attack

↳ Malicious activity aiming to collect, disrupt, damage, destroy information or systems

Countermeasure

↳ a tool | technique to block harmful activities and prevent unauthorised access

Risk

↳ The chance of harm occuring based on potential impact and likelihood

Security Policy

↳ rules to maintain the security of systems and data

System Resource → Asset

↳ Important programs, system, equipment or people critical to operations

Threat

↳ Anything that could harm operations, assests or info systems

Vulnerability

↳ a weakness in a system that could be exploited by a threat

↓

The relationship among these terms

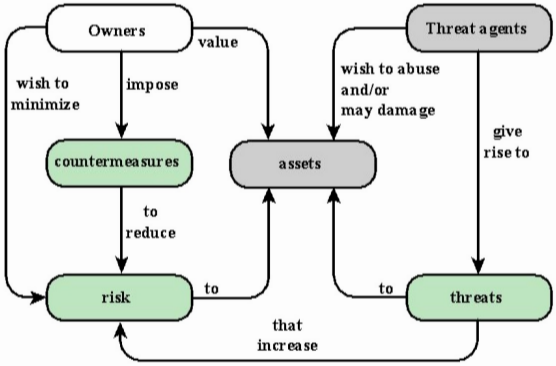


Figure 1.2 Security Concepts and Relationships

Q1.

a) Give one sentence answers (attempt all parts on one side of a page) **[0.5 x 4 = 2]**

i. **Define the terms:** i) Information Security, ii) Cybersecurity and iii) Network security.
Information security refers to the practice of protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction. Cybersecurity is a subset of information security that specifically focuses on protecting computer systems, networks, and digital devices from cyber threats. Network security is a component of cybersecurity that is concerned with securing computer networks against unauthorized access, attacks, and other threat

ii. **Explain** the linkage between asset, vulnerability, threat and attack.
An asset refers to any valuable resource. A vulnerability is a weakness or flaw in the security of an asset. A threat is any potential danger or harm to an asset, either intentional or accidental. An attack is a deliberate action taken by an attacker to exploit a vulnerability and compromise an asset. Attacks can take many forms, including malware infections, phishing scams, denial-of-service attacks, or physical theft. The goal of information security is to identify and mitigate vulnerabilities and threats to protect assets from attacks.

iii. **Why** an organization needs InfoSec team in addition to Network Security staff in the IT department?
Network Security staff has limited knowledge about i) organization wide risk assessment and security policy create and enforcement and ii) cybersecurity issues like web security, software security and operating system security.

iv. **How** a vulnerability creates an opportunity for an attack?
Vulnerability in application and/or system software, hardware or firmware allows undetected access to system resources that otherwise are not accessible to unauthorized users. For example, the access could be a privileged command prompt on a target node, creation a process undetected, or a favorable change in configuration for future reentry, etc.

Assets of a Computer System

1. Hardware
- computer systems
- data storage devices
- data processing devices
- data communication devices

2. Software
- Operating System
- System utilities
- Applications

3. Data
- files
- database
- password files

4. Communication Facilities and networks
- LAN, WAN communication
  - links
  - bridges
  - routers ...

VULNERABILITIES, THREATS, ATTACKS

Categories of vulnerabilities

1. Corrupted → loss of integrity → gives wrong ans
   - stored data values may differ from what they should be because they have been improperly modified.
2. Leaky → loss of confidentiality
   - someone who should not have access to some or all of the information available through the network obtains such access.
3. Unavailable or very slow → loss of availability
   - using the system or network becomes impossible or impractical.

Threats
- capable of exploiting vulnerabilities
- represent potential security harm to an asset

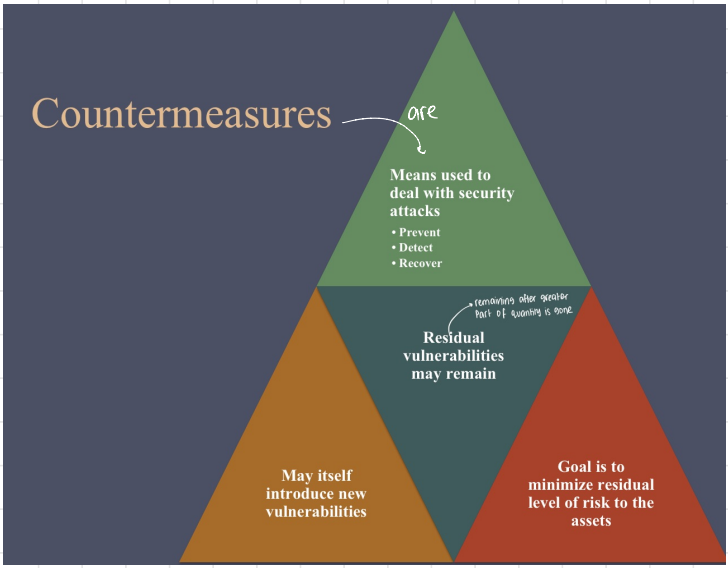
Attacks → it is a threat made carried out

1. Passive
- attempts to learn or use info from system
- it does not affect system resources

2. Active
- attempt to alter system resources or affect their operations

3. Insider
- initiated by an entity inside the security parameter

4. Outsider
- initiated from outside the parameter



Threat Consequence	Threat Action (Attack)
Unauthorized Disclosure — threat to confidentiality A circumstance or event whereby an entity gains access to data for which the entity is not authorized.	**Exposure:** Sensitive data are directly released to an unauthorized entity. **Interception:** An unauthorized entity directly accesses sensitive data traveling between authorized sources and destinations. **Inference:** A threat action whereby an unauthorized entity indirectly accesses sensitive data (but not necessarily the data contained in the communication) by reasoning from characteristics or byproducts of communications. **Intrusion:** An unauthorized entity gains access to sensitive data by circumventing a system's security protections.
Deception — threat to integrity A circumstance or event that may result in an authorized entity receiving false data and believing it to be true.	**Masquerade:** An unauthorized entity gains access to a system or performs a malicious act by posing as an authorized entity. **Falsification:** False data deceive an authorized entity. **Repudiation:** An entity deceives another by falsely denying responsibility for an act.
Disruption — threat to availability/system integrity A circumstance or event that interrupts or prevents the correct operation of system services and functions.	**Incapacitation:** Prevents or interrupts system operation by disabling a system component. **Corruption:** Undesirably alters system operation by adversely modifying system functions or data. **Obstruction:** A threat action that interrupts delivery of system services by hindering system operation.
Usurpation — threat to system integrity A circumstance or event that results in control of system services or functions by an unauthorized entity.	**Misappropriation:** An entity assumes unauthorized logical or physical control of a system resource. **Misuse:** Causes a system component to perform a function or service that is detrimental to system security.

Table 1.2

Threat Consequences, and the Types of Threat Actions That Cause Each Consequence

Based on RFC 4949

**Table is on page 10 in the textbook.

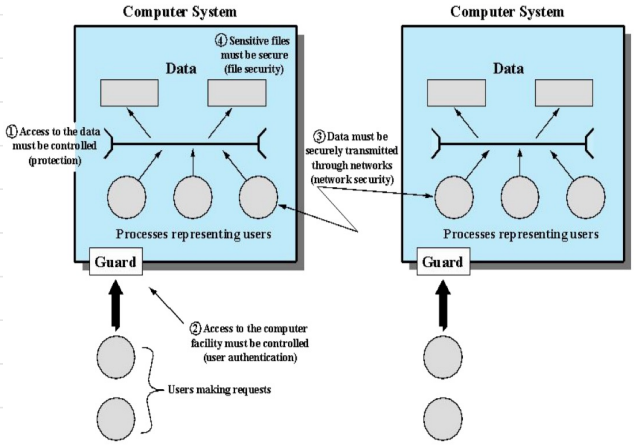
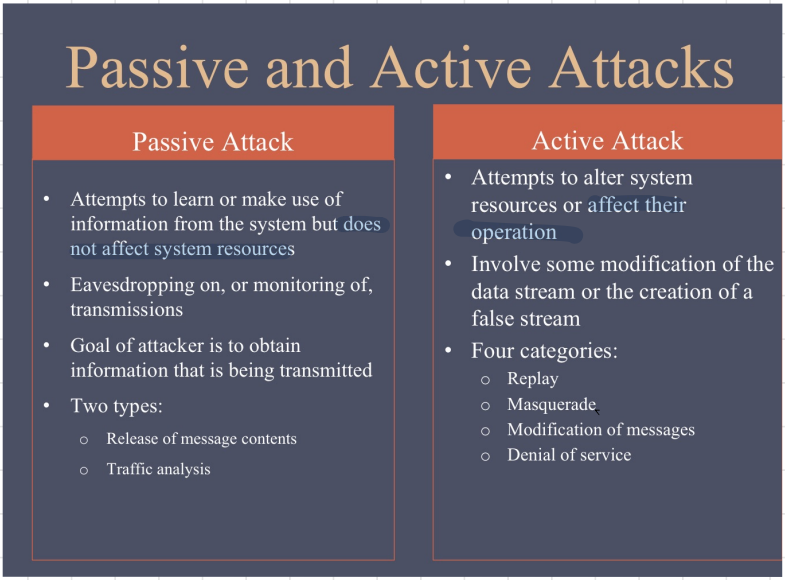


Figure 1.3 Scope of Computer Security. This figure depicts security concerns other than physical security, including control of access to computers systems, safeguarding of data transmitted over communications systems, and safeguarding of stored data.

Table 1.3
Computer and Network Assets, with Examples of Threats

	Availability	Confidentiality	Integrity
Hardware	Equipment is stolen or disabled, thus denying service.	An unencrypted CD-ROM or DVD is stolen.	
Software	Programs are deleted, denying access to users.	An unauthorized copy of software is made.	A working program is modified, either to cause it to fail during execution or to cause it to do some unintended task.
Data	Files are deleted, denying access to users.	An unauthorized read of data is performed. An analysis of statistical data reveals underlying data.	Existing files are modified or new files are fabricated.
Communication Lines and Networks	Messages are destroyed or deleted. Communication lines or networks are rendered unavailable.	Messages are read. The traffic pattern of messages is observed.	Messages are modified, delayed, reordered, or duplicated. False messages are fabricated.



There are two types of passive attacks:

1.Release of message contents:
This is straightforward, where sensitive information,
like a phone conversation, email, or file transfer
,is intercepted by an opponent.
The goal is to prevent unauthorized access to
the contents of these communications.

2.Traffic analysis:
- This is more subtle.
- Even if messages are encrypted to hide their contents,
an attacker could still observe the patterns of communication,
such as
  - ↳ the sender and receiver's identities,
  - ↳ frequency,
  - ↳ length of messages.

This information could help an attacker make educated guesses
about the nature of the communication.

Passive attacks are hard to detect
because they don't alter the data.
Messages are sent and received normally,
with neither party aware
that someone might be monitoring them.
Since detection is difficult, prevention, usually through encryption,
is the primary defense against these attacks.

1.Replay attack:
This occurs when an attacker captures a data unit
and retransmits it to create unauthorized effects.

2.Masquerade attack:
In this case, one entity pretends to be another.
This often involves
  - ↳ capturing and
  - ↳ replaying authentication sequences,

allowing a user with limited privileges to
impersonate a more privileged user.

3.Message modification:
This involves altering a legitimate message,
  - ↳ delaying it, or
  - ↳ changing the order of messages

to create unauthorized effects.
For example, changing a message from
“Allow John Smith to read the confidential file”
to “Allow Fred Brown to read the confidential file.”

4.Denial of service (DoS):
This attack disrupts normal communication, either by
  - ↳ blocking messages to a specific target or
  - ↳ by overwhelming a network,

causing degraded performance.

Active attacks are the opposite of passive attacks.
While passive attacks are
  - ↳ hard to detect but can be prevented,

active attacks are
  - ↳ difficult to completely prevent

since they require constant physical protection of communication facilities.
Instead, the focus is on detecting
  - ↳ active attacks and
  - ↳ recovering from their effects,

as detection can also help deter future attacks.

Media protection: (i) Protect information system media, both paper and digital; (ii) limit access to information on information system media to authorized users; and (iii) sanitize or destroy information system media before disposal or release for reuse.

Physical and environmental protection: (i) Limit physical access to information systems, equipment, and the respective operating environments to authorized individuals; (ii) protect the physical plant and support infrastructure for information systems; (iii) provide supporting utilities for information systems; (iv) protect information systems against environmental hazards; and (v) provide appropriate environmental controls in facilities containing information systems.

Planning: Develop, document, periodically update, and implement security plans for organizational information systems that describe the security controls in place or planned for the information systems and the rules of behavior for individuals accessing the information systems.

Personnel security: (i) Ensure that individuals occupying positions of responsibility within organizations (including third-party service providers) are trustworthy and meet established security criteria for those positions; (ii) ensure that organizational information and information systems are protected during and after personnel actions such as terminations and transfers; and (iii) employ formal sanctions for personnel failing to comply with organizational security policies and procedures.

Risk assessment: Periodically assess the risk to organizational operations (including mission, functions, image, or reputation), organizational assets, and individuals, resulting from the operation of organizational information systems and the associated processing, storage, or transmission of organizational information.

Systems and services acquisition: (i) Allocate sufficient resources to adequately protect organizational information systems; (ii) employ system development life cycle processes that incorporate information security considerations; (iii) employ software usage and installation restrictions; and (iv) ensure that third-party providers employ adequate security measures to protect information, applications, and/or services outsourced from the organization.

System and communications protection: (i) Monitor, control, and protect organizational communications (i.e., information transmitted or received by organizational information systems) at the external boundaries and key internal boundaries of the information systems; and (ii) employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational information systems.

System and information integrity: (i) Identify, report, and correct information and information system flaws in a timely manner; (ii) provide protection from malicious code at appropriate locations within organizational information systems; and (iii) monitor information system security alerts and advisories and take appropriate actions in response.

Table 1.4

Security Requirements

(FIPS 200)

(page 2 of 2)

(Table can be found on pages 16-17 in the textbook.)

Fundamental Security Design Principles

Economy of mechanism	Fail-safe defaults	Complete mediation	Open design
Separation of privilege	Least privilege	Least common mechanism	Psychological acceptability
Isolation	Encapsulation	Modularity	Layering
Least astonishment			

→ that can guide the development of protection mechanisms

∫as

It has not been possible to create security design and implementation methods that completely eliminate security flaws and prevent all unauthorized actions.

Attack Surfaces

Consist of the reachable and exploitable vulnerabilities in a system

Examples:

- Open ports on outward facing Web and other servers, and code listening on those ports
- Services available on the inside of a firewall
- Code that processes incoming data, email, XML, office documents, and industry-specific custom data exchange formats
- Interfaces, SQL, and Web forms
- An employee with access to sensitive information vulnerable to a social engineering attack

Attack Surface Categories

Network Attack Surface

Vulnerabilities over an enterprise network, wide-area network, or the Internet

Included in this category are network protocol vulnerabilities, such as those used for a denial-of-service attack, disruption of communications links, and various forms of intruder attacks

Software Attack Surface

Vulnerabilities in application, utility, or operating system code

Particular focus is Web server software

Human Attack Surface

Vulnerabilities created by personnel or outsiders, such as social engineering, human error, and trusted insiders

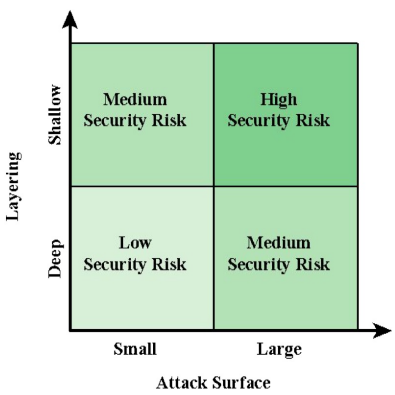


Figure 1.4 Defense in Depth and Attack Surface

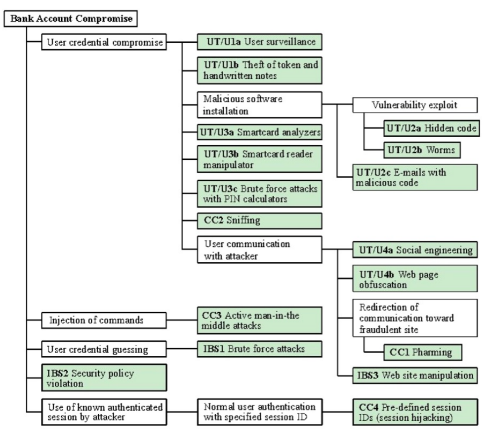


Figure 1.5 An Attack Tree for Internet Banking Authentication

Computer Security Strategy

Standards

- Standards have been developed to cover management practices and the overall architecture of security mechanisms and services
- The most important of these organizations are:
  - **National Institute of Standards and Technology (NIST)**
    - **NIST is a U.S. federal agency that deals with measurement science, standards, and technology related to U.S. government use and to the promotion of U.S. private sector innovation**
  - **Internet Society (ISOC)**
    - **ISOC is a professional membership society that provides leadership in addressing issues that confront the future of the Internet, and is the organization home for the groups responsible for Internet infrastructure standards**
  - **International Telecommunication Union (ITU-T)**
    - **ITU is a United Nations agency in which governments and the private sector coordinate global telecom networks and services**
  - **International Organization for Standardization (ISO)**
    - **ISO is a nongovernmental organization whose work results in international agreements that are published as International Standards**

CHP 2

CRYPTOGRAPHIC TOOLS

Symmetric Encryption

aka → conventional encryption
→ single key encryption

↳ provides confidentiality for transmitted or stored data

↳ has 2 requirements

1. Strong encryption algo

The algorithm must ensure that an attacker, even if they know
the algorithm and
have access to multiple ciphertexts and
their corresponding plaintexts,
cannot decipher the ciphertext or discover the key.

2. Secure key management

The sender and receiver must obtain the secret key
securely and
keep it protected.
If someone discovers the key and knows the algorithm,
they can read all communications encrypted with that key.

A symmetric encryption scheme consists of five key components:

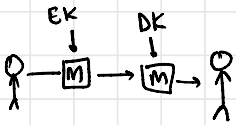
1.Plaintext:
The original message or data
that is input into the algorithm.

2.Encryption algorithm:
This algorithm applies various substitutions
and transformations to the plaintext.

3.Secret key:
This key is also input into the encryption algorithm
and determines how
↳ the substitutions and
↳ transformations
are performed.

4.Ciphertext:
The scrambled output generated by
the encryption algorithm, which depends on both
↳ the plaintext and
↳ the secret key.
Using different keys for the same message
will result in different ciphertexts.

5.Decryption algorithm:
This is the encryption algorithm used in reverse,
taking
↳ the ciphertext and
↳ the secret key
to produce the original plaintext.



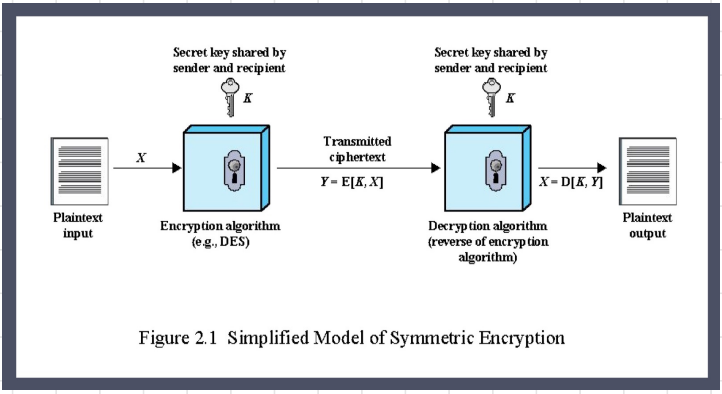


Figure 2.1 Simplified Model of Symmetric Encryption

Attacking Symmetric Encryption

Cryptanalytic Attacks

- they rely on
  - nature of algo
    - some knowledge on the characteristic of the plain text
    - some sample plaintext - ciphertext pairs
- it exploits the characteristics of the algo to try to deduce
  - specific plaintext
  - OR
  - key being used

if sucessful all future and past mssgs encrypted with that key are compromised

Brute Force Attacks

- trys all possible combination of keys on ciphertext until an readable plaintext is obtained
- on avg half of all possible keys must be tried to achive sucess

Block Ciphers

(most common)

- a symmetric encryption algo
- process the plain-text input in fixed sized blocks
- then produces a block of cipher text of equal size for each plaintext block

* the algo processes longer plaintext amounts as a series of fixed size blocks

3 types

1. Data Encryption Standard (DES)
2. Triple DES
3. Advance Encryption Standard (AES)

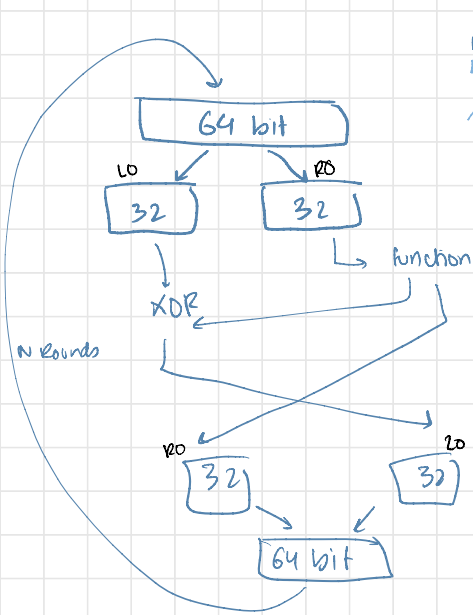
	DES	Triple DES	AES
Plaintext block size (bits)	64	64	128
Ciphertext block size (bits)	64	64	128
Key size (bits)	56	112 or 168	128, 192, or 256

DES = Data Encryption Standard
AES = Advanced Encryption Standard

Comparison of Three Popular Symmetric Encryption Algorithms

Fiestel cypher

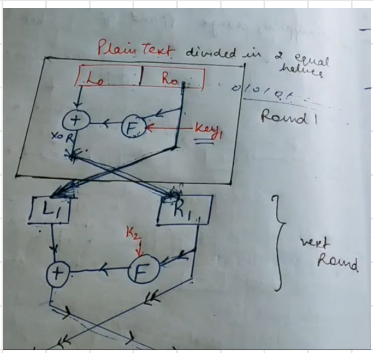
Sub in block cipher



Master key
Keys if 10 N rounds
then 10 subgenerated keys for every round

function (sub generated keys for main key)

N Rounds → if DES 16 rounds



Now,

1) **Block size** → Larger block size, more security

2) **key size** → Larger key size means more security but may decrease the speed of encryption/decryption.

3) **no. of rounds** → more rounds, more secure

4) **Subkey generation algo** → more complex algo, harder for attacker to steal data

5) **function / Round function F**
→ more complex fn, harder for the cryptanalyst to attack.

Data Encryption Standard (DES)

Until recently was the most widely used encryption scheme
- ↳ FIPS PUB 46
- ↳ Referred to as the Data Encryption Algorithm (DEA)
- ↳ Uses 64 bit plaintext block and 56 bit key to produce a 64 bit ciphertext block

Adv
1. Avalance effect
   ↳ small change in plain text equal big change in cipher text
2. Completeness effect
   ↳ each bit of cipher text depends on plain text

Concerns about DES's security can be divided into two categories:

Algorithm concerns:
There are worries that vulnerabilities might be found by analyzing the characteristics of the DES algorithm. Despite extensive study and attempts to identify weaknesses, no critical flaws have been reported.
w

Key length concerns:
The more pressing issue is the key length.
With only 56 bits, there are about 72 quadrillion possible keys. → approximately 2^{56}
Given modern processing speeds, this key length is inadequate.

Research suggests that
1.contemporary multicore computers can process around one billion key combinations per second.
2.modern Intel processors can process about half a billion encryptions per second.
3.supercomputers can process around ten trillion encryptions per second.

Key size (bits)	Cipher	Number of Alternative Keys	Time Required at 10^9 decryptions/s	Time Required at 10^{13} decryptions/s
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	2^{55} ns = 1.125 years	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	2^{127} ns = 5.3×10^{21} years	5.3×10^{17} years
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	2^{167} ns = 5.8×10^{33} years	5.8×10^{29} years
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	2^{191} ns = 9.8×10^{40} years	9.8×10^{36} years
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	2^{255} ns = 1.8×10^{60} years	1.8×10^{56} years

Average Time Required for Exhaustive Key Search
USING BRUTE FORCE

D.E.S

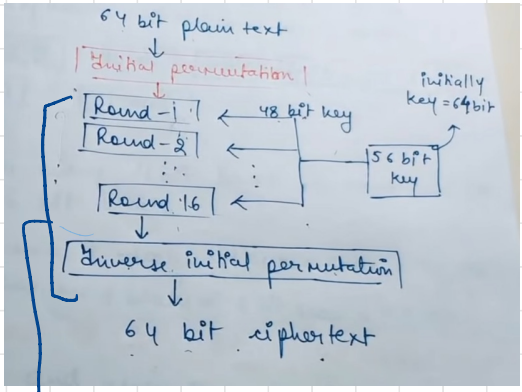
Data Encryption Standard

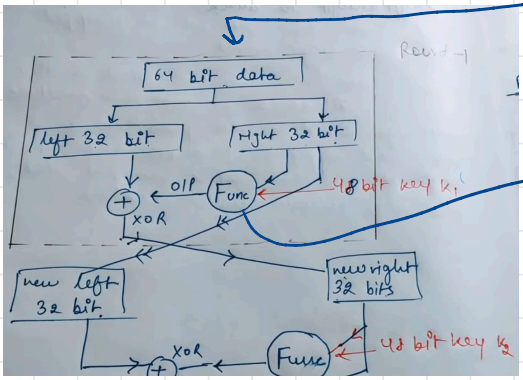
(i) block cipher
(ii) symmetric cipher (same key for encryption + decryption)
(iii) 64 bit plain text block
It encrypts the data in blocks of size 64 bits each
(iv) 16 rounds each round is a feistel round.

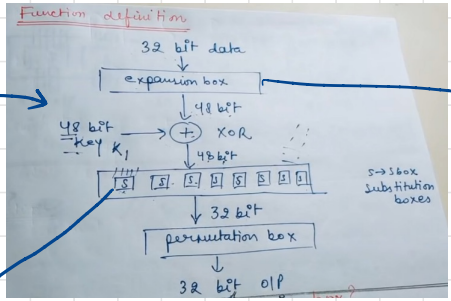
steps
(i) Initial permutation
(ii) 16 feistel rounds
(iii) swapping | left right swap
(iv) Final permutation | Inverse initial permutation

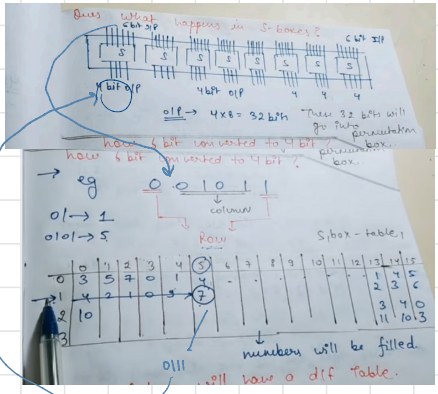
Basic structure

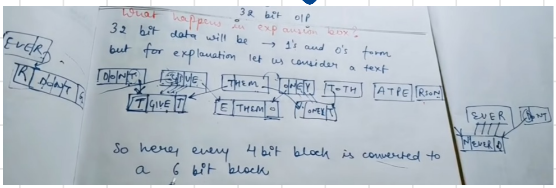
64 bit plain text
↓
Initial permutation
Initially key = 64 bit



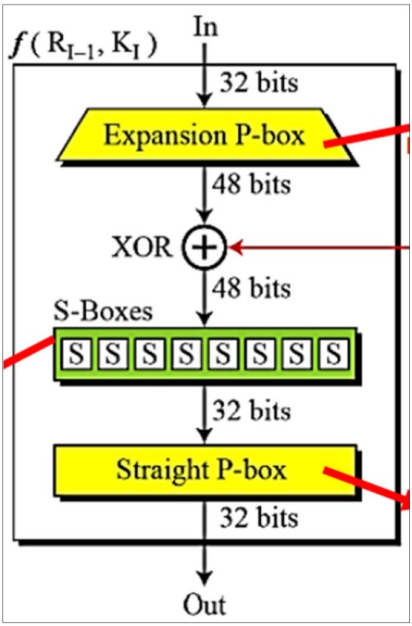


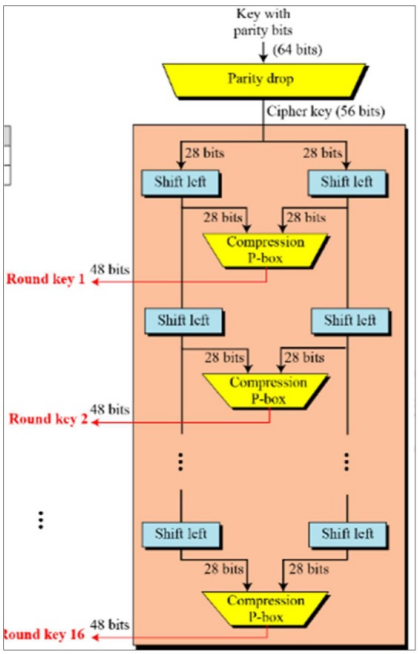






Note: Attempt this only if you think that you cannot score more than 5.5 marks in Q2.





----------(O)----------

Triple DES (3DES)

- Repeats basic DES algorithm three times using either two or three unique keys
- First standardized for use in financial applications in ANSI standard X9.17 in 1985

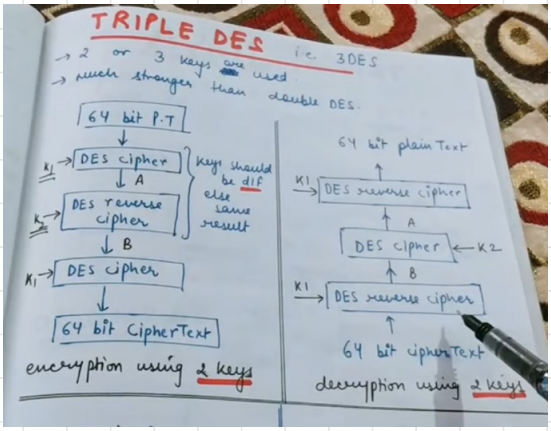
Adv

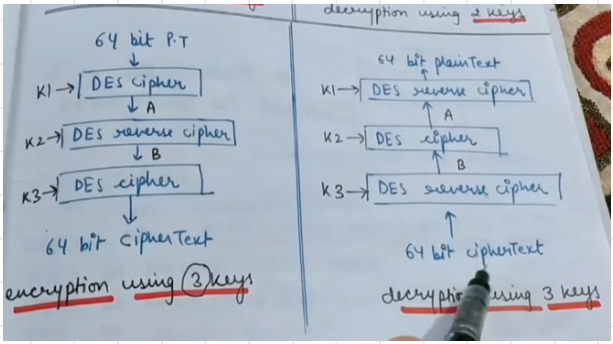
1. Increased key length to 168 bits
   - ↳ which significantly reduces vulnerability to brute force attacks compared to DES
2. Proven Algorithm
   - ↳ the encryption algo is same as in DES
   - ↳ this gives users confidence in security against attacks

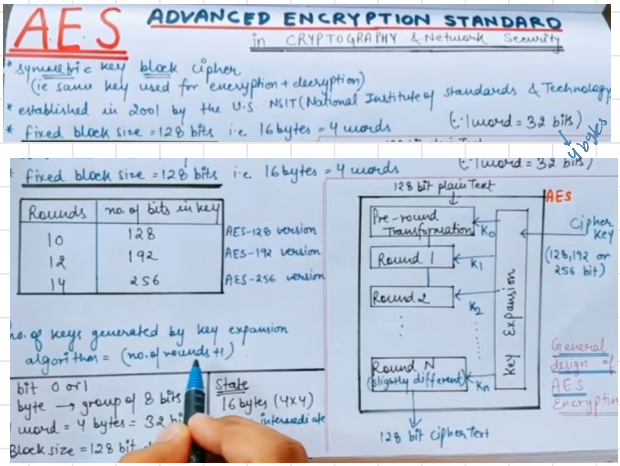
which has been thoroughly tested over time without any effective cryptanalytic attacks (other than brute-force

Con

1. Performance
   - ↳ the algo is relatively slow in software as it requires 3 times the calculation of DES
2. Small block size
   - ↳ both DES and 3DES use a block size of 64 bit which is not ideal for efficiency
   - ↳ a larger block size would be better
3. Not for long term use







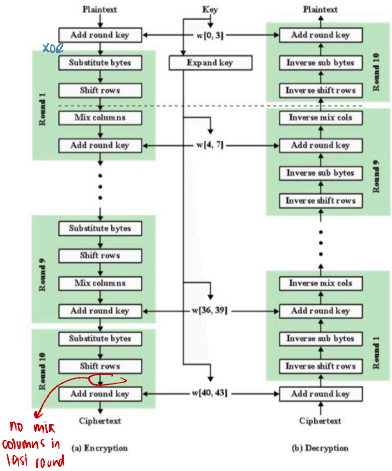
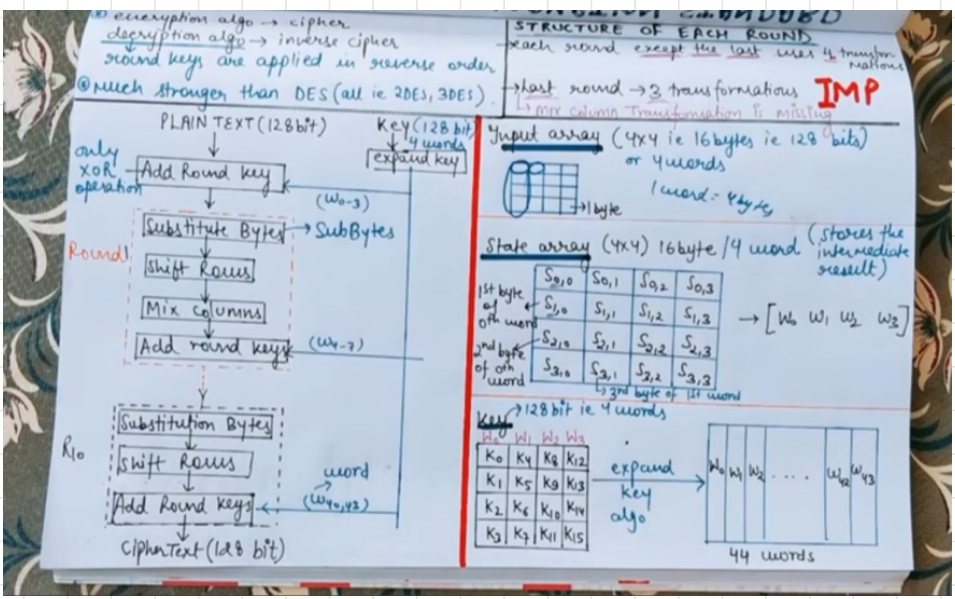
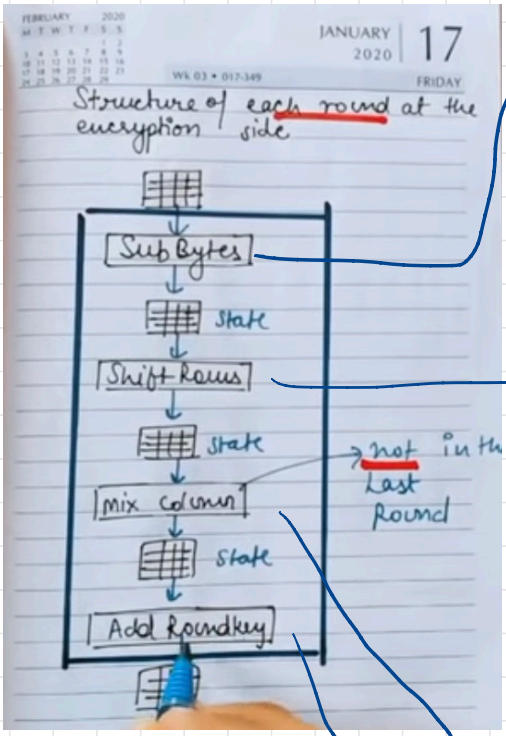
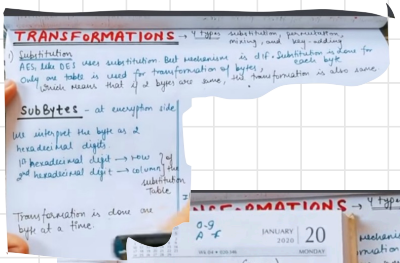
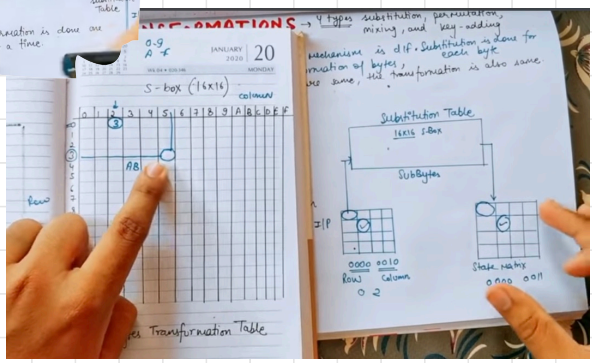


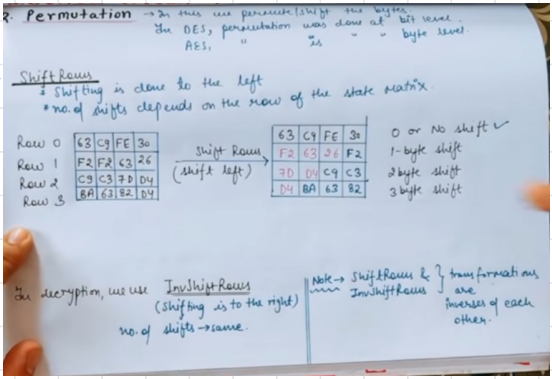
Figure 20.3 AES Encryption and Decryption

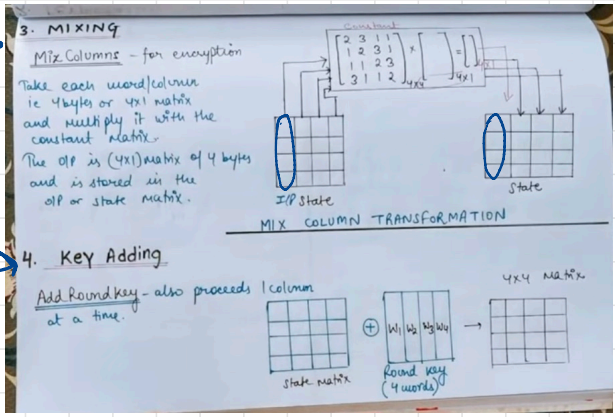



JANUARY 2020 17
FRIDAY
Structure of each round at the encryption side
SubBytes
State
Shift Rows
State
Mix Columns
not in the Last Round
State
Add Roundkey


TRANSFORMATIONS → 4 types substitution, permutation, mixing, and key-adding
1) Substitution
AES, like DES uses substitution but mechanism is diff. Substitution is done for each byte
Only one table is used for transformation of bytes; (which means that if 2 bytes are same, their transformation is also same.
SubBytes - at encryption side
We interpret the byte as 2 hexadecimal digits
1st hexadecimal digit → row
2nd hexadecimal digit → column
of the substitution Table
Transformation is done one byte at a time


S-box (16x16)
Substitution Table
16x16 S-Box
SubBytes
State matrix
Row
Column


2. Permutation → In this we permute/shift the bytes.
In DES, permutation was done at bit level.
AES, " " byte level.
ShiftRows
Shifting is done to the left
no. of shifts depends on the row of the state matrix
Row 0 63 C9 FE 30
Row 1 F2 F2 63 26
Row 2 C9 C3 7D D4
Row 3 BA 63 82 D4
Shift Rows (shift left)
63 C9 FE 30
F2 63 26 F2
7D D4 C9 C3
D4 BA 63 82
0 or No shift
1-byte shift
2 byte shift
3 byte shift
In decryption, we use InvShiftRows (Shifting is to the right)
no. of shifts → same.
Note → ShiftRows & InvShiftRows } transformations are inverses of each other.


3. MIXING
Mix Columns - for encryption
Take each word/column ie 4bytes or 4x1 matrix and multiply it with the constant matrix.
The O/P is (4x1) matrix of 4 bytes and is stored in the O/P or state matrix.
constant
I/P State
State
MIX COLUMN TRANSFORMATION
4. Key Adding
Add Roundkey - also proceeds 1 column at a time.
4x4 Matrix
State matrix
Round key (4 words)

Advanced Encryption Standard (AES)

Needed a replacement for 3DES

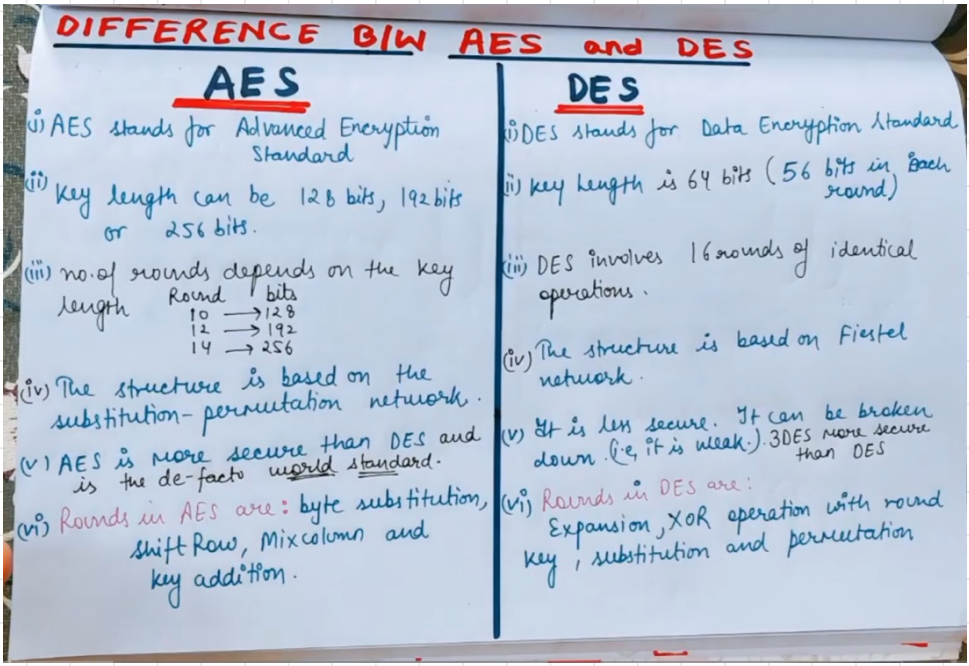
- 3DES was not reasonable for long term use

NIST called for proposals for a new AES in 1997

- Should have a security strength equal to or better than 3DES
- Significantly improved efficiency
- Symmetric block cipher
- 128 bit data and 128/192/256 bit keys

Selected Rijndael in November 2001

- Published as FIPS 197



Practical Security Issues

↳ Symmetric encryption handles larger data (greater than a single block: 64bit, 128bit; email message; network packets)
by breaking it into fixed length block for encryption

↓ simmplest approach

Electric Code Block

↳ each block of plaintext is
encrypted using the same key

CON

↳ not secure for lengthy messages
↳ as patterns in plain text can
be exploited by cryptoanalysis

eg. if certain fields are known to appear at the beginning of a message, the attacker can use known plaintext-ciphertext pairs to aid decryption.

↓ to overcome ECB weakness

Modes of Operations of Block cipher

↳ a no. of alternative techniques

Mode	Description	Typical Application
Electronic Codebook (ECB) ∪	Each block of 64 plaintext bits is encoded independently using the same key.	•Secure transmission of single values (e.g., an encryption key)
Cipher Block Chaining (CBC) ∪	The input to the encryption algorithm is the XOR of the next 64 bits of plaintext and the preceding 64 bits of ciphertext.	•General-purpose block-oriented transmission •Authentication
Cipher Feedback (CFB) ∪	Input is processed s bits at a time. Preceding ciphertext is used as input to the encryption algorithm to produce pseudorandom output, which is XORed with plaintext to produce next unit of ciphertext.	•General-purpose stream-oriented transmission •Authentication
Output Feedback (OFB)	Similar to CFB, except that the input to the encryption algorithm is the preceding DES output.	•Stream-oriented transmission over noisy channel (e.g., satellite communication)
Counter (CTR) √	Each block of plaintext is XORed with an encrypted counter. The counter is incremented for each subsequent block.	•General-purpose block-oriented transmission •Useful for high-speed requirements

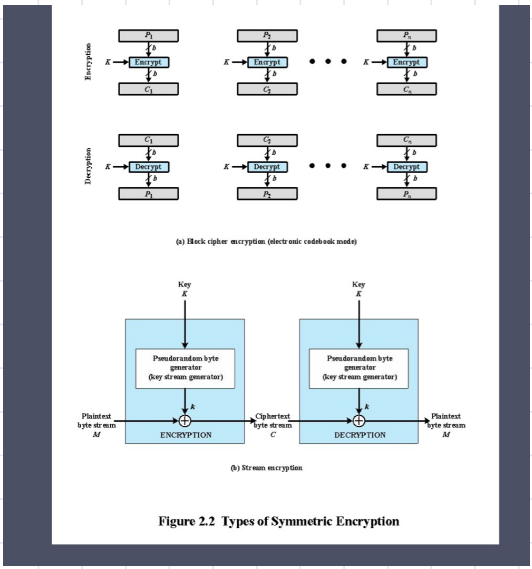
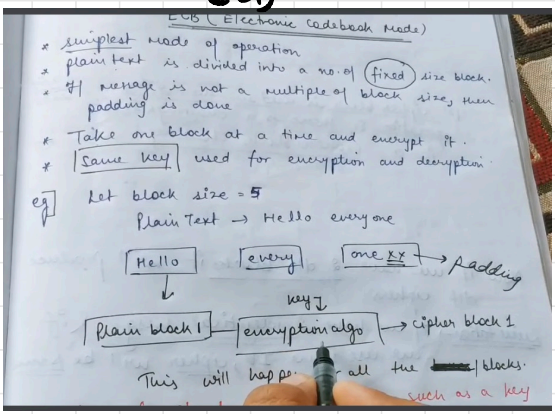
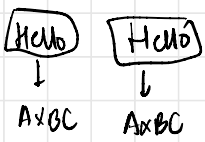


Figure 2.2 Types of Symmetric Encryption

ECB

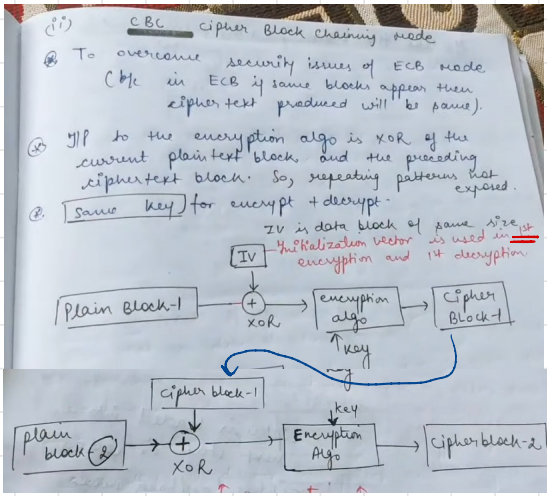




↳ same key for e and d

↳ same output

CBC

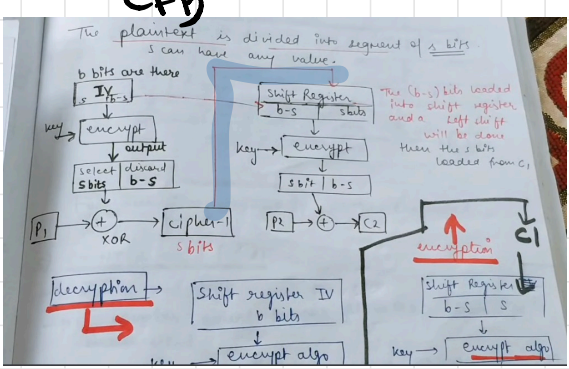


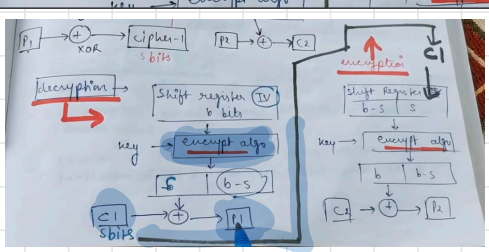
↳ NO same output

↳ same key for e and d

↳ if 2 same msgs, born use same IV then same output

CFB

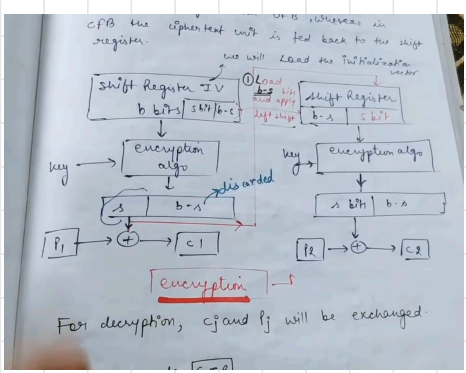




OFB

↳ same as CFB

↳ only take s bits before doing XOR



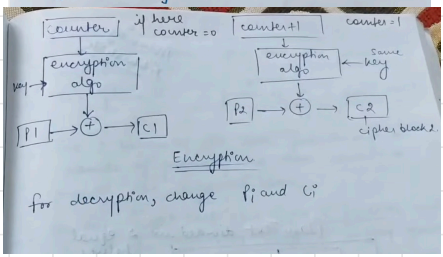
CTR

5) COUNTER MODE [CTR]

→ Simple and fast

→ a counter, equal to the plaintext block size is used.

* counter is initialised to some value and then incremented by 1 for each subsequent block.



TYPES OF SYMMECTRIC ENCRYPTION

Block Cipher

- ↳ Processess the input one block of elements at a time
- ↳ Produces an output block for each input block
- ↳ more common

PROS

- ↳ Can reuse keys

Prefered when encryptins
- ↳ file transfers
- ↳ email
- ↳ database

} involving blocks of data

Stream Cipher

- ↳ Processes the input elements continously
- ↳ Produces output one element at a time
- ↳ Encrypts plain text one byte at a time
- ↳ Pseudorandom stream → unpredictable, w/o knowledge of the input key

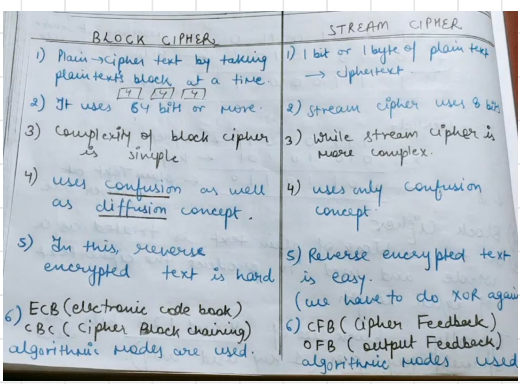
PROS

- ↳ uses lesser code
- ↳ almost always faster

Prefered when encryptins
- ↳ data communication
- ↳ web links

Message Authentication → aka Data Authentification

1. Protects against active attacks
   e.g. data falsification, transaction tampering
2. Verifies authencity of received messages
   - ↳ contents have not been altered
   - ↳ are from authentic source
   - ↳ timely and in correct seaquence
3. Can use Conventional encryption
   - ↳ only S (sender) and R (receiver) share a key



Block reordering is a threat

Symmetric encryption alone, particularly in ECB mode, is not suitable for data authentication. An attacker can reorder ciphertext blocks, allowing each to decrypt successfully, which may change the overall meaning of the data. While sequence numbers can be used at some levels (like in IP packets), they are generally not assigned to each block of plaintext, making block reordering a significant threat.

Types of Authentification

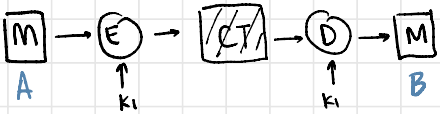
↳ verify users identity

1. Message Encryption
2. Message Authentification Code (MAC)
3. Hash functions

1. Message Encryption

↳ cipher text

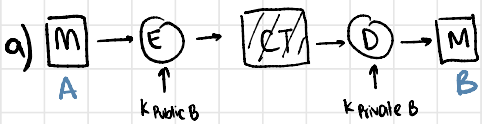
i) Symmetric Encryption



↳ K_1 is shared b/w S and R

ii) Assycmetric Encryption

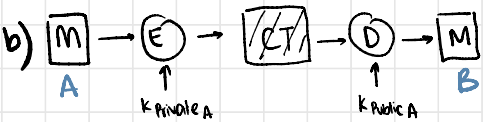
a)



↳ Authentification ✗ → B dosent know who is seading

↳ Confedentiality ✓

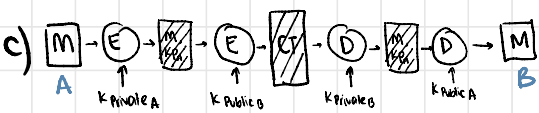
b)



↳ Authentification ✓

↳ Confedentiality ✗ → any one can open using A's Public key

c)



↳ Authentification ✓

↳ Confedentiality ✓

2. Message Authentication Code (MAC)

↳ uses a secret key

↳ to generate a fixed length code → MAC →

- It is then appended with the message.
- The communicating parties will share a secret common key. which will be used to create the MAC

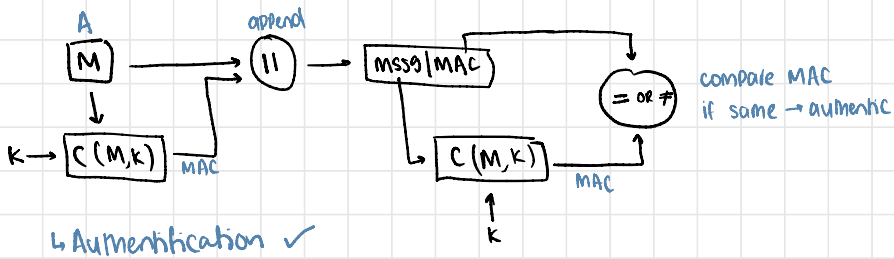
Let A → sender
B → receiver

When A sends a msg to B, it calculates the MAC as a fn of the message and the key.

$MAC = C(M, K)$

- C: Authentication function
- M: msg
- K: key

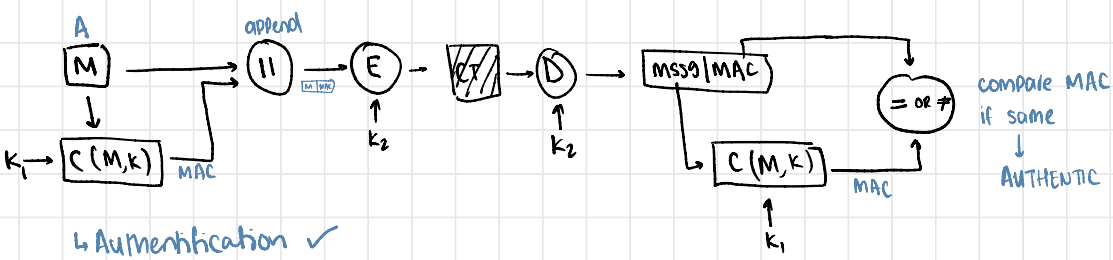
1.)



↳ Authentication ✓

↳ Confidentiality X

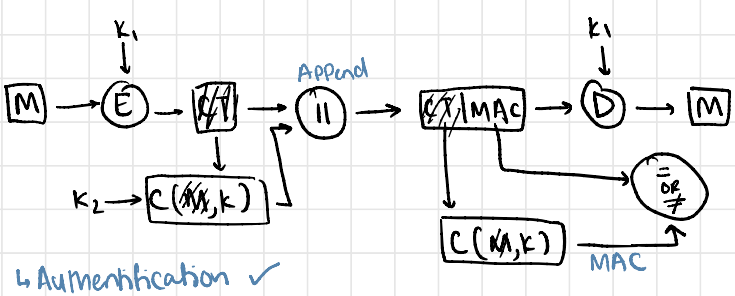
2.) Authentication tied to cipher text



↳ Authentication ✓

↳ Confidentiality ✓

3) Authentication tied to cipher text and K2



↳ Authentication ✓

↳ Confidentiality ✓

Message Authentication w/o Confidentiality

↳ the approaches discussed do not encrypt the mssg which means confidentiality is not provided

↳ gives authentification and confidentiality

Message encryption by itself does not provide a secure form of authentication
It is possible to combine authentication and confidentiality in a single algorithm
by encrypting a message plus its authentication tag
Typically, though, authentication is provided separately from encryption.

Situations where authentication without confidentiality is preferable include:

1.Broadcast Messages:
In cases like network notifications or alarm signals,
it's more efficient to broadcast the message
in plaintext with an authentication tag,
allowing a single destination to monitor authenticity.

2.High Load Exchanges:
When one party is overwhelmed,
they may selectively authenticate randomly chosen messages
instead of decrypting all incoming ones.

3.Program Authentication:
A computer program can be run in plaintext
to save processing resources,
with an authentication tag checked
only when integrity verification is needed.

Overall, both authentication and encryption play important roles in meeting security needs.

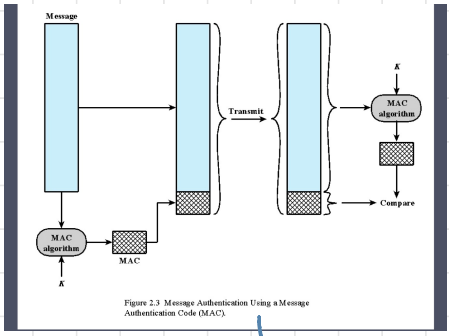
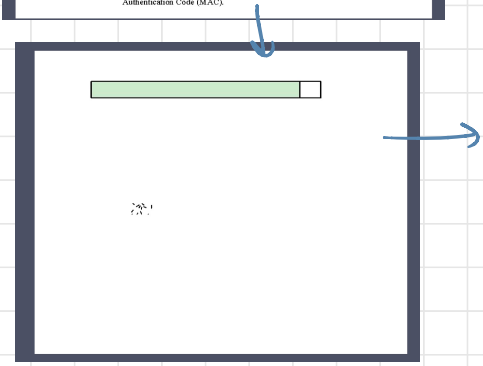


Figure 2.3 Message Authentication Using a Message Authentication Code (MAC).

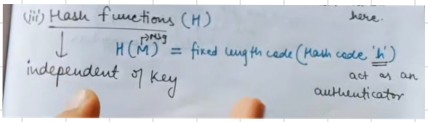


3. Hash Function (H)

↳ to produce a finger print of a
  ↳ file
  ↳ message
  ↳ other block of data

↳ to be useful for message authentication it must have the following **PROPERTIES**

*hashes are deterministic → predictable



msg authentication

1. can process data of any size
2. produces a fixed length output
3. H(x) is relatively easy to compute for any given x
4. One way or pre image resistant
   ↳ it should be computationally infeasible to find x such that H(x)=h
   → is easy to generate a code given a mssg but impossible to generate mssg given a code
   → so if the hacker sees the hash they can not reverse engineer the original data
5. it should be computationally infeasible to find $y \neq x$ such that H(y)= H(x)
   → prevents from forging mssg by finding alternate inputs with the same hash
6. Collision resistant or strong strong collision resistance
   ↳ it should be computationally infeasible to find pair(x,y) such that H(x)= H(y)
   → ensures 2 diff mssgs do not produce the same hash value

Security of Hash Functions

There are two approaches to attacking a secure hash function:

- Cryptanalysis
  - Exploit logical weaknesses in the algorithm
- Brute-force attack
  - Strength of hash function depends solely on the length of the hash code produced by the algorithm

SHA most widely used hash algorithm

Additional secure hash function applications:

- Passwords
  - Hash of a password is stored by an operating system
- Intrusion detection
  - Store H(F) for each file on a system and secure the hash values

HASH FUNCTIONS

- Similar to MAC (msg. authentication code)
  BUT it doesn't use a key.
- Takes in variable size message and produces a fixed output
  called Hash Code | Hash value | or Message Digest
- The only I/P is the message.
- A hash value h is generated by a fn H
  H(M) = fixed length code h
  variable length msg — hash code
- They are also called Compression Functions.
- There are dif methods to provide authentication in dif situations

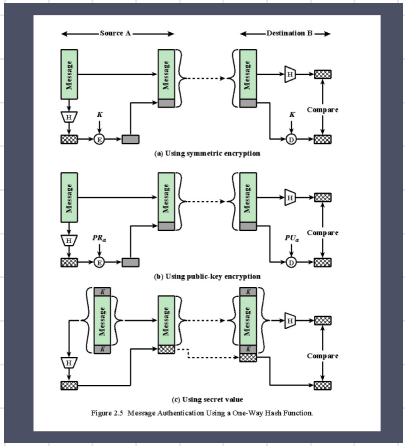
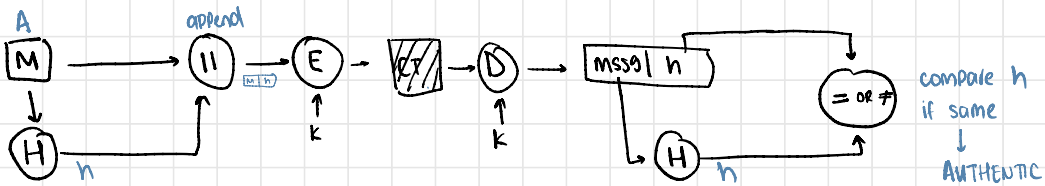


Figure 2.5 Message Authentication Using a One-Way Hash Function.

1)

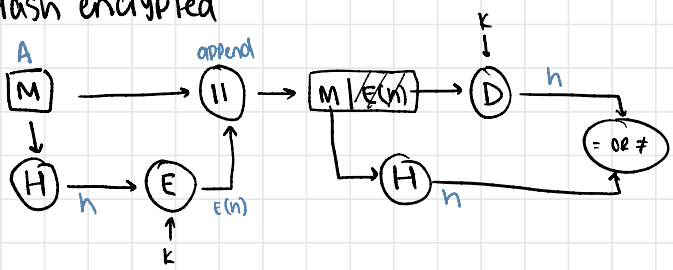


compare h
if same
↓
AUTHENTIC

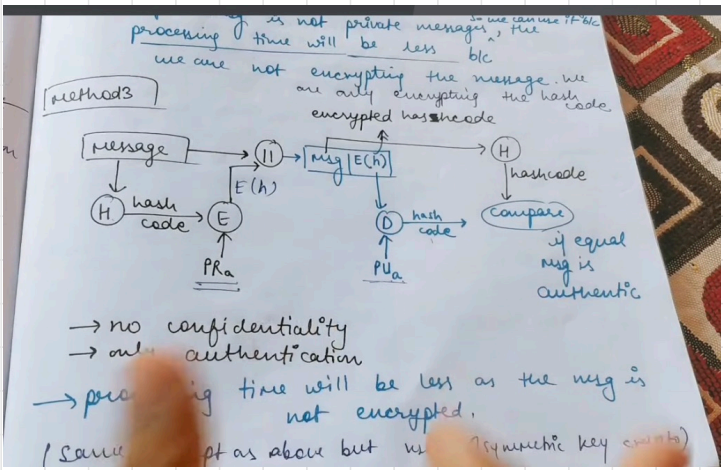
↳ Authentification ✓
↳ Confedentiality ✓

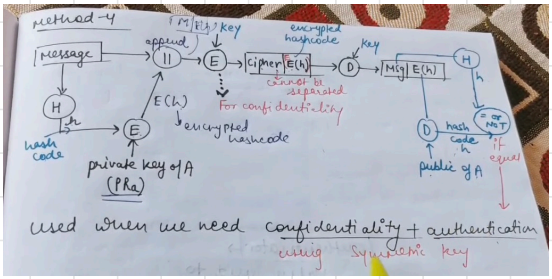
b/c only A & B share the secret key, the msg must have come from A & has not been altered.

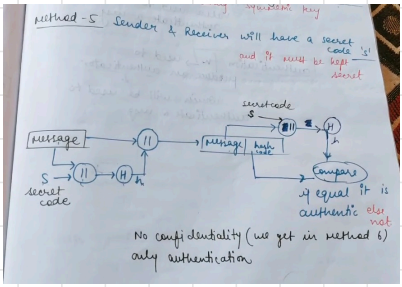
2) Hash encrypted

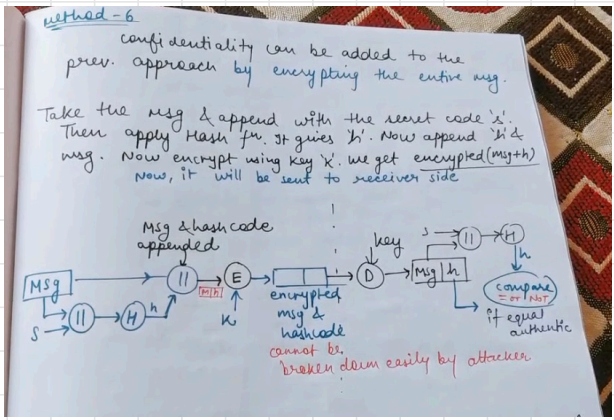


↳ Authentification ✓
↳ Confedentiality ✗


processing time will be less b/c
we are not encrypting the message. We are only encrypting the hash code
method3
encrypted hashcode
message
M | E(h)
hash code
E(h)
PRa
PUa
hashcode
compare
if equal msg is authentic
→ no confidentiality
→ only authentication
→ processing time will be less as the msg is not encrypted.


method-4
message
append
key
encrypted hashcode
Cipher | E(h)
cannot be separated
For confidentiality
Msg | E(h)
hash code
E(h)
encrypted hashcode
private key of A
(PRa)
public of A
if equal
used when we need confidentiality + authentication
using Symmetric key


method-5 Sender & Receiver will have a secret code 's' and it needs to be kept secret
secret code
message
hash code
compare
if equal it is authentic else not
No confidentiality (we get in method 6)
only authentication


method-6
confidentiality can be added to the prev. approach by encrypting the entire msg.
Take the msg & append with the secret code 's'. Then apply Hash fn. It gives 'h'. Now append 'h' & msg. Now encrypt using key 'k'. We get encrypted(msg+h)
Now, it will be sent to receiver side
Msg & hash code appended
encrypted msg & hashcode
cannot be broken down easily by attacker
Msg
compare
if equal authentic

Public key Encryption

↳ introduced by Diffie and Hellman in 1976

↳ Asymmetric Encryption

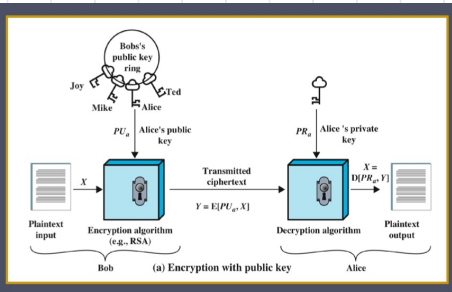
↳ It uses 2 seperate keys

1. Pubic key ⟶ encryption
   ↳ available to anyone
2. Private key ⟶ decryption
   ↳ confidential to owner

CONS

↳ computational overhead

↳ often requires complex protocols
   ↳ sometimes involving central authorities



- **Plaintext**
  - Readable message or data that is fed into the algorithm as input
- **Encryption algorithm**
  - Performs transformations on the plaintext
- **Public and private key**
  - Pair of keys, one for encryption, one for decryption
- **Ciphertext**
  - Scrambled message produced as output
- **Decryption key**
  - Produces the original plaintext

→ Directed towards confidentiality

1. Each user generates a pair of keys to be used for the encryption and decryption of messages.

2. Each user places one of the two keys in a public register or other accessible file. This is the public key. The companion key is kept private. As Figure 2.6a suggests, each user maintains a collection of public keys obtained from others.

3. If Bob wishes to send a private message to Alice, Bob encrypts the message using Alice's public key.

4. When Alice receives the message, she decrypts it using her private key. No other recipient can decrypt the message because only Alice knows Alice's private key.

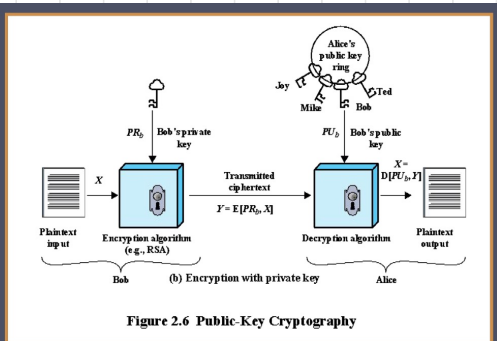


Figure 2.6 Public-Key Cryptography

- User encrypts data using his or her own private key
- Anyone who knows the corresponding public key will be able to decrypt the message

→ Directed towards authentification and integrity

If a user is able to successfully recover the plaintext from Bob's ciphertext using Bob's public key, this indicates that only Bob could have encrypted the plaintext, thus providing authentication.

Further, no one but Bob would be able to modify the plaintext because only Bob could encrypt the plaintext with Bob's private key.

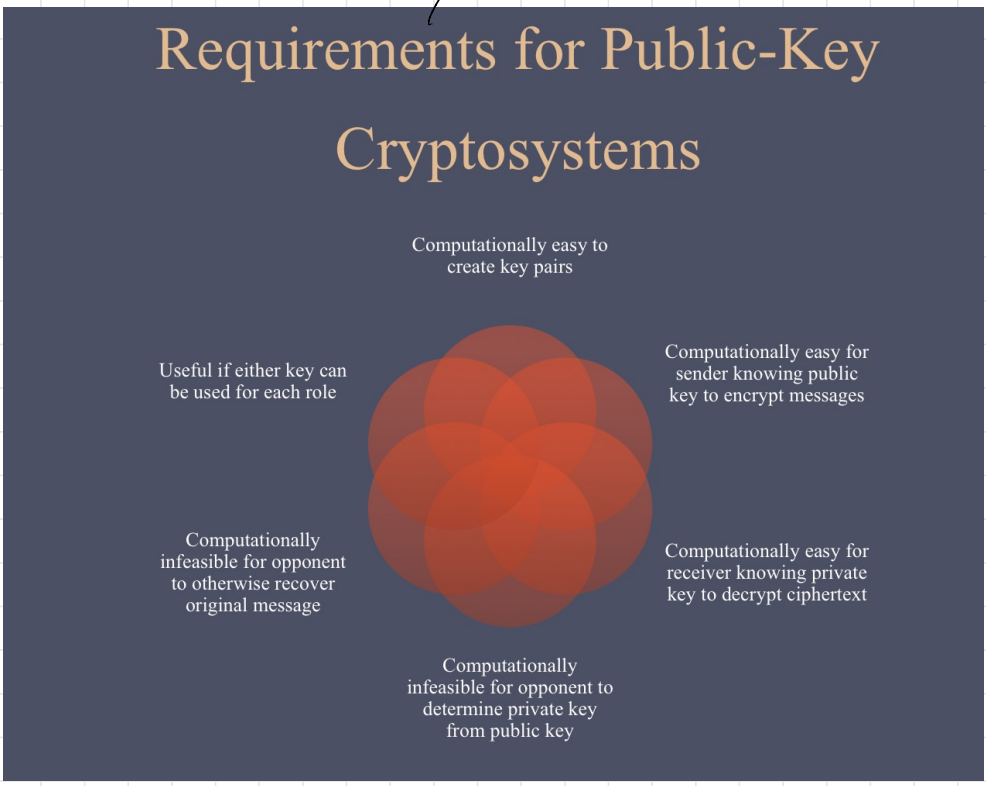
Categories for use of Public key cryptosystems

1. Digital Signature
2. Symmectric key distribution
3. Encryption of secret keys

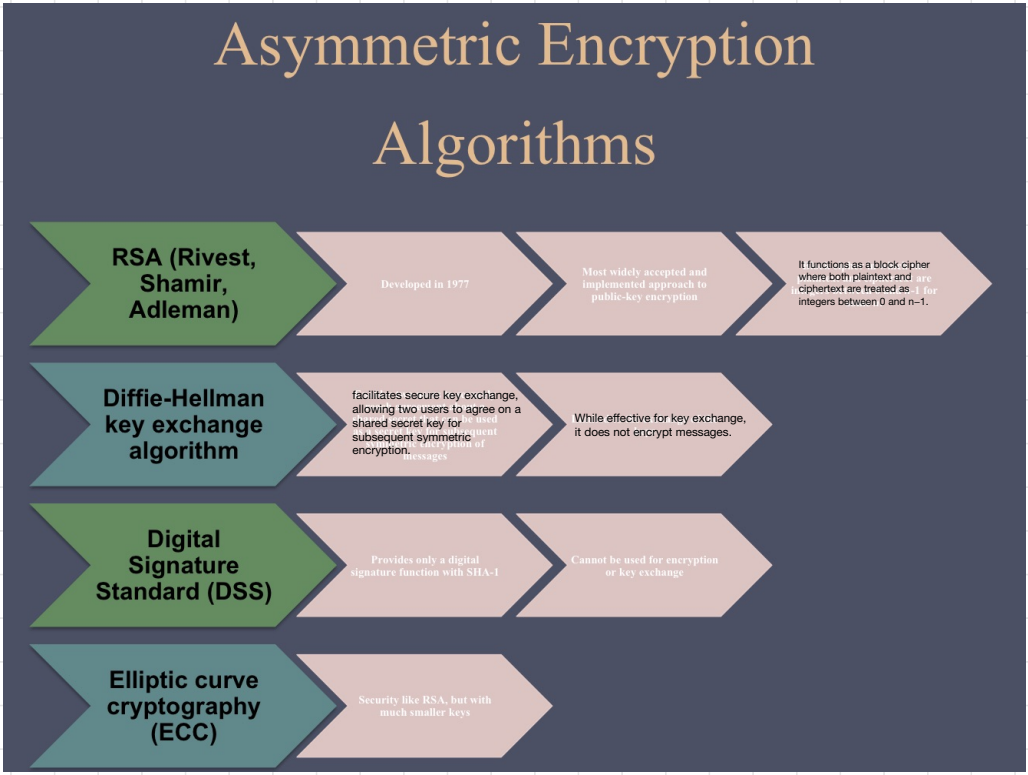
Applications for Public-Key Cryptosystems

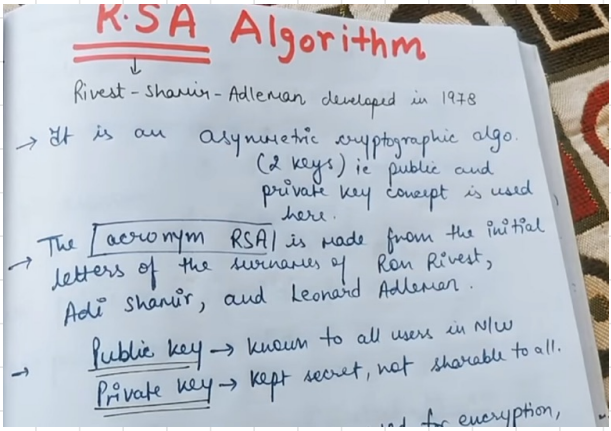
Algorithm	Digital Signature	Symmetric Key Distribution	Encryption of Secret Keys
RSA	Yes	Yes	Yes
Diffie-Hellman	No	Yes	No
DSS	Yes	No	No
Elliptic Curve	Yes	Yes	Yes

by Diffie and Hellman



Asymmetric Encryption Algorithms





RSA

↳ p and q given

↳ find n, k, e

↳ $n = p * q$ (always prime)

↳ $\phi = (p-1)(q-1)$

↳ $\gcd(e, \phi) = 1$ → using trial and error → find e (take smallest value)

↳ $d.e \bmod \phi \equiv 1$ → linear congruencies → find d

encryption: $c = m^e \bmod n$ (e given in question)

decryption: $m = c^d \bmod n$

} on each letter

Q) $p=7$, $q=17$

$n = 7 \times 17 = 119$

$\phi = (7-1)(17-1) = 96$

$\gcd(e, 96) = 1$

$(5, 96) = 1$

$e = 5$

$d \times e \bmod \phi = 1$

$5d \bmod 96 = 1$

$5 \times 19 \bmod 96 = 1$

$d = 19$

$c = m^5 \bmod 119$

$d = c^{19} \bmod 119$

Q) $p=3$, $q=11$

$n = 3 \times 11 = 33$

$\phi = (3-1)(11-1) = 20$

$\gcd(e, 20) \equiv 1$

$(7, 20) = 1$ → trial and error

$e = 7$

$d \times e \bmod \phi = 1$

$7d \bmod 20 = 1$

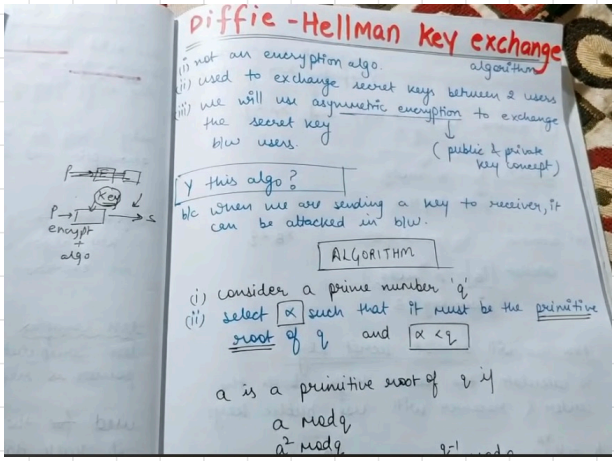
$7 \times 3 \bmod 20 = 1$ (21) → trial and error

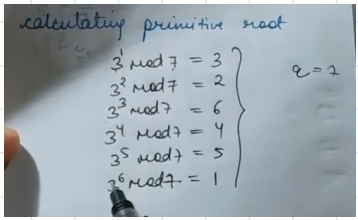
$d = 3$

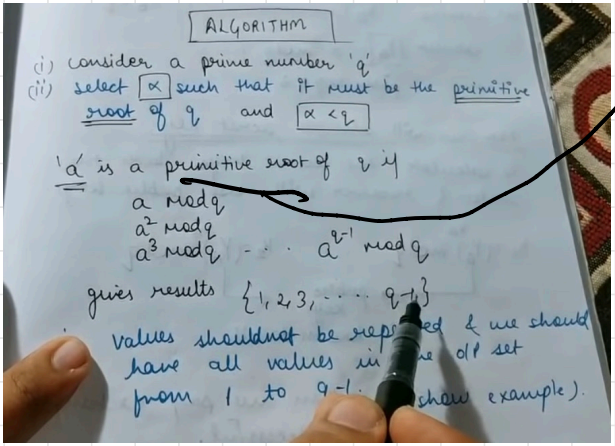
$c = m^7 \bmod 33$

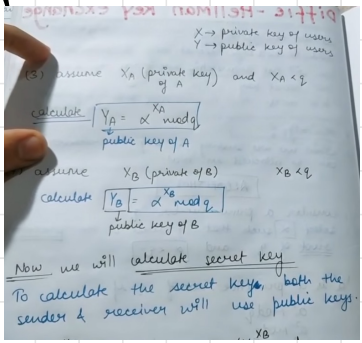
$d = c^3 \bmod 33$

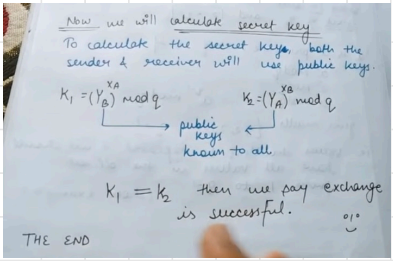
Delfie Hellman











Q) Diffie Hellman
$q = 7, X_A = 3, X_B = 4$
$\alpha = 5$

$Y_A = \alpha^{X_A} \bmod q$
$= 5^3 \bmod 7$
$= 6$

$Y_B = \alpha^{X_B} \bmod q$
$= 5^4 \bmod 7$
$= 2$

$K_1 = (Y_B)^{X_A} \bmod q$
$= 2^3 \bmod 7$
$= 1$

$K_2 = (Y_A)^{X_B} \bmod q$
$= 6^4 \bmod 7$
$= 1$

SAME
So exchanged

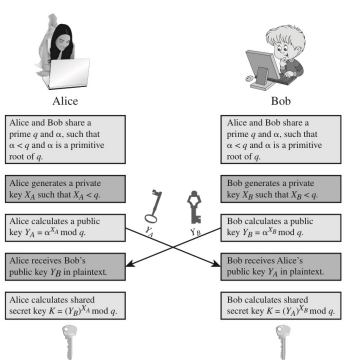


Figure 21.10 **Diffie-Hellman Key Exchange**

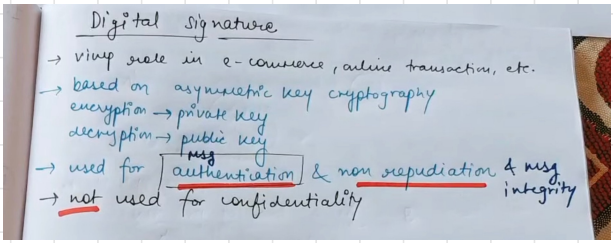
Digital Signatures

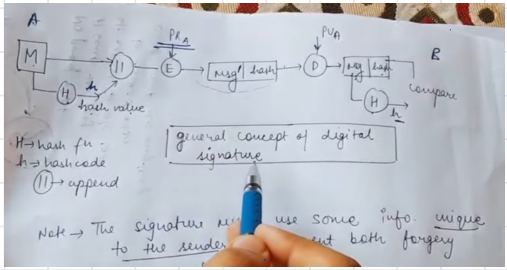
↳ It is a unique bit pattern generated based on a specific file / msg / data block

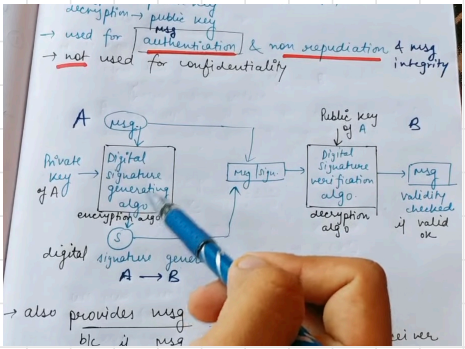
↳ It verifies 2 key aspects → assymetric

1. The data block was signed by the claimed signer → used for msg authentication and integrity
2. The data block has not been modified since it was signed → ensuring its integrity

↳ FIPS 186-4 outlines 3 approved DSA
1. Digital Signature Algorithm (DSA)
2. RSA Digital Signature
3. ECDSA







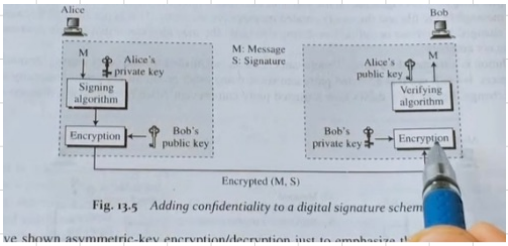
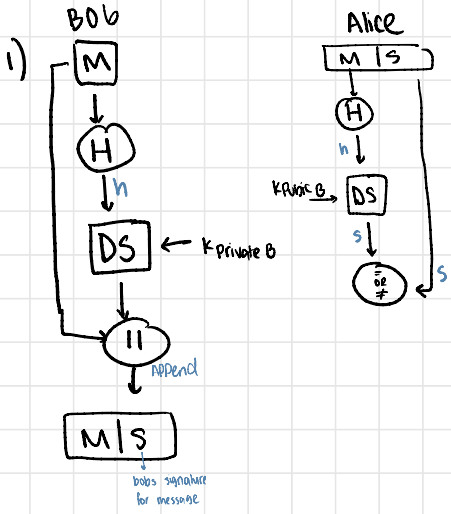
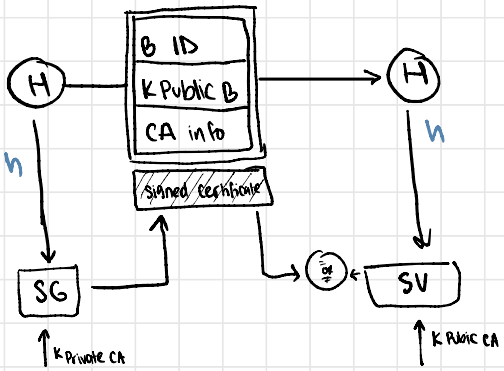


Fig. 13.5 Adding confidentiality to a digital signature scheme

1)



2)



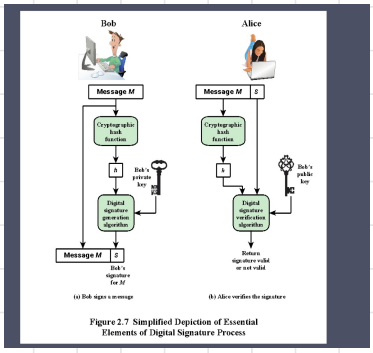


Figure 2.7 Simplified Depiction of Essential Elements of Digital Signature Process

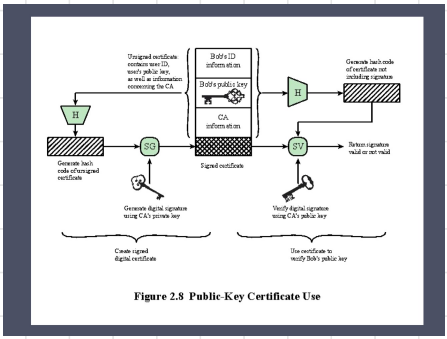
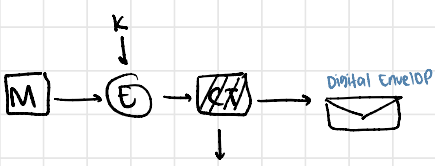


Figure 2.8 Public-Key Certificate Use

3)



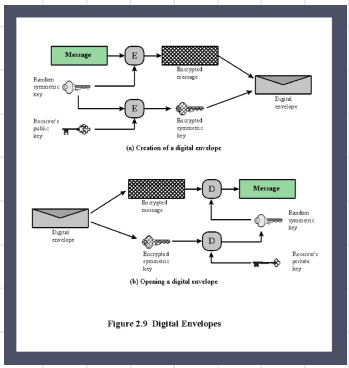


Figure 2.9 Digital Envelopes

RFC 4949 defines public-key infrastructure (PKI)
as the system of tools and rules used to
create,
manage, and
validate digital certificates
with public-key cryptography.

PKI's goal is to enable secure access to public keys.
To verify a certificate, you need the public key of the Certificate Authority (CA) that signed it, often organized in a hierarchy with a root CA at the top.
These CAs sign end-user certificates or delegate to intermediate CAs.

This model has problems.
1. it relies on users to understand and decide if a certificate should be trusted,
but most users lack the knowledge to make safe decisions, risking security.
2. all CAs in the trust store are assumed equally reliable, but breaches, like the DigiNotar compromise in 2011, show that not all are secure.
DigiNotar's breach enabled "man-in-the-middle" attacks, leading to its removal from trust stores and bankruptcy.
Other CAs, like Comodo, have also faced security breaches.
Furthermore, different browsers and systems use different trust stores, giving users inconsistent security levels.

To address these issues, various proposals have emerged.
1. Some recognize that users don't always need identity verification but need assurance of continuity—that the site and key they're accessing remain the same over time and match what other users see.
2. One solution, certificate pinning, helps browsers confirm the site's consistency over time, as seen in Google Chrome and Firefox's "Certificate Patrol."
3. Another approach, with tools like the Perspectives Project, uses network notary servers to confirm that a certificate matches what others see at different times and locations. Systems like Google Certificate Transparency also track certificate use over time.
4. Noting any unexpected certificate changes can signal a security issue, although sometimes changes are due to legitimate updates.

Approaching certificate expiry or issues with organizations using multiple certificates for the same server need management by extensions like notary systems.

The IETF's Public Key Infrastructure X.509 (PKIX) working group created a standard model for using certificates online, based on X.509.
Key elements in the PKIX model include:

End Entity: The user or server that receives a certificate.
Certificate Authority (CA): Issues certificates.
Registration Authority (RA): Manages user registration.
CRL Issuer/Repository: Manages certificate revocation lists (CRLs).

PKIX defines various management functions,
such as user registration, key initialization, certification by CAs, key recovery, certificate revocation, and cross-certification between CAs.
These functions may require support from specific management protocols.

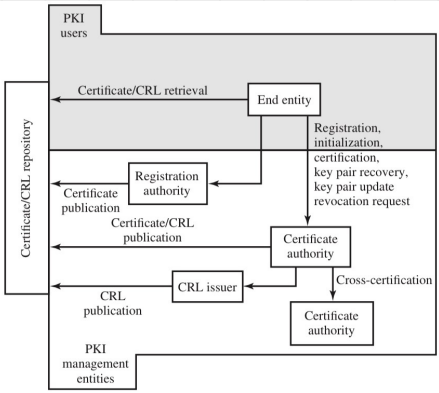


Figure 23.4 **PKIX Architectural Model**

RANDOM NUMBERS

Random Numbers

Uses include generation of:

- Keys for public-key algorithms
- Stream key for symmetric stream cipher
- Symmetric key for use as a temporary session key or in creating a digital envelope
- Handshaking to prevent replay attacks
- Session key

Random Number Requirements

Randomness

- Criteria:
  - Uniform distribution
    - Frequency of occurrence of each of the numbers should be approximately the same
  - Independence
    - No one value in the sequence can be inferred from the others

Unpredictability

each number should be unpredictable based on the previous number

- Each number is statistically independent of other numbers in the sequence
- Opponent should not be able to predict future elements of the sequence on the basis of earlier elements

Pseudorandom Numbers

↳ deterministic algo

↳ creates a sequence of numbers

that aren't truly random

but can pass tests for randomness

↳ likely to be predictable

TRNG Numbers

The Random Number Generator

↳ non deterministic

True random number generator (TRNG):

- Uses a nondeterministic source to produce randomness
- Most operate by measuring unpredictable natural processes
  - e.g. radiation, gas discharge, leaky capacitors
- Increasingly provided on modern processors

Practical Application: Encryption of Stored Data

Common to encrypt transmitted data

Much less common for stored data

There is often little protection beyond domain authentication and operating system access controls

Data are archived for indefinite periods

Even though erased, until disk sectors are reused data are recoverable

Approaches to encrypt stored data:

- Use a commercially available encryption package
- Back-end appliance
- Library based tape encryption
- Background laptop/PC data encryption

Commercial Encryption Packages: Programs like Pretty Good Privacy (PGP) allow users to create a key from a password to encrypt files on their hard drives. PGP does not store the password; instead, it generates it to decrypt files, ensuring strong protection as long as users choose secure passwords.

Back-end Appliances: These hardware devices sit between servers and storage systems, encrypting data sent to storage and decrypting data retrieved from it. They operate with minimal latency and are more efficient than encryption software, which can slow down processes.

Library-based Tape Encryption: This involves a co-processor board within tape drives that encrypts data with a non-readable key. The encrypted tapes can be securely transported, and the key can be shared securely for decryption at a different facility.

Background Data Encryption: Various vendors offer software that provides transparent encryption for files, folders, or entire disks. Solutions like Windows BitLocker and MacOS FileVault encrypt entire drives or images, with key management solutions in place to ensure that only the data owner can access the information.

Summary

- Confidentiality with symmetric encryption
  - Symmetric encryption
  - Symmetric block encryption algorithms
  - Stream ciphers
- Message authentication and hash functions
  - Authentication using symmetric encryption
  - Message authentication without message encryption
  - Secure hash functions
  - Other applications of hash functions
- Random and pseudorandom numbers
  - The use of random numbers
  - Random versus pseudorandom
- Public-key encryption
  - Structure
  - Applications for public-key cryptosystems
  - Requirements for public-key cryptography
  - Asymmetric encryption algorithms
- Digital signatures and key management
  - Digital signature
  - Public-key certificates
  - Symmetric key exchange using public-key encryption
  - Digital envelopes
- Practical Application: Encryption of Stored Data