TY BTech Semester-V (AY 2022-23)
Computer Science and Engineering

Disclaimer:

a. Information included in these slides came from multiple sources. We have tried our best to cite the sources. Please refer to the references to learn about the sources, when applicable.

b. The slides should be used only for preparing notes, academic purposes (e.g. in teaching a class), and should not be used for commercial purposes.

Information and Cyber Security

Examination Scheme:

Class Continuous Assessment: 60 Marks Lab Continuous Assessment: 50 Marks End Semester Examination: 40 Marks Credit: 3+1

Course Objectives:

Course Objectives:

1. Knowledge

(i) To focus on the models, tools, and techniques for enforcement of security with some emphasis on the use of cryptography. Students will learn security from multiple perspectives.

(ii) To educate students on the fundamental principles and techniques of computer and network security.

2. Skills (i) Acquire background on hash functions; authentication; firewalls; intrusion detection techniques

(ii) Gain hands-on experience with programming and simulation techniques for security protocols

3. Attitude (i) Understand the tradeoffs and criteria/concerns for security countermeasure development.

(ii) Learn to apply methods for authentication, access control, intrusion detection and prevention

Course Outcomes:

- Analyze and resolve security issues in networks and computer systems to secure an IT infrastructure.
- Apply methods for authentication, access control, intrusion detection and prevention
- Develop policies and procedures to manage enterprise security risks.
- Evaluate and communicate the human role in security systems with an emphasis on ethics, social engineering vulnerabilities and training.
- Identify software security vulnerabilities, summarize and mitigate security risks associated with integrating systems.

Pre-requisites

- *Operating Systems and Computer Networks*

Syllabus

Unit: I	**Foundations of Information Security** Information Security fundamentals, it’s need, Confidentiality, Integrity, Availability (CIA triad), Security Policies, Procedures, Guidelines, Standards Administrative Measures and Technical Measures, Attacks, Vulnerability, Security Goals, Security Services and Defence mechanisms **Cryptographic Techniques** Conventional substitution and transposition ciphers, One-time Pad, Block cipher and Stream Cipher, Cipher modes of operations Steganography. Symmetric Cryptographic Techniques: DES, AES	**9 Hrs**

Syllabus (Continue)

Books:- (Text)	**1. Cryptography and Network Security, William Stallings, Pearson Education 5th Edition, ISBN 13: 978-0-13-609704-4** **2. Computer Security: Principles and Practices, Willaim Stallings and Lawrie Brown, Pearson Education, ISBN 13-9780134794396**
Books:- (Reference)	1. Cryptography and Network Security, Berouz Forouzan 2 edition, TMH, ISBN :9780070702080 2. Applied Cryptography, Bruice Schneier, 2nd Edition, Wiely India Pvt Ltd, ISBN 978-81-265-1368-0 3. Computer Security: Art and Science, by Matt Bishop, Pearson Education, ISBN:9788177584257

Supplementary Reading:

1. E-books
2. Web links
3. MOOCs

Unit-I

Foundations of Information Security & Cryptographic Techniques

Foundations of Information Security

❖ **Cyber security** or **information security** are the techniques of protecting computers, networks, programs and data from unauthorized access or attacks that are aimed for exploitation.

Benjamin Franklin once said

Three people can keep a secret…....

…………. if two of them are dead!

Security is Not Easy to Achieve:

- Human tendency
- Problems of storage and communication
- Trust in all the parties

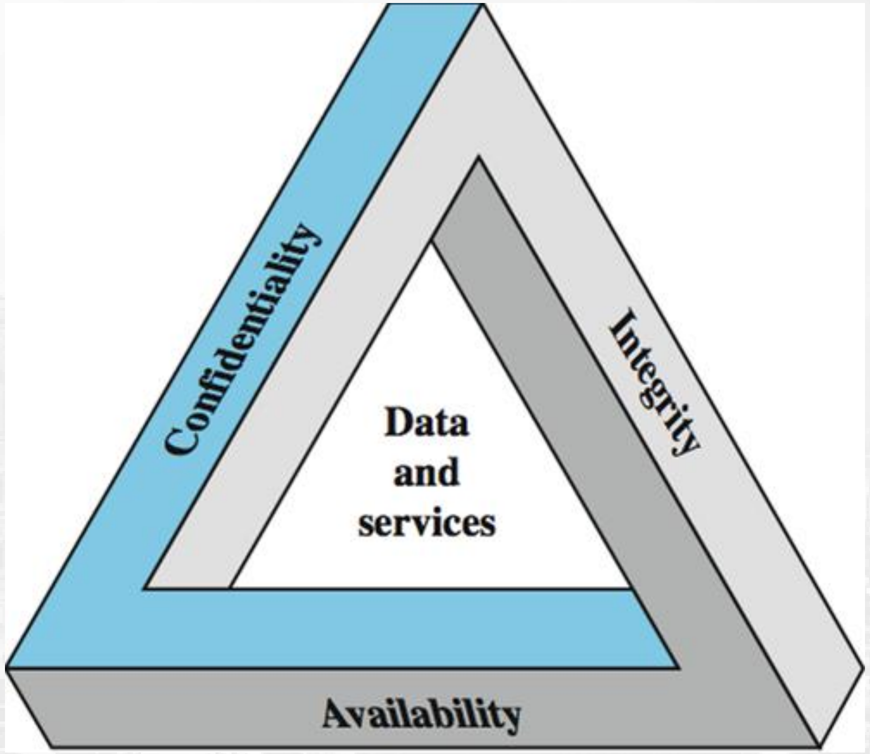
Don’t forget these roots !!

Attacks -- Services -- Defense

?

Key Security Concepts

- **Elements of Information Security**
  - Confidentiality: authorised user can access data
  - Integrity: validity of data
  - Availability



Security Policies, Procedures, Guidelines…

- **What is a security policy?**
- A security policy is a document that states in writing how a company plans to protect its physical and information technology (IT) assets. Security policies are living documents that are continuously updated and changing as technologies, vulnerabilities and security requirements change.
- A company's security policy may include an acceptable use policy. These describe how the company plans to educate its employees about protecting the company's assets. They also include an explanation of how security measurements will be carried out and enforced, and a procedure for evaluating the effectiveness of the policy to ensure that necessary corrections are made.

Security Policies, Procedures, Guidelines…

- **Why are security policies important?**
- Security policies are important because they protect an organizations' assets, both physical and digital. They identify all company assets and all threats to those assets.
- Physical security policies are aimed at protecting a company's physical assets, such as buildings and equipment, including computers and other IT equipment. Data security policies protect intellectual property from costly events, like data breaches and data leaks.

Security Policies, Procedures, Guidelines…

- **Physical security policies**
- Protecting IT physical assets is particularly important because the physical devices contain company data. If a physical IT asset is compromised, the information it contains and handles is at risk. In this way, information security policies are dependent on physical security policies to keep company data safe.
- Physical security policies include the following information:
- sensitive buildings, rooms and other areas of an organization;
- who is authorized to access, handle and move physical assets;
- procedures and other rules for accessing, monitoring and handling these assets; and
- responsibilities of individuals for the physical assets they access and handle.
- Security guards, entry gates, and door and window locks are all used to protect physical assets. Other, more high-tech methods are also used to keep physical assets safe. For example, a biometric verification system can limit access to a server room.

Security Policies, Procedures, Guidelines…

- **Information security policies**
- These policies provide the following advantages.
- **Protect valuable assets.** These policies help ensure the confidentiality, integrity and availability -- known as the *CIA triad* -- of data. They are often used to protect sensitive customer data and personally identifiable information.
- **Guard reputations.** Data breaches and other information security incidents can negatively affect an organization's reputation.
- **Ensure compliance with legal and regulatory requirements.** Many legal requirements and regulations are aimed at security sensitive information. For example, Payment Card Industry Data Security Standard dictates how organizations handle consumer payment card information. Health Insurance Portability and Accountability Act details how companies handle protected health information. Violating these regulations can be costly.

Security Policies, Procedures, Guidelines…

- **Dictate the role of employees.** Every employee generates information that may pose a security risk. Security policies provide guidance on the conduct required to protect data and intellectual property. **Identify third-party vulnerabilities.** Some vulnerabilities stem from interactions with other organizations that may have different security standards. Security policies help identify these potential security gaps.

Policy

- Set of detailed rules as to what is allowed on the system and what is not allowed.
- User Policy
- System Policy
- Network Policy
- US Law
- Trust

Policy Making

Formulations:

- General “catch-all” policy
- Specific asset-based policy
- General policy, augmented with standards and guidelines

Role:

- Clarify what and why of protection
- State responsibility for protection
- Provide basis for interpreting and resolving conflicts
- Retain validity over time

Standards & Guidelines

- Standards:
  - Codification of successful security practice
  - Platform-independent, enforceable
  - Change over time (slowly)
- Guidelines:
  - Interpret standards for particular environment
  - May be violated if needed

Building Policy

- Assign an owner
- Be positive
  - Motivate behavior
  - Allow for error
- Include education
- Place authority with responsibility
- Pick basic philosophy
  - Paranoid
  - Prudent
  - Permissive
  - Promiscuous
- Don’t depend on “impossible to break”

Security Through Obscurity

- If we don’t tell them, they won’t know (false)
  - Found by experimentation
  - Found through other references
  - Passed around by word of mouth
- Often used as basis for ignoring risks
- Local algorithm, unavailable sources - no real security

Going Public

- Vendor / CERT-CC (Computer Emergency Response Team-Coordination Center)
- Other Administrators (Warning)
- User community (Danger)
- Internet community (Infectious Danger)

User-level Policy

- Authentication: Method, Protection, Disclosure
- Importing software: Process, Safeguards, Location
- File protection: Default, Variations
- Equipment management: Process, Physical Security
- Backups: How, When
- Problem reporting: Who, How, Emergencies

System-level Policy

- Default configuration
- Installed Software
- Backups
- Logging
- Auditing
- Updates
- Principle servers or clients

Network-level Policy

- Supported services
- Exported services: Authentication, Protection, Restriction
- Imported services: Authentication, Protection, Privacy
- Network security mechanisms

US Law

- General advice - not legal counsel
- Before performing legal actions -- consult a lawyer!
- Legal Options
- Legal Hazards
- Being the target of an investigation
- General Tips
- Civil Actions
- Intellectual Property
- Liability

Legal Options

- Think before you pursue legal action
- Civil actions
- Reasons to prosecute:
  - Filing insurance claim
  - Involved with privacy data
  - Avoid being an accessory to later break-ins
  - Avoid civil suit with punitive damages
  - Avoid liability from your users

Legal Hazards

- Computer-illiterate agents
- Over-zealous compliance with search order
- Attitude and behavior of investigators
  - Work loss
  - Problems from case
  - Problems with working relationships
- Publicity loss
- Seizure of equipment
- Positive trend in enforcement community

Being the Target

- COOPERATE
- Individual involvement:
  - Document level of authorized access
  - Limit level of seizure, prosecution
- Officers will seize everything related to unauthorized use
- Wait for return can be very long
- Can challenge reasons for search
- Involve legal help soonest!

General Tips (1)

- Replace welcome messages with warning messages
- Put ownership or copyright notices on each source file
- Be certain users are notified of usage policy
- Notify all users on what may be monitored
- Keep good backups in safe location
- When you get suspicious, start a diary/journal of observations

General Tips (2)

- Define, in writing, authorization of each user and employee & have them sign it
- Ensure employees return equipment on termination
- Do not allow users to conduct their own investigations
- Make contingency plans with lawyer and insurance
- Identify qualified law enforcement at local, federal

Lawsuits

- Can sue anyone for any reasonable claim of damages or injury
- Caveats:
  - Very expensive
  - Long delays
  - May not win
  - May not collect anything
- Vast majority of actions -- settled out of court
- CONSULT A LAWYER FIRST

Intellectual Property

- Copyright infringement
  - Expression of idea
  - Derivative work
  - Outside of fair use
- Trademark violation
  - Use of registered words, symbols, phrases
  - Lack of credit
- Patent concerns
  - Application of idea
  - Based on prior art
  - Prevents redundant application

Liability

- Personal liability
- Corporate liability
- Good security helps to limit liabilities

Trust

- Tools of computer security are resident on computers
- Just as mutable as any other information on computers
- Can we trust our computer?
- Can we trust our software?
- Can we trust our suppliers?
- Can we trust our people?
- Trust, but verify

Trusting Our Computer

- Hardware bugs
- Hardware features
- Peripheral bugs/features
- Microcode problems

Trusting Our Software

- Operating system bugs and features
- System software back-doors
- Who wrote the software?
- Who maintains the software?
- Is GOTS / COTS trustworthy?

Trusting Our Suppliers

- Development process
- Bugs
- Testing
- Configuration control
- Distribution control
- Hacker challenges

Trusting Our People

- Vendors
- Consultants
- Employees
- System administrators
- Response personnel

Trust, but Verify

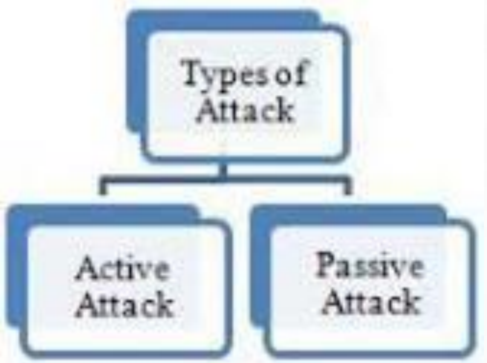
- Trust with a suspicious attitude
- Ask questions
- Do background checks
- Test code
- Get written assurances
- Anticipate problems and attacks

Aspects of Security

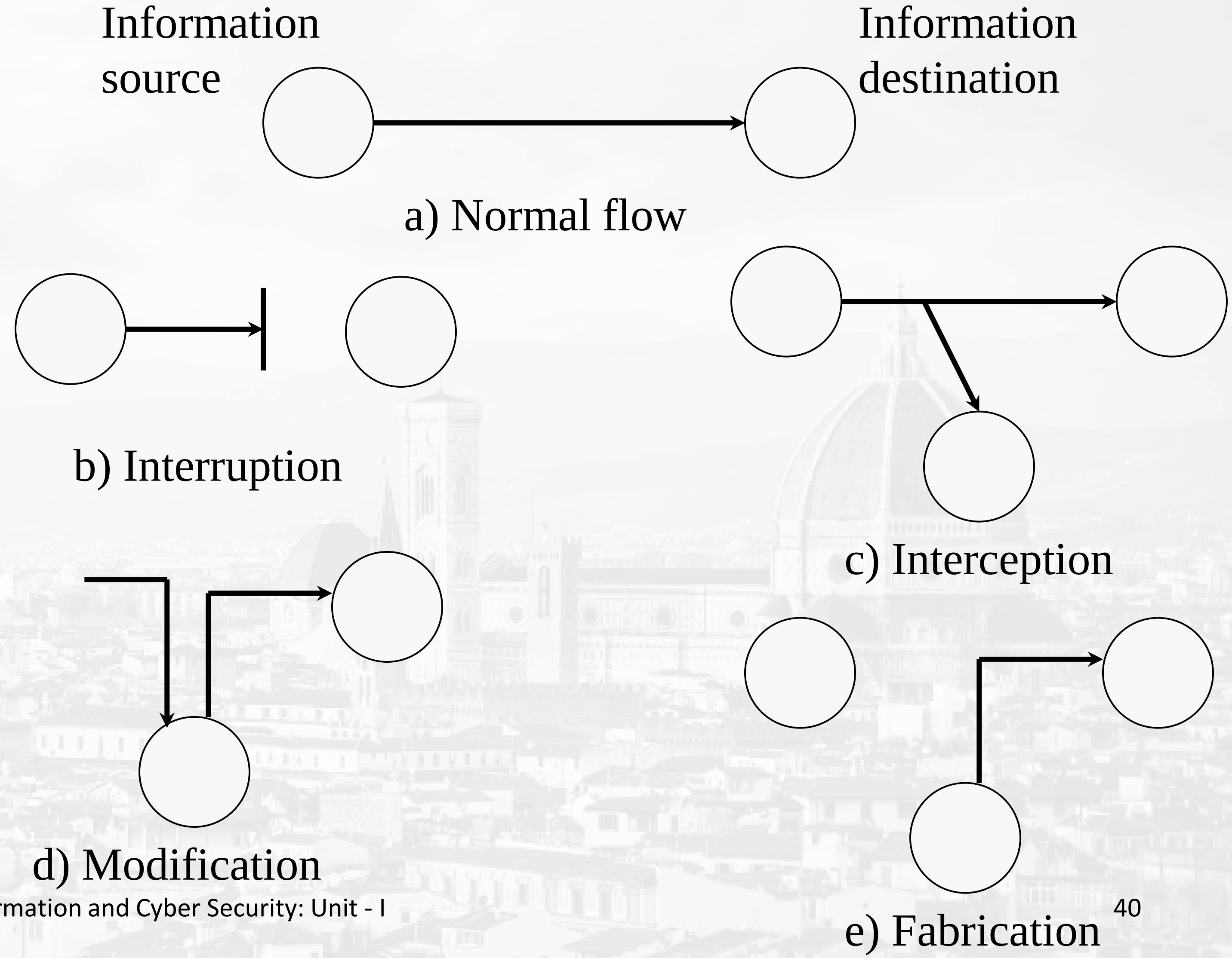
- consider 3 aspects of information security:
  - **security attack:** Any action that compromises the security of information owned by an organization.
  - **security mechanism:** A process that is designed to detect, prevent, or recover from a security attack.
  - **security service:** A processing or communication service that enhances the security of the data processing systems and the information transfers of an organization.

- note terms
  - ***threat***: a potential for violation of security
  - ***attack***: an assault on system security, a deliberate attempt to evade security services

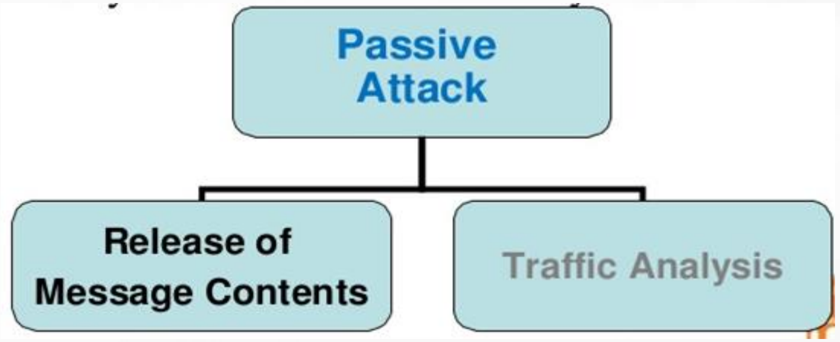
Security Attacks - Security threats

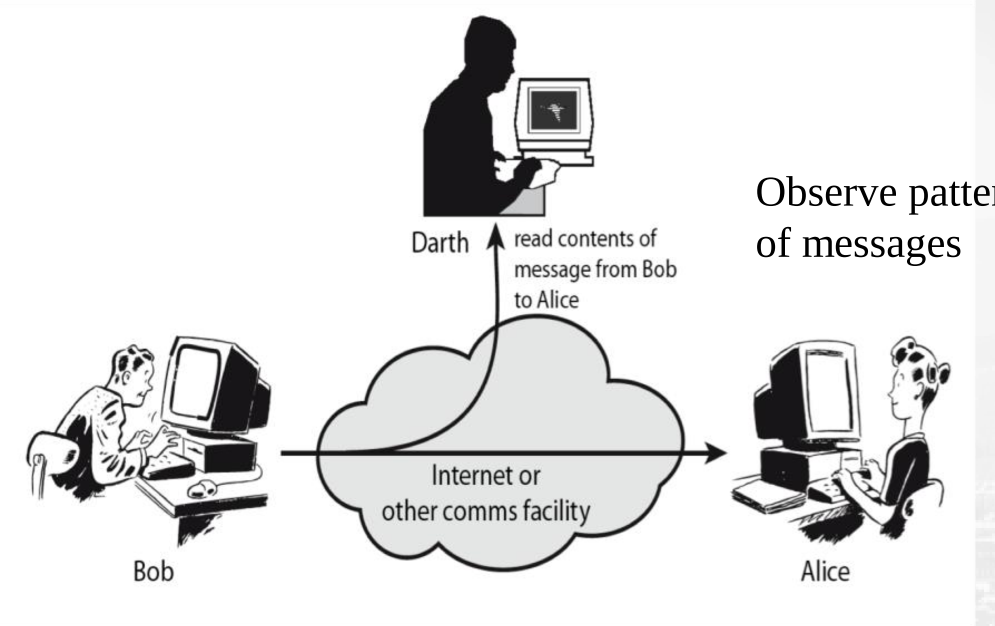


- Interruption – attack on availability
- Interception – attack on confidentiality
- Modification – attack on integrity
- Fabrication – attack on authenticity



❖ Passive Attack: make use of information from the system but does not affect system resource

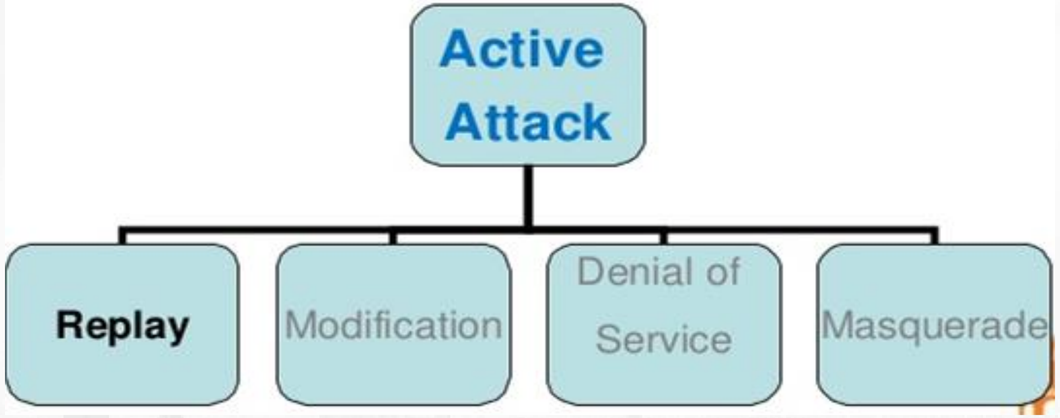


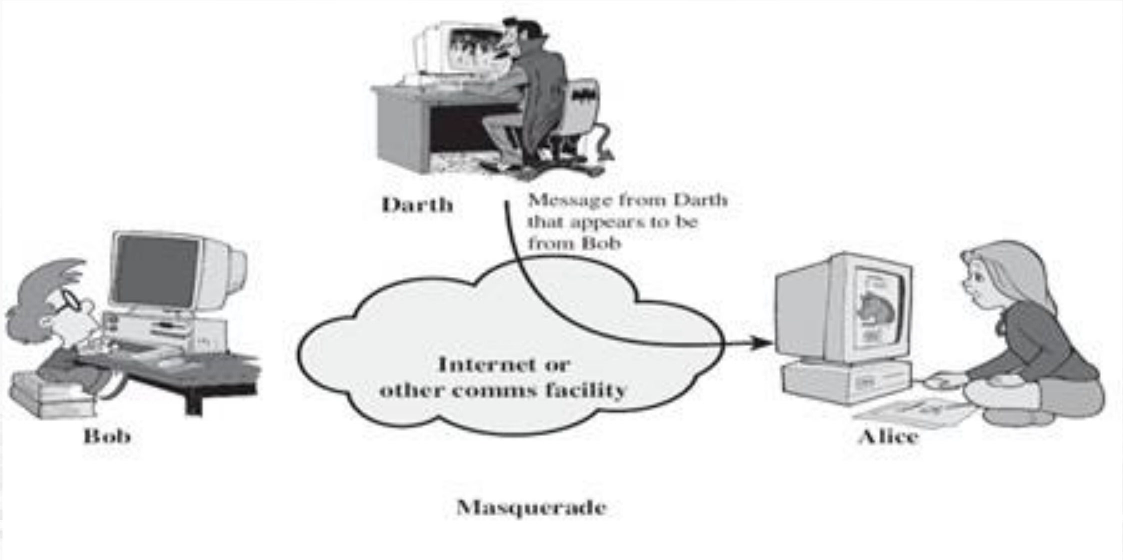


Observe pattern of messages

Note: in dealing with passive attacks is on prevention rather than detection. i.e. encryption

❖ Active Attack: modification of the data stream or the creation of a false stream





Attackers

Adversary	Goal
Student	To have fun snooping on people's e-mail
Cracker	To test out someone's security system; steal data
Sales rep	To claim to represent all of Europe, not just Andorra
Businessman	To discover a competitor's strategic marketing plan
Ex-employee	To get revenge for being fired
Accountant	To embezzle money from a company
Stockbroker	To deny a promise made to a customer by e-mail
Con man	To steal credit card numbers for sale
Spy	To learn an enemy's military or industrial secrets
Terrorist	To steal germ warfare secrets

Security Services/Goals

- Confidentiality (privacy)
- Authentication (who created or sent the data)
- Integrity (has not been altered)
- Non-repudiation (the order is final)
- Access control (prevent misuse of resources)
- Availability (permanence, non-erasure)

Security Mechanism

- feature designed to detect, prevent, or recover from a security attack
- no single mechanism that will support all services required
- however one particular element underlies many of the security mechanisms in use:
  - **cryptographic techniques**
- **specific security mechanisms:**
  - encipherment, digital signatures, access controls, data integrity, authentication exchange, traffic padding, routing control, notarization
- **pervasive security mechanisms:**
  - trusted functionality, security labels, event detection, security audit trails, security recovery

Quiz : Match the Following?

a)Interruption	1)integrity
b)Interception	2)availability
c)Modification	3)authentication
d)Fabrication	4)confidentiality

Methods of Defense

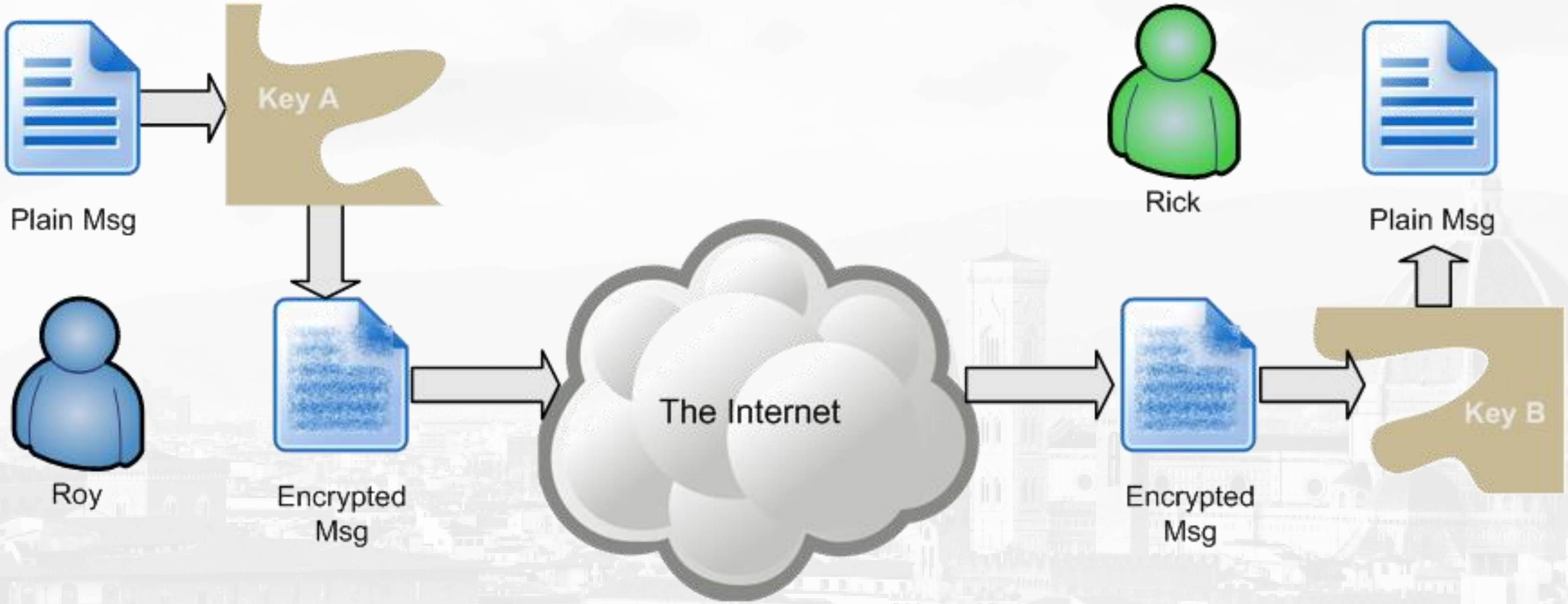
- Cryptography Controls
- Software Controls
- Hardware Controls
- Management & Audit Controls
- Physical Controls
- Law Enforcement Controls
- Education Controls

Various Security Domains

- Number Theory: Cryptography – Infrastructure – Physical - Security Management - Audit control
- Cyber Enforcement: IT Laws Courts
- Program / Database / Operating System /Net Work Security
- Image, Audio, Video Enabled Security
- Biometric Security , Intelligent Security, Social Engineering
- Market Trends – Enterprise, Cloud, Wireless Security, Intelligence

The Operational Model of Network Security

- Prevention is better than cure



Problem

Problem 1: Consider an automated teller machine (ATM) in which users provide a personal identification number (PIN) and a card for account access. Give examples of confidentiality, integrity, and availability requirements associated with the system and, in each case, indicate the degree of importance of the requirement.

Solution: The system must keep personal identification number (PIN) confidential, both in the host system and during transmission for a transaction. In addition, for security the personal identification number must encrypted.

It must protect the integrity of account records and of individual transactions.

Availability of the host system is important to the economic well being of the bank, but not to its fiduciary responsibility. The availability of individual teller machines is of less concern.

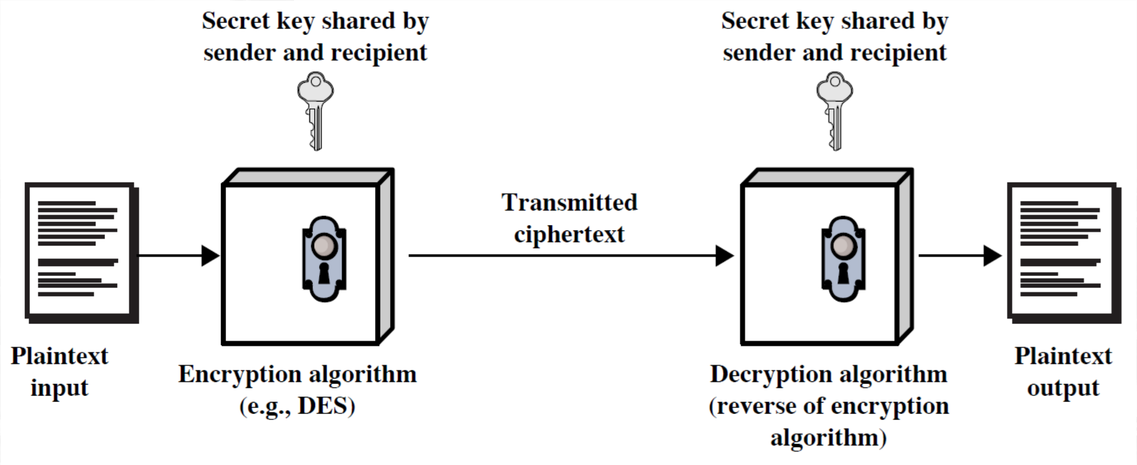
Classical Cryptography

Basic Terminology

- Plaintext- the original message
- Ciphertext - the coded message
- Cipher - algorithm for transforming plaintext to ciphertext
- Key - info used in cipher known only to sender/receiver
- Encipher (encrypt) - converting plaintext to ciphertext
- Decipher (decrypt) - recovering ciphertext from plaintext
- Cryptography - study of encryption principles/methods
- Cryptanalysis (codebreaking) - the study of principles/ methods of deciphering ciphertext without knowing key
- cryptology - the field of both cryptography and cryptanalysis

Encryption Methods

- **Symmetric encryption-** DES, Triple DES, AES
- **Asymmetric encryption-** RSA, ECC
  - The security of encryption algorithm depends upon the key



- Symmetric encryption or conventional / private-key/ single-key
- sender and recipient share a common key
- all classical encryption algorithms are private-key
- was only type prior to invention of public-key in 1970's

Cryptography

-- Parameters used by cryptographic systems are:

- The type of **operations** used for transforming plaintext to ciphertext
  - e.g. substitution and transposition
- The number of **keys** used e.g. symmetric, asymmetric
- The way in which the plaintext is **processed** e.g. block cipher, stream cipher

❑ Substitution Ciphers:

Classical Ciphers:

- Plaintext is viewed as a sequence of elements (e.g., bits or characters)
- Substitution cipher : replacing each element of the plaintext with another element.
- Transposition (or permutation) cipher : rearranging the order of the elements of the plaintext.
- Product cipher : using multiple stages of substitutions and transpositions

Caesar Cipher

- Earliest known substitution cipher. Invented by Julius Caesar
- Each letter is replaced by the letter **three** positions further down the alphabet.

Plain: a b c d e f g h i j k l m n o p q r s t u v w x y z

Cipher: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

- Example: **mit pune** □ **PLW SXQH**

E.g. break ciphertext using shift 2 "**GCUA VQ DTGCM**"
Answer: easy to break

  - Mathematically, map letters to numbers:

a	b	c	d	e	f	g	h	i	j	k	l	m
0	1	2	3	4	5	6	7	8	9	10	11	12
n	o	p	q	r	s	t	u	v	w	x	y	z
13	14	15	16	17	18	19	20	21	22	23	24	25

- Then the general Caesar cipher is:

$$c = E_K(p) = (p + k) \bmod 26$$
$$p = D_K(c) = (c - k) \bmod 26$$

- Brute force attack is easily possible

Monoalphabetic Cipher

- Shuffle the letters and map each plaintext letter to a different random ciphertext letter:

  Plain letters: abcdefghijklmnopqrstuvwxyz

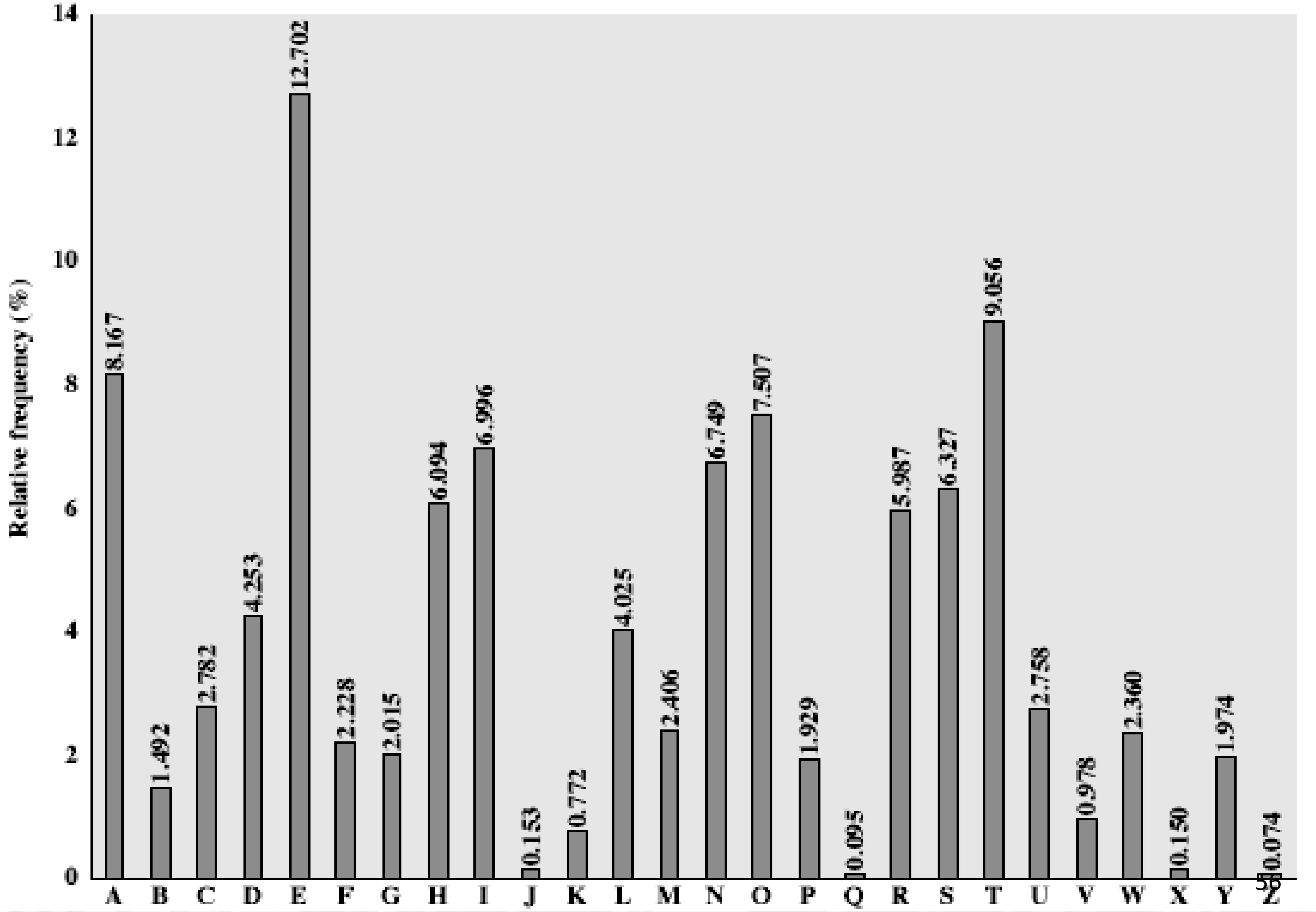
  Cipher letters: DKVQFIBJWPESCXHTMYAUOLRGZN

  Plaintext: ifwewishtoreplaceletters

  Ciphertext: WIRFRWAJUHYFTSDVFSFUUFYA

- Now we have a total of **26! = 4 x 10^{26} keys**.
- With so many keys, it is secure against brute-force attacks.
- But not secure against some cryptanalytic attacks.
- Problem is language characteristics.

English Letter Frequencies



- given ciphertext:
  UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
  VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
  EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ

- count relative letter frequencies (see text)
- guess **P & Z** are **e and t**
- guess **ZW** is **th** and hence **ZWP** is **the**

```
UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
 t a          e  e te  a that e e a          a
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
   e t     ta t ha e ee   a e  th     t   a
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
 e   e e tat   e    the    t
```

- proceeding with trial and error finally get:
  it was disclosed yesterday that several informal but
  direct contacts have been made with political
  representatives of the Viet cong in Moscow

One-Time Pad

- The number of possible keys is equal to the number of possible plaintexts
- The key is selected at random from the choice of all possible keys
- Any key should only be used once
- It is unbreakable since ciphertext bears no statistical relationship to the plaintext

Plaintext:	M	E	E	T	M	E	O	U	T	S	I	D	E
Numerical Plaintext:	12	4	4	19	12	4	14	20	19	18	8	3	4
OTP:	B	D	U	F	G	H	W	E	I	U	F	G	W
Numerical OTP:	1	3	20	5	6	7	22	4	8	20	5	6	22
Numerical Ciphertext:	13	7	24	24	18	11	10	24	1	12	13	9	
Ciphertext:	N	H	Y	Y	S	L	K	Y	B	M	N	J	

4 + 22 = 0 modular 26

$$c_i = p_i \text{ XOR } k_i$$

Two fundamental difficulties:

- Problem of making large quantities of random keys
- Problem of key distribution and protection.

❖ Because of these difficulties, the one-time pad is of limited utility and is useful primarily for low-bandwidth channels requiring very high security.

Transposition Cipher

- The order of alphabets in the plaintext is rearranged to form a cipher text.
- The Rail Fence cipher is a form of transposition cipher
- Plaintext is written down as a sequence of diagonals and then read off as a sequence of rows.
- The example message is: "meet me after the toga party" with a rail fence of depth 2.

```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

Ciphertext:
MEMATRHTGPRYETEFETEOAAT

- Write the message in a rectangle, row by row, and read the message off, column by column, but permute the order of the columns
- The order of the columns then becomes the key to the algorithm

```
Key:        4 3 1 2 5 6 7
Plaintext:  a t t a c k p
            o s t p o n e
            d u n t i l t
            w o a m x y z
```

The key is 4312567

Ciphertext: TTNAAPTMTSUOAODWCOIXKNLYPETZ60

Transposition Cipher(2)

- Plaintext written in a row under the key and then arrange the column as per alphabetical order.

❖ **Single Columnar Transposition**

Preparing the Key:

- Numbered each letter of the key as per their appearance in the alphabet

h	e	a	v	e	n
4	2	1	6	3	5

Preparing the Plaintext: **we are the best**

h	e	a	v	e	n
4	2	1	6	3	5
W	E	A	R	E	T
H	E	B	E	S	T

Transposition Cipher(3)

Encryption:

a	e	e	h	n	v
1	2	3	4	5	6
A	E	E	W	T	R
B	E	S	H	T	E

Decryption: ABEEESWHTTRE

h	e	a	v	e	n
4	2	1	6	3	5
W	**E**	**A**	R	**E**	T
H	**E**	**B**	E	**S**	T

Problem: Using Transposition cipher encrypt message "WE ARE THE BEST" use key 'HEAVEN'

Transposition Ciphers(4)

❖ Double Columnar Transposition

h	e	a	v	e	n
4	2	1	6	3	5
W	E	A	R	E	T
H	E	B	E	S	T

ABEEESWHTTRE

a	n	o	t	h	e	r
1	4	5	7	3	2	6
A	B	E	E	E	S	W
H	T	T	R	E		

Modes of Operation

- A block cipher processes the data blocks of fixed size.
- Usually, the size of a message is larger than the block size.
- Hence, the long message is divided into a series of sequential message blocks, and the cipher operates on these blocks one at a time.
- The size of plaintext block and ciphertext block is same.
- When no. of bits in the plaintext/message are not multiple of the block size ---- Modes of operation

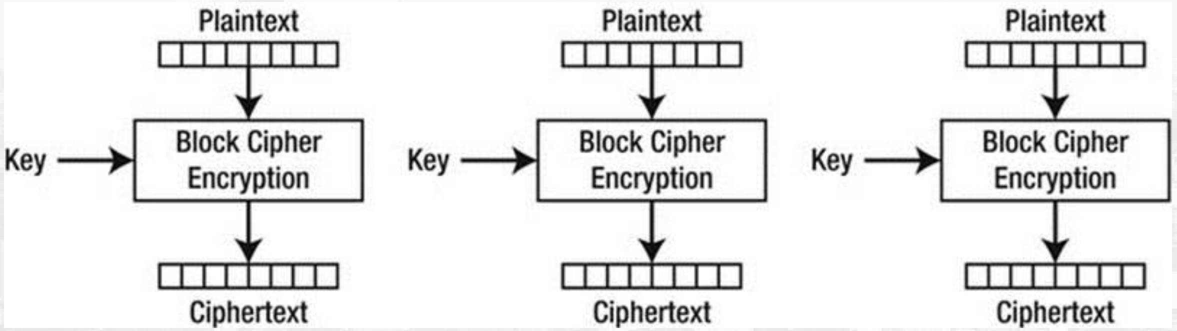
Block Cipher Modes of Operation

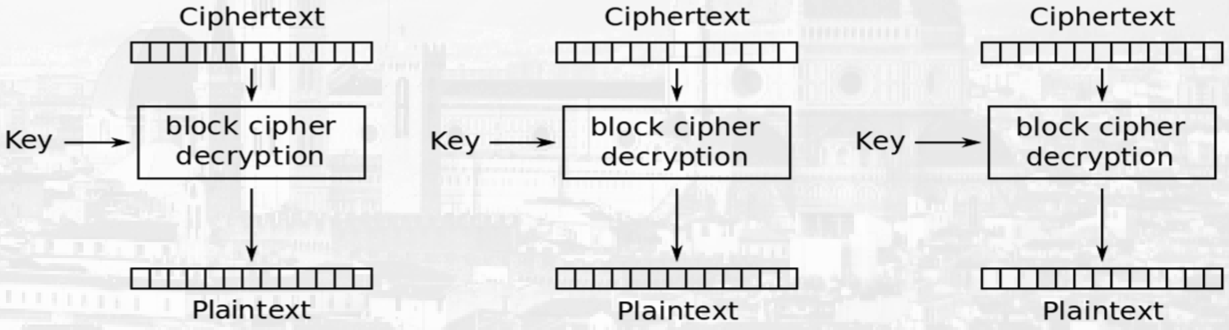
- **Electronic Code book (ECB) mode**
- **Cipher Block Chaining (CBC) mode**
- **Feedback (CFB) modes**
- **Counter (CTR) mode**

Electronic Code book (ECB) mode

- The message is divided into blocks, and each block is encrypted separately.
- If two plaintext blocks are identical then the ciphertext block are also same . Therefore, **a known plaintext attack is possible**.
- **uses**: secure transmission of single values

$$C_i = E_K(P_i)$$

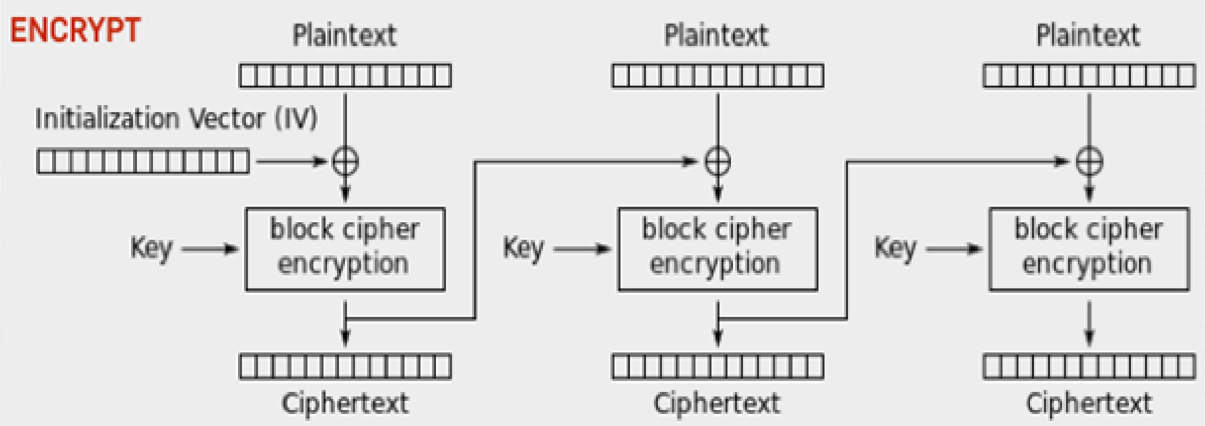


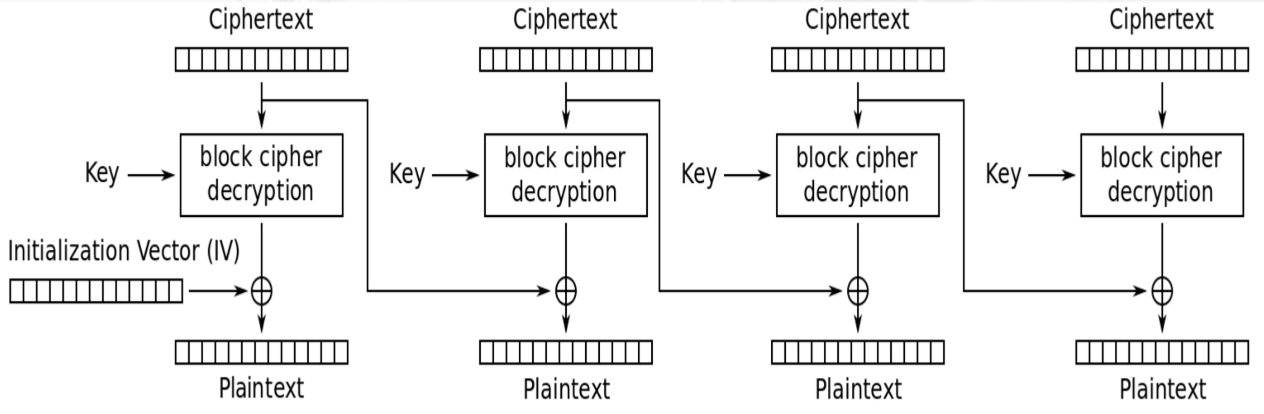


Cipher Block Chaining (CBC) mode

- The message is divided into blocks, and each block is encrypted separately.
- An initialisation is random number is used to increase security.
- It can be used to generate the hash value.
- **uses:** bulk data encryption, authentication

$$C_i = E_K(P_i \text{ XOR } C_{i-1})$$
$$C_{-1} = IV$$





Advantages:

- For identical block of plaintext, different ciphertext blocks are generated. So secure than ECB mode.
- Hash value, i.e. last ciphertext block, helps to identify if the message is original or modified.

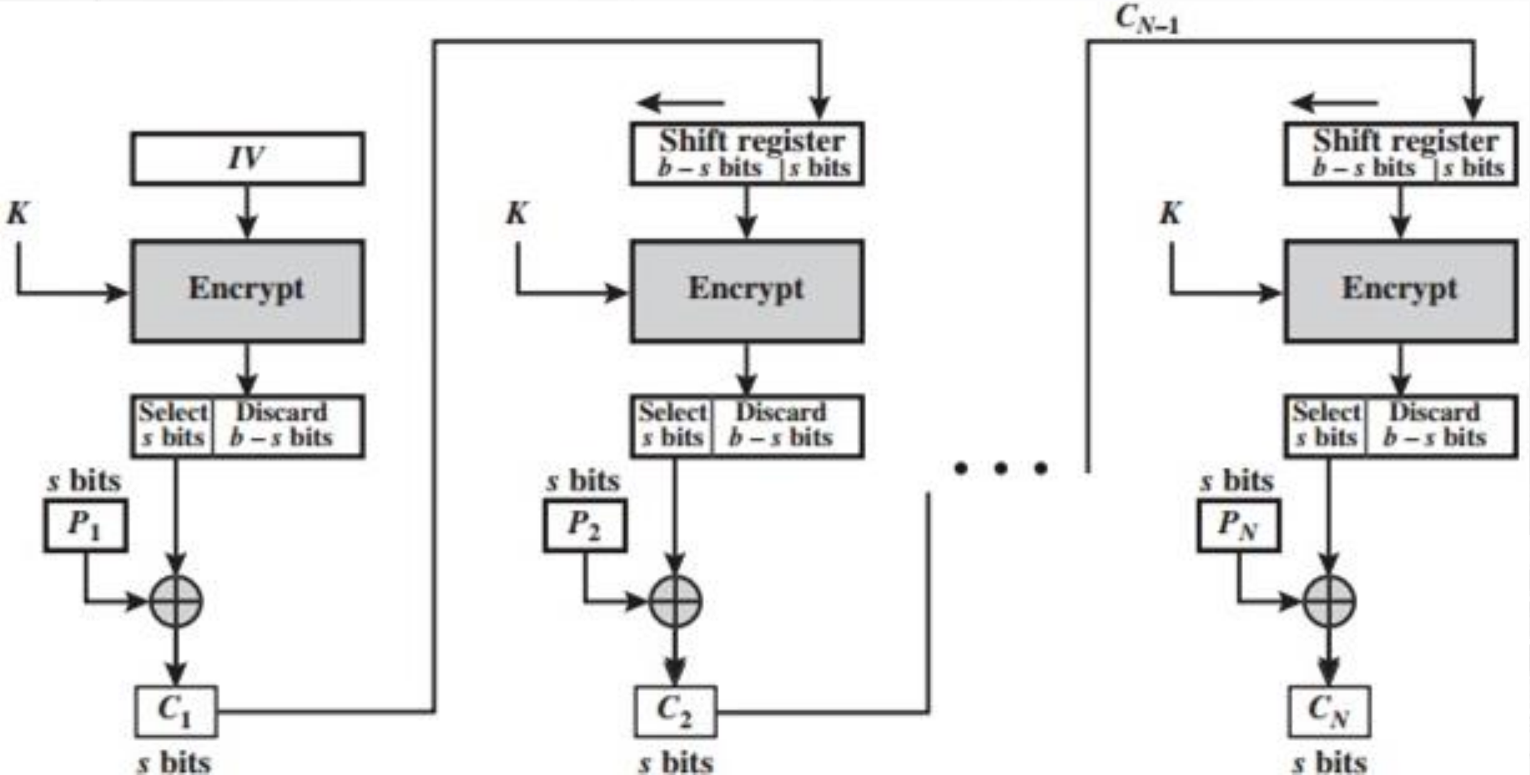
Disadvantages:

- Parallel operation cannot be performed.
- Lost/missing of any block of ciphertext stops the decryption process of the remaining blocks.

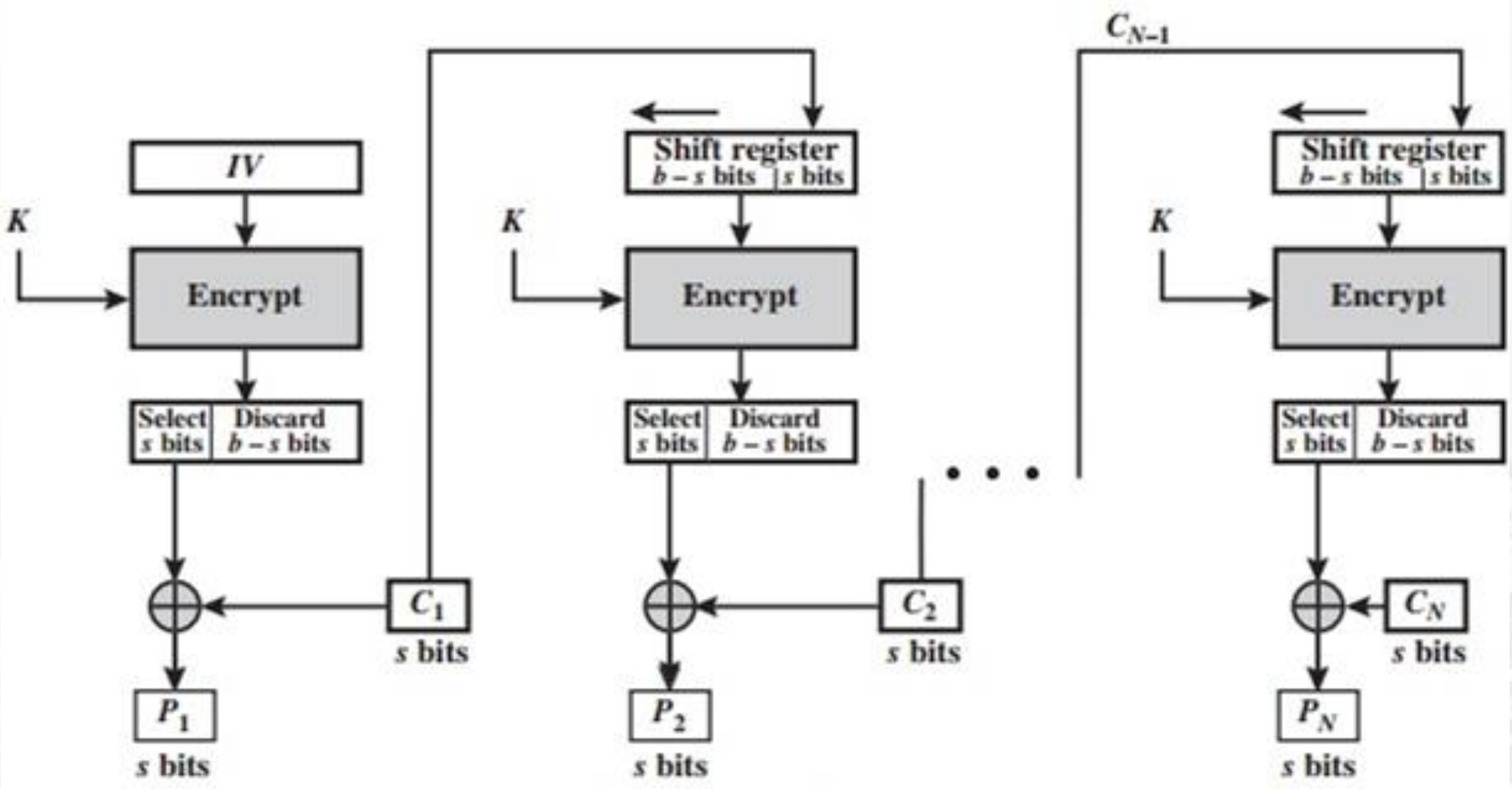
Cipher Feedback (CFB) Mode

- Can be used when the block size is smaller than the required block size.
- The block size may be a bit or bytes, so there is no need of padding.
- **uses:** stream data encryption, authentication

$$C_i = P_i \text{ XOR } E_K(C_{i-1})$$
$$C_{-1} = IV$$



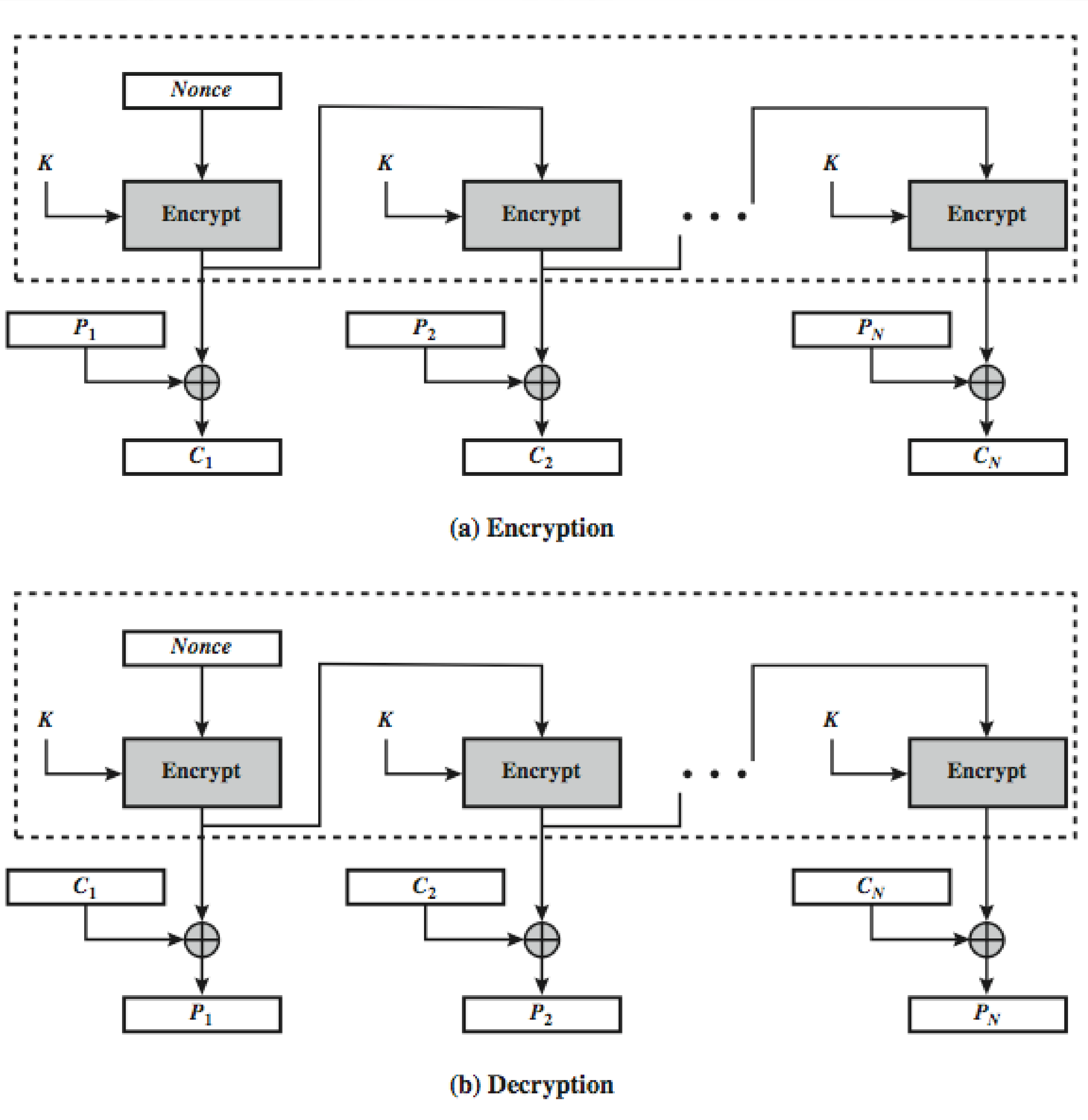
❖ CFB is suffered from bit errors. If in the incoming cipher block, any one bit error is there, then it causes the bit error at the same bit position in the plaintext block.



Output Feedback (OFB) Mode

- message is treated as a stream of bits
- Free from bit error rate.
- Information about the key is not required, which help the cryptanalyst to break the cipher easily.

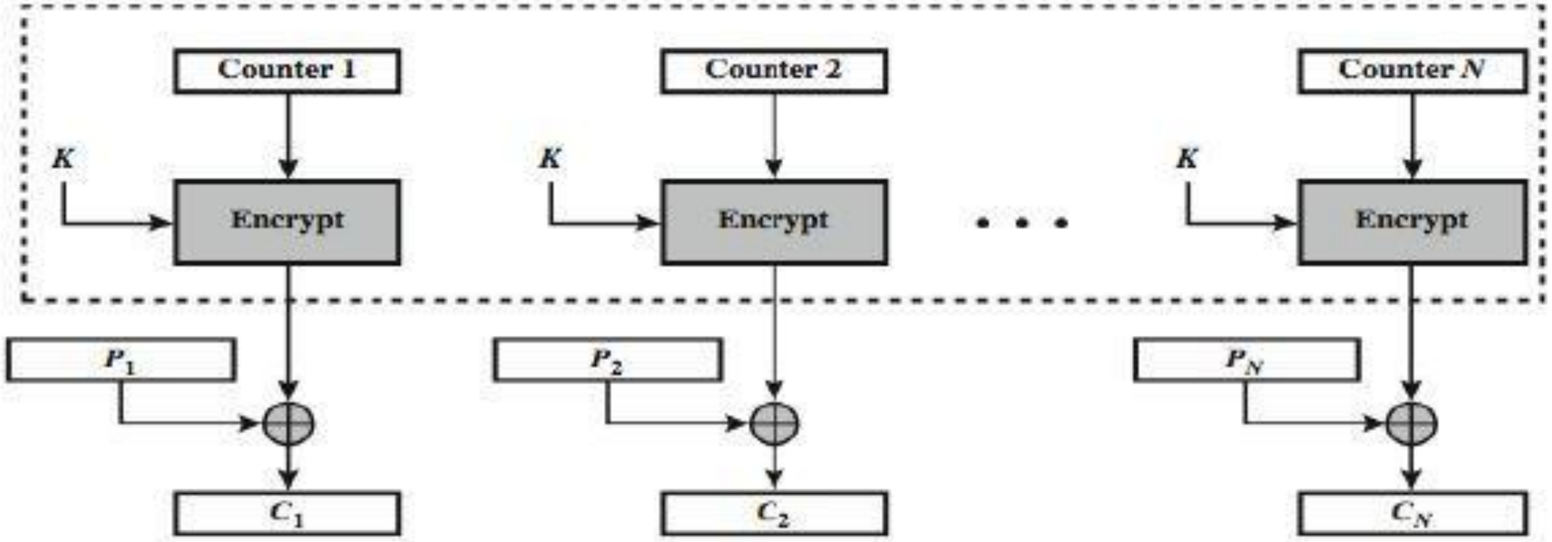
- https://www.ibm.com/docs/en/zos/2.4.0?topic=operation-output-feedback-ofb-mode


Nonce
K
Encrypt
K
Encrypt
K
Encrypt
P_1
P_2
P_N
C_1
C_2
C_N
(a) Encryption
Nonce
K
Encrypt
K
Encrypt
K
Encrypt
C_1
C_2
C_N
P_1
P_2
P_N
(b) Decryption

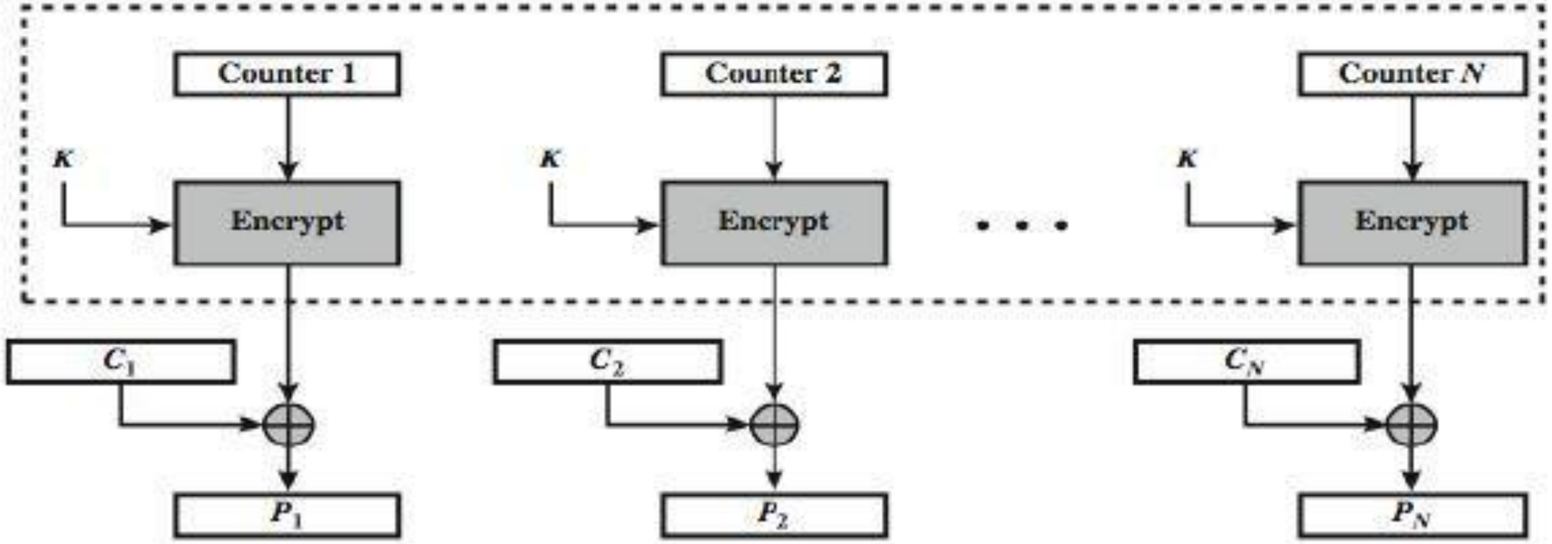
Counter (CTR) Mode

- It may be faster than of cipher block chaining mode.
- Encryption can be done in parallel.
- Padding is not required.
- Processing of plaintext blocks can be done randomly
- Integrity of the message is not maintained
- Reuse of counter value, compromise the security.
- It is ATM and IP sec

https://www.ibm.com/docs/en/zos/2.4.0?topic=operation-galoiscounter-mode-gcm



(a) Encryption



(b) Decryption

XTS-AES Mode

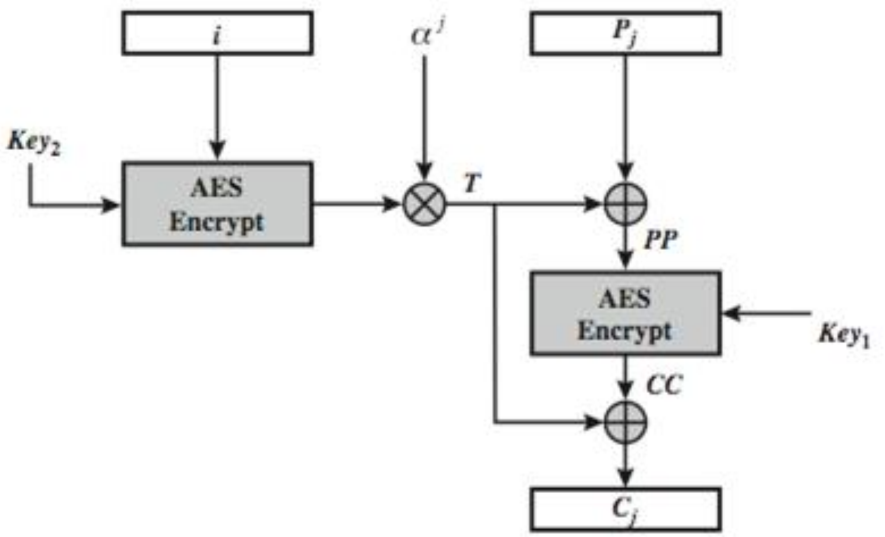
- new mode, for block oriented storage use
  - in IEEE Std 1619-2007
- concept of tweakable block cipher
- different requirements to transmitted data
- uses AES twice for each block

  $T_j = E_{K2}(i) \text{ XOR } \alpha^j$

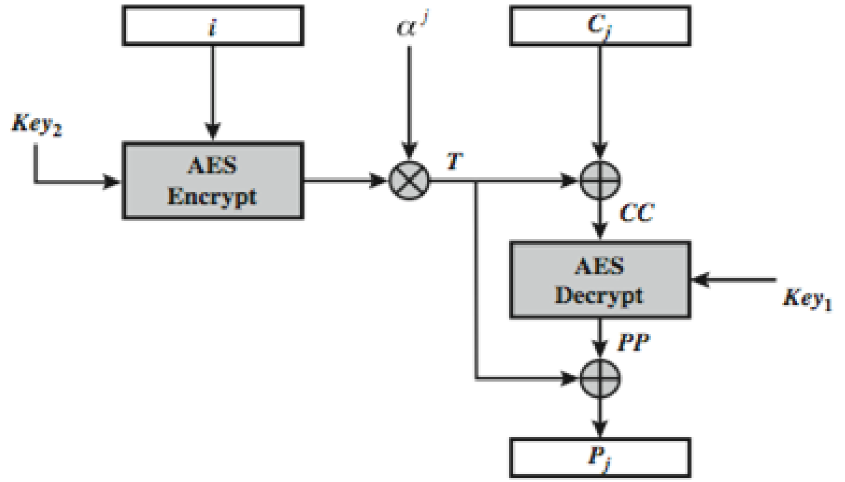
  $C_j = E_{K1}(P_j \text{ XOR } T_j) \text{ XOR } T_j$

  where i is tweak & j is sector no
- each sector may have multiple blocks

XTS-AES Mode per block

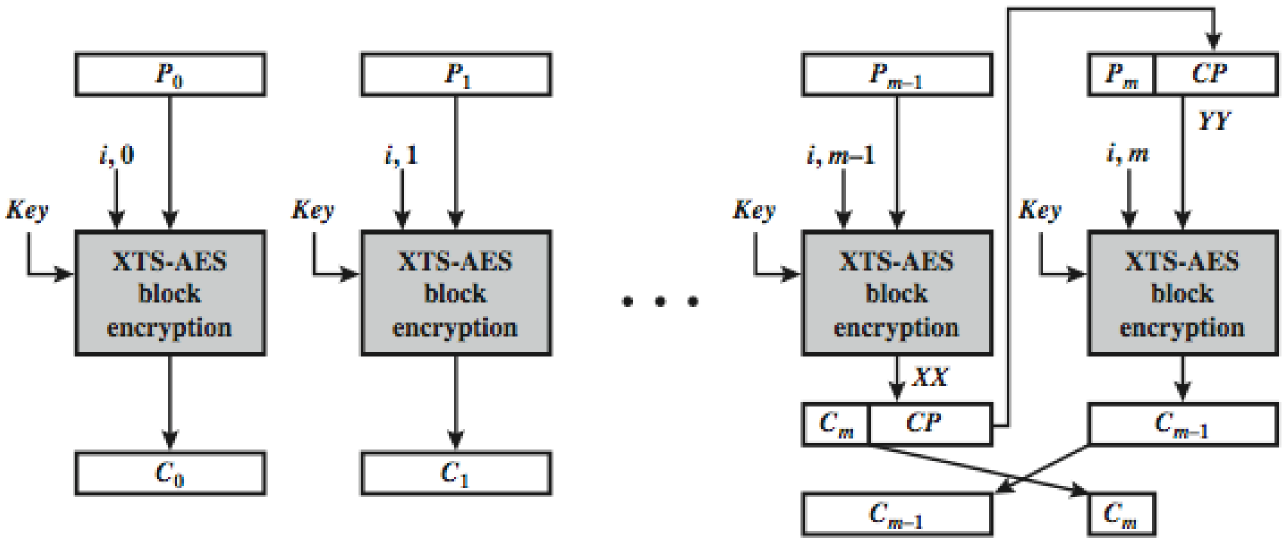


(a) Encryption

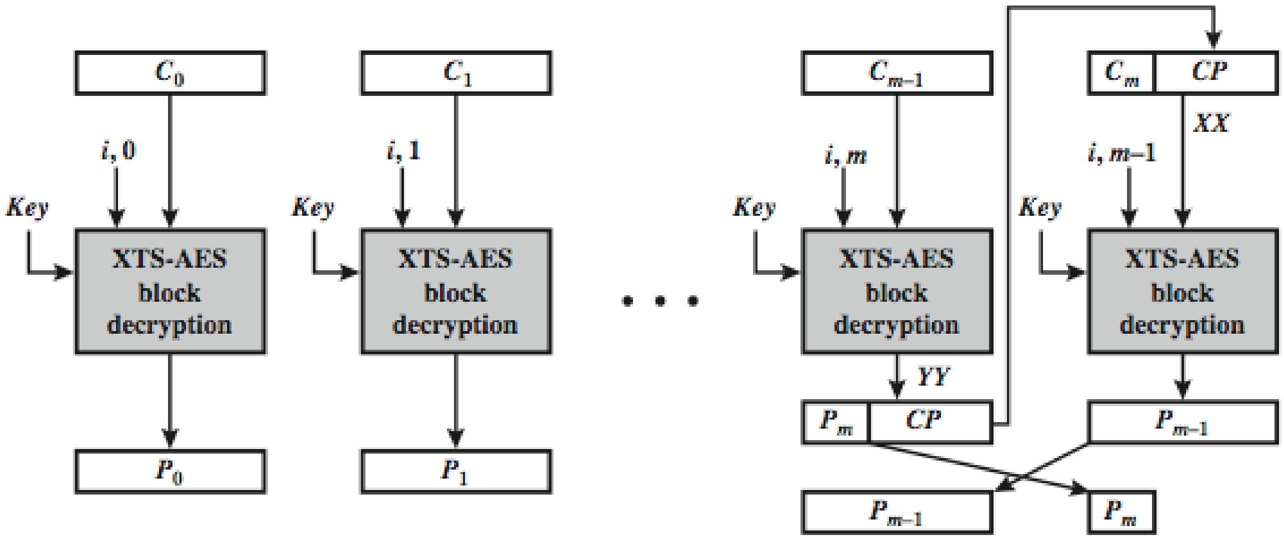


(b) Decryption

XTS-AES Mode Overview



(a) Encryption



(b) Decryption

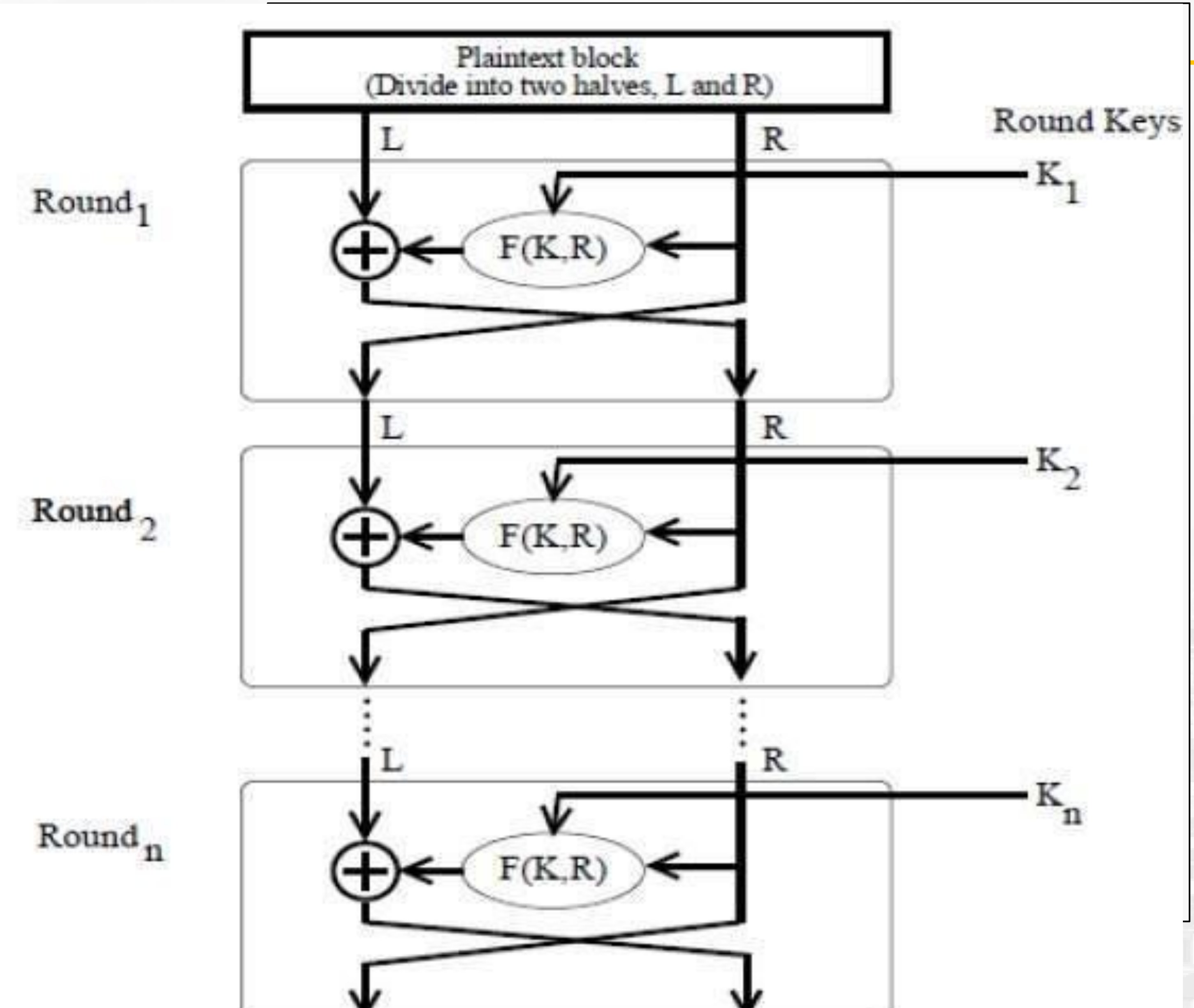
Advantages and Limitations of XTS-AES

- efficiency
  - can do parallel encryptions in h/w or s/w
  - random access to encrypted data blocks
- has both nonce & counter
- addresses security concerned related to stored data

Feistel Ciphers

- ❖ Most symmetric block ciphers are based on a **Feistel Cipher Structure**
- ❖ Needed since must be able to **decrypt** ciphertext to recover messages efficiently
- ❖ Block ciphers look like an extremely large substitution would need table of 2^{64} entries for a 64-bit block
- ❖ Instead create from smaller building blocks using idea of a product cipher
- ❖ Horst Feistel devised the feistel cipher
  - based on concept of invertible product cipher

- partitions input block into two halves
  - process through multiple rounds which:
    - perform a substitution on left data half
    - based on round function of right half & sub key
    - then have permutation swapping halves
- The plaintext is divided into two halves (L_0 and R_0). Then the two halves pass through **n** rounds of processing then combine to produce the cipher block.
- Each round i has as input L_{i-1} and R_{i-1} derived from the previous round as well as a sub-key K_i derived from the overall K


Plaintext block
(Divide into two halves, L and R)
Round Keys
L
R
Round 1
K1
F(K,R)
L
R
Round 2
K2
F(K,R)
L
R
Round n
Kn
F(K,R)

The design of Feistel cipher depends on following parameter:

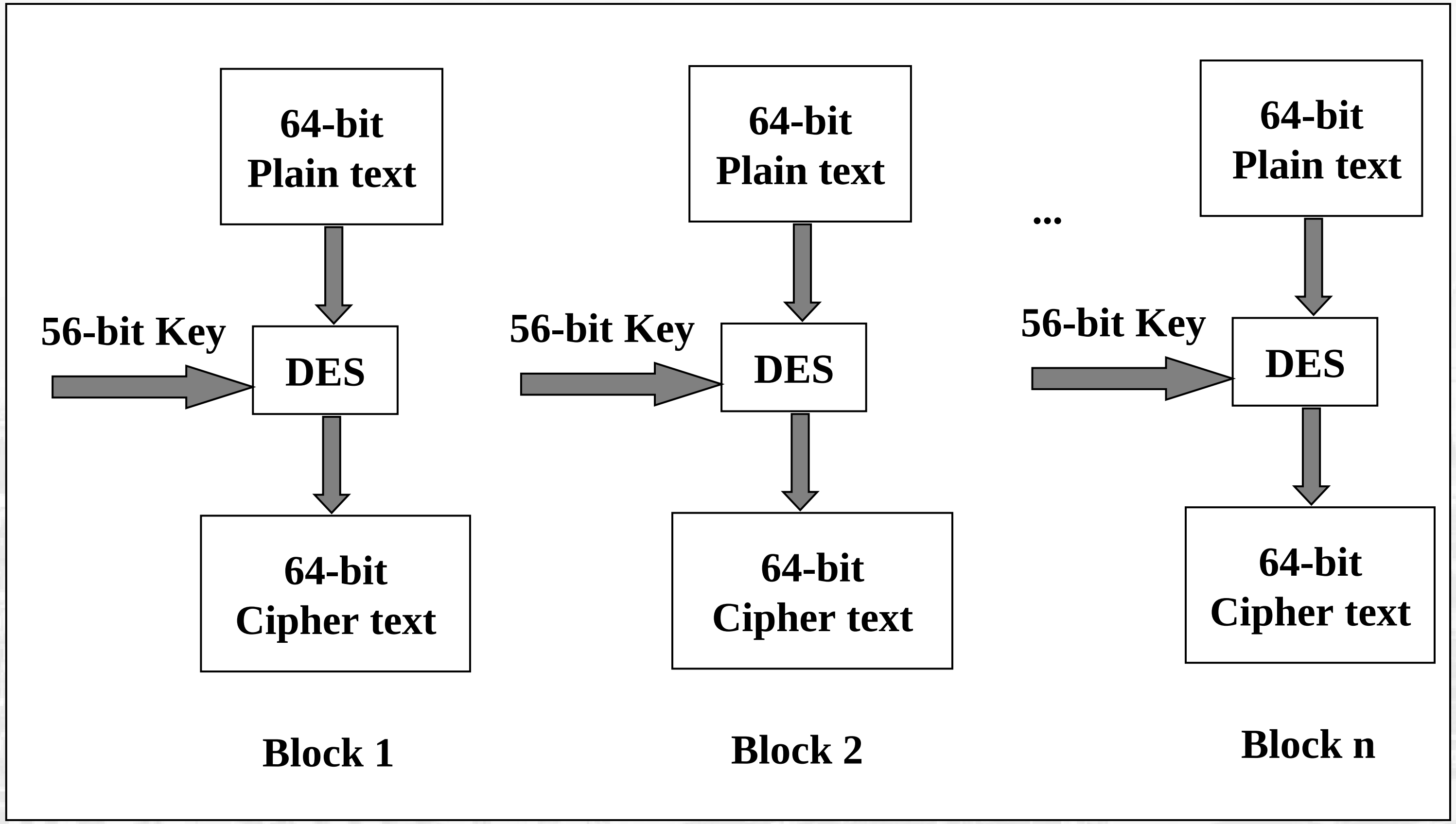
- **Block Size:** (larger block means greater security) 64 bits.
- **Key Size**:56-128 bits.
- **Number of Rounds**: a single round offers inadequate security, a typical size is 16 rounds.
- **Sub-key Generation Algorithms**: greater complexity should lead to a greater difficulty of cryptanalysis.
- **Round function:** Again, greater complexity generally means greater resistance to cryptanalysis.

Data Encryption Standard (DES)

- IBM developed Lucifer cipher
  - by team led by Feistel
  - used 64-bit data blocks with 128-bit key
- then redeveloped as a commercial cipher with input from NSA and others in 1973 NBS issued request for proposals for a national cipher standard
- IBM submitted their revised Lucifer which was eventually accepted as the DES
- encrypts 64-bit data using 56-bit key
- DES has become widely used, especially in financial applications

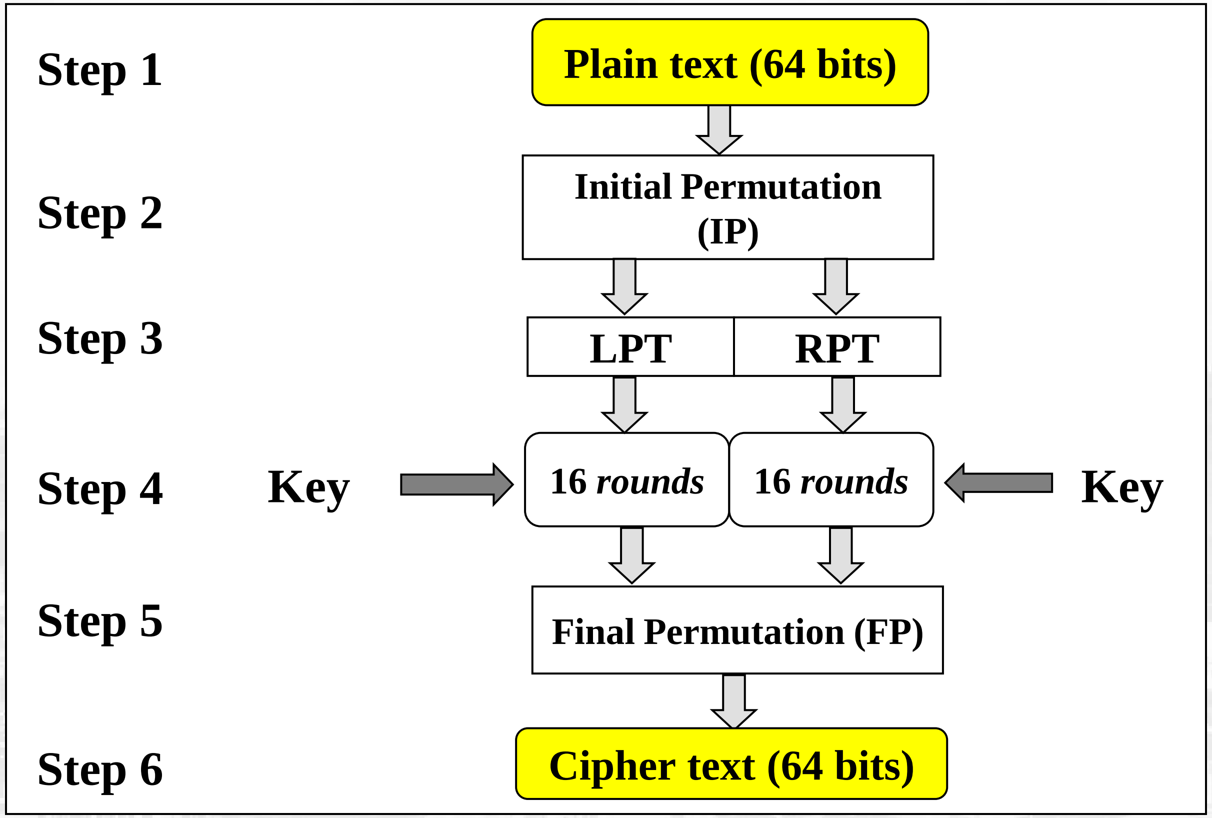
Conceptual View of DES

- Every 8th bit of the key is discarded to produce a 56-bit key
- Same algorithm and key are used for encryption and decryption



Broad Level Steps in DES

- ❖ DES is based on substitution (called as confusion) and transposition (called as diffusion)
- ❖ Each round performs the steps of substitution and transposition



❖ Initial Permutation (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

- **As we have noted, the initial permutation (IP) happens only once and it happens before the first round.**
- **It suggests how the transposition in IP should proceed, is as shown in the figure.**
- **For example, it says that the IP replaces the first bit of the original plain text block with the 58th bit of the original plain text, the second bit with the 50th bit of the original plain text block, and so on.**
- **This is nothing but jugglery of bit positions of the original plain text block. the same rule applies to all the other bit positions shown in the figure.**

❖ Details of One Round in DES

Key Transformation

↓

Expansion Permutation

↓

S-Box Substitution

↓

P-Box Permutation

↓

XOR and Swap

- As we have noted after IP is done, the resulting 64-bit permuted text block is divided into two half blocks.
- Each half-block consists of 32 bits, and each of the 16 rounds, in turn, consists of the broad level steps outlined in the figure

Step 1: Key Transformation and Compression Permutation

- We have noted initial 64-bit key is transformed into a 56-bit key by discarding every 8th bit of the initial key. Thus, for each a 56-bit key is available. From this 56-bit key, a different 48-bit Sub Key is generated during each round using a process called key transformation. For this, the 56-bit key is divided into two halves, each of 28 bits. These halves are circularly shifted left by one or two positions, depending on the round.
- For example, if the round numbers 1, 2, 9, or 16 the shift is done by only one position for other rounds, the circular shift is done by two positions. The number of key bits shifted per round is shown in the figure.

Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bits shifts	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

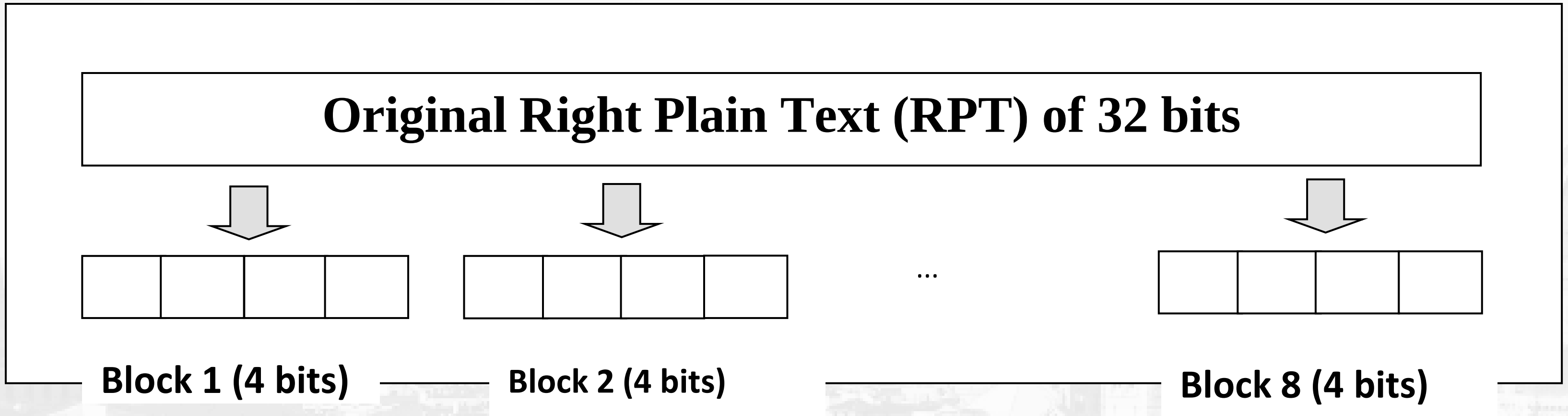
Compression Permutation

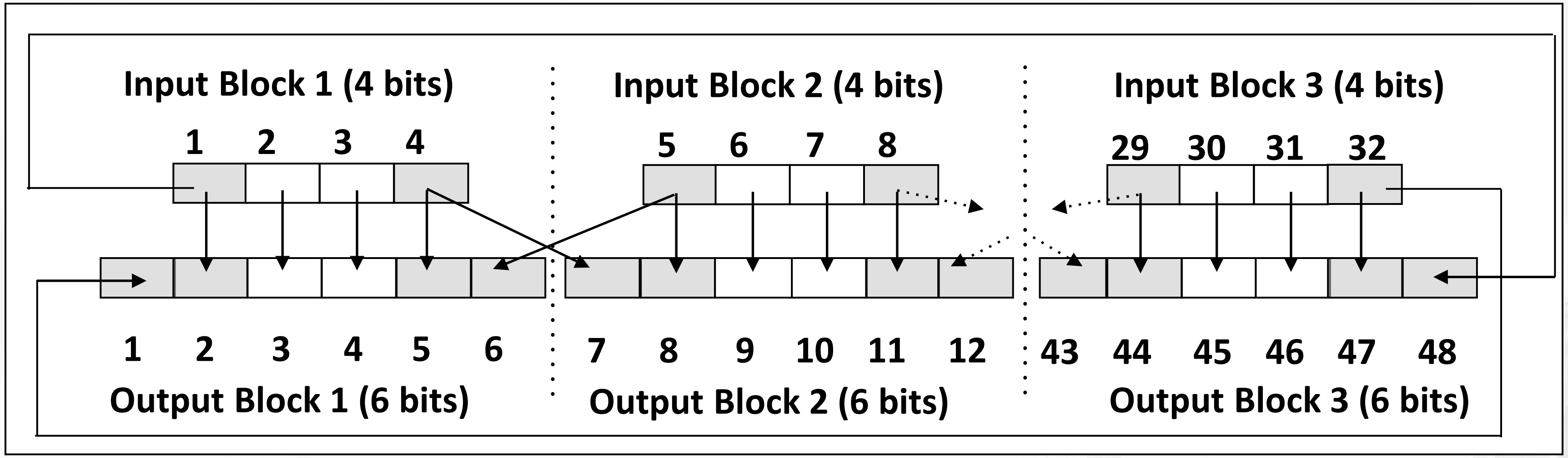
- After an appropriate shift, 48 of the 56 bits are selected. for selecting 48 of the 56 bits the table is shown in the figure given below. For instance, after the shift, bit number 14 moves to the first position, bit number 17 moves to the second position, and so on.
- If we observe the table carefully, we will realize that it contains only 48-bit positions. Bit number 18 is discarded (we will not find it in the table), like 7 others, to reduce a 56-bit key to a 48-bit key. Since the key transformation process involves permutation as well as a selection of a 48-bit subset of the original 56-bit key it is called Compression Permutation

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Step 2: Expansion Permutation

- 32-bit RPT is divided into 8 blocks (each block 4-bits)
- Each 4-bit block is expanded to 6-bit block. Two bits -- repeated first and forth bits



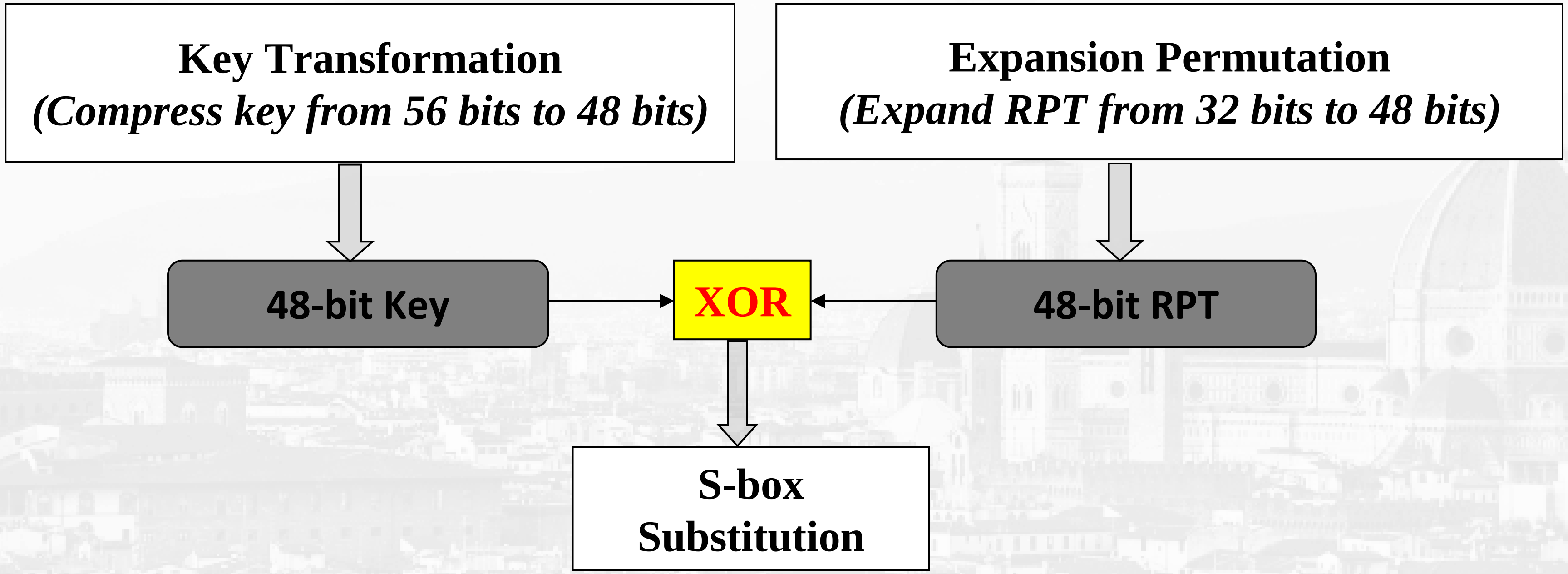


RPT Expansion Permutation Table

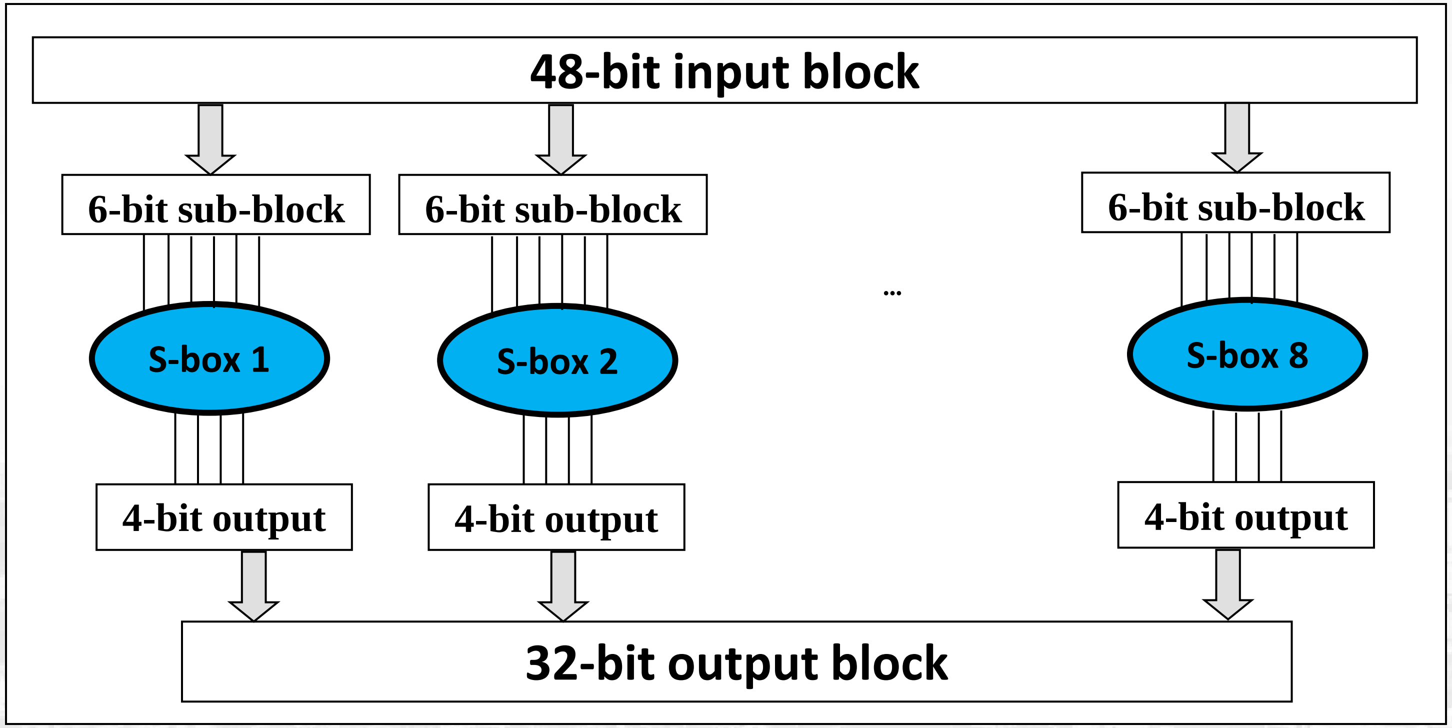
32	1	2	3	4	5	4	5	6	7	8	9
8	9	10	11	12	13	12	13	14	15	16	17
16	17	18	19	20	21	20	21	22	23	24	25
24	25	26	27	28	29	28	29	30	31	32	1

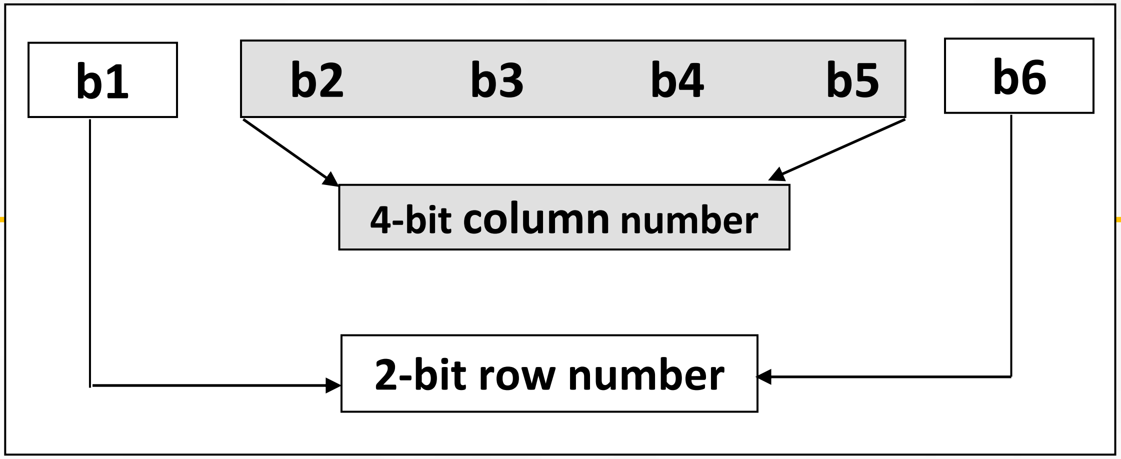
Step 3: S-box substitution

- Output: 32 bit



- Eight S-boxes that accept 6 bit inputs and produce 4 bit outputs





input 101110

primo ed ultimo bit

10

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	0110	0111	1111
00	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
01	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
10	4	1	7	8	13	6	2	11	15	12	9	7	3	10	5	0
11	15	12	10	2	4	9	1	7	5	11	3	14	10	0	6	13

Box S1

output 11 in binario = 1011

S_1	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S_2	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S_3	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

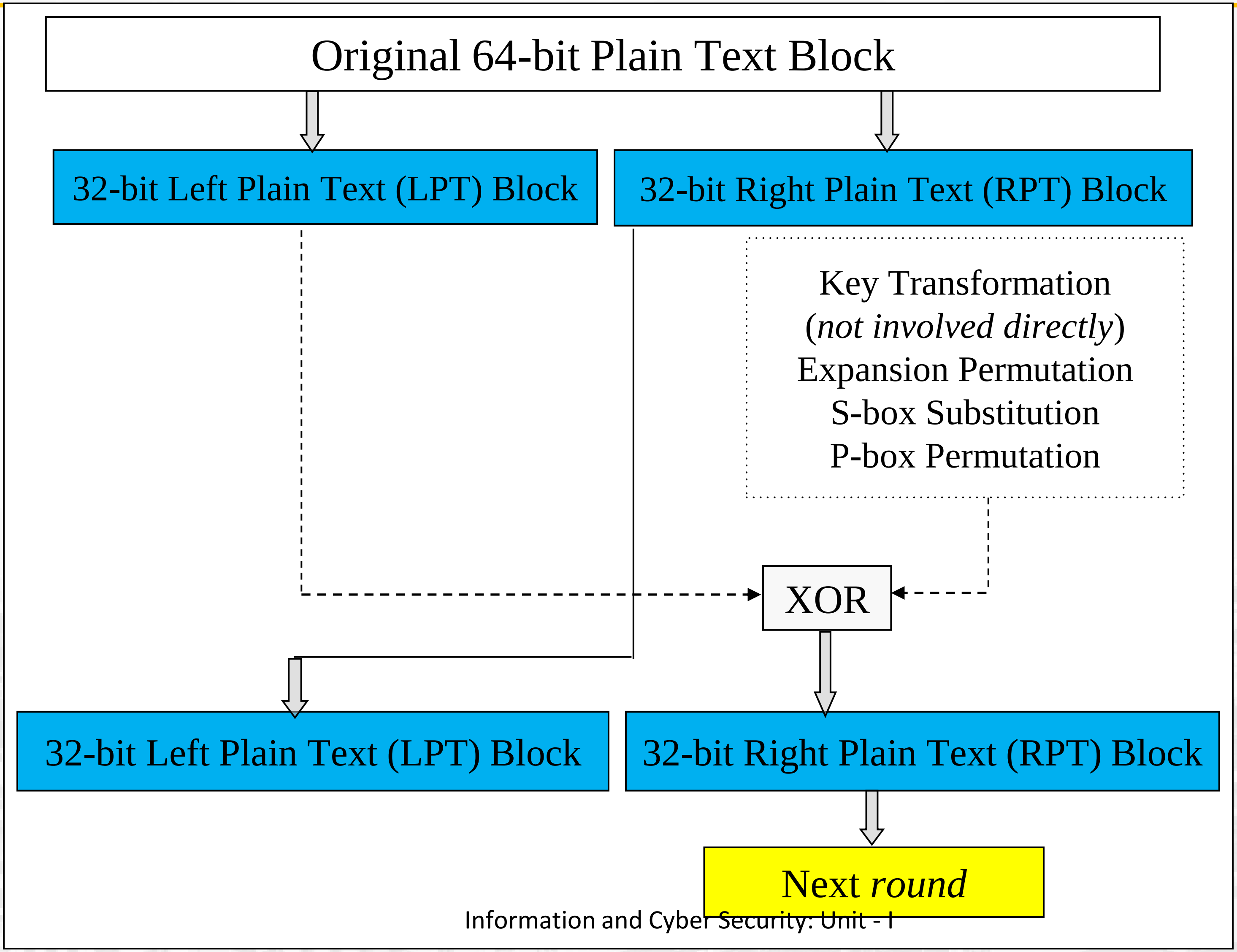
S_4	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

Step 4: P-box permutation

- The output of S-box consists of 32 bits
- It is straight permutation. No bits are used twice and no bits are ignored
- Replacement of each bit with another bit

16	7	20	21	29	12	28	17	1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

Step 5: XOR and Swap



❖ Final permutation

At the end of all 16 rounds, the final permutation is performed only once with Permutation

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

DES Decryption

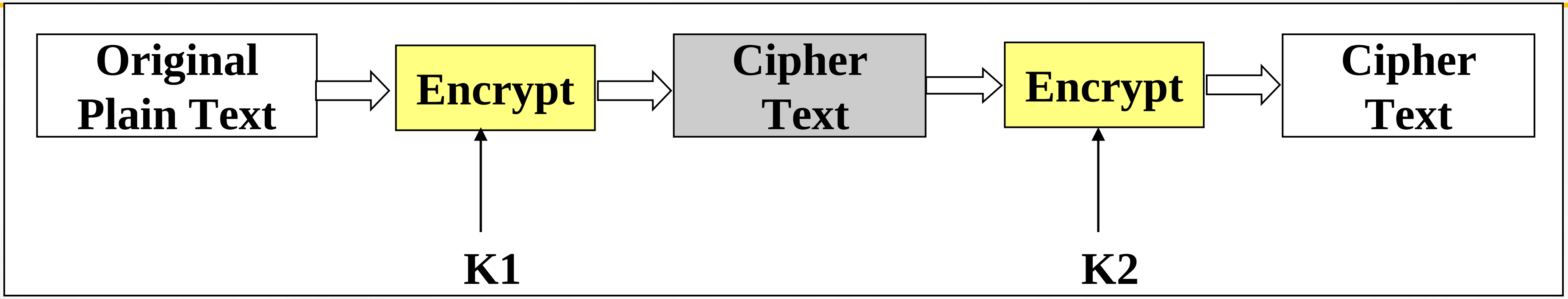
- Same algorithm and key are used for encryption and decryption
- Key reversal is used i.e. K16, K15, …… K1

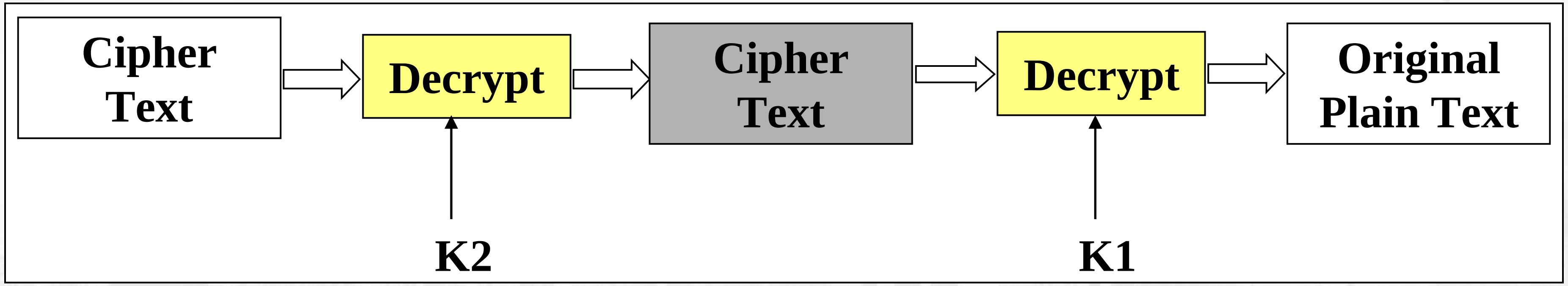
Analysis of DES

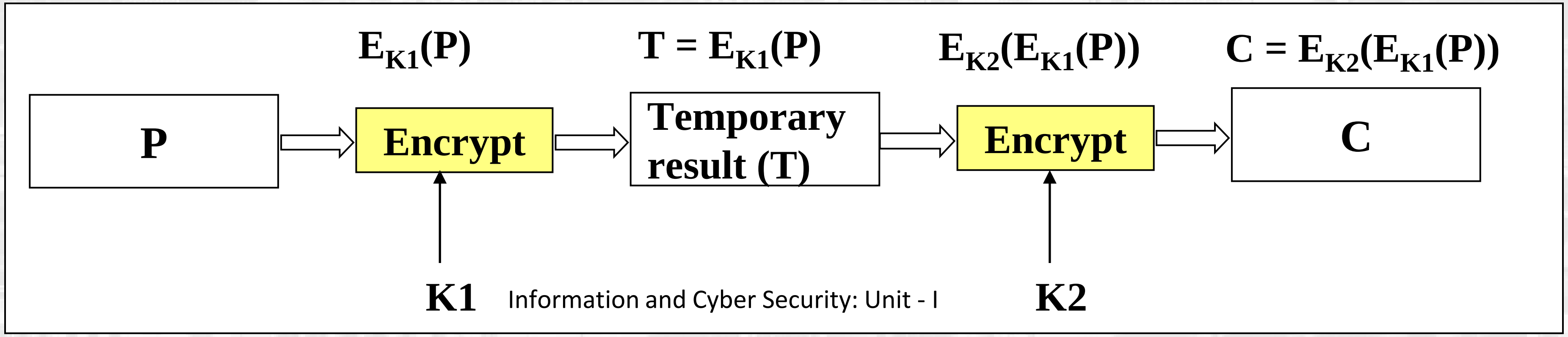
- **Use of S-boxes**: The table used for substitution in DES are kept secret by IBM. It takes 17 years come up with internal design of the S-boxes.
- **Key Length**: There are 2^{56} possible keys i.e. 7.2×10^{16} keys. Thus, it seems that a brute-force attack on DES is impractical. A single computer performing one DES encryption per microsecond would require more than 1000 years to break DES.

Variations of DES: Double DES

- Meet-in-the-middle attack

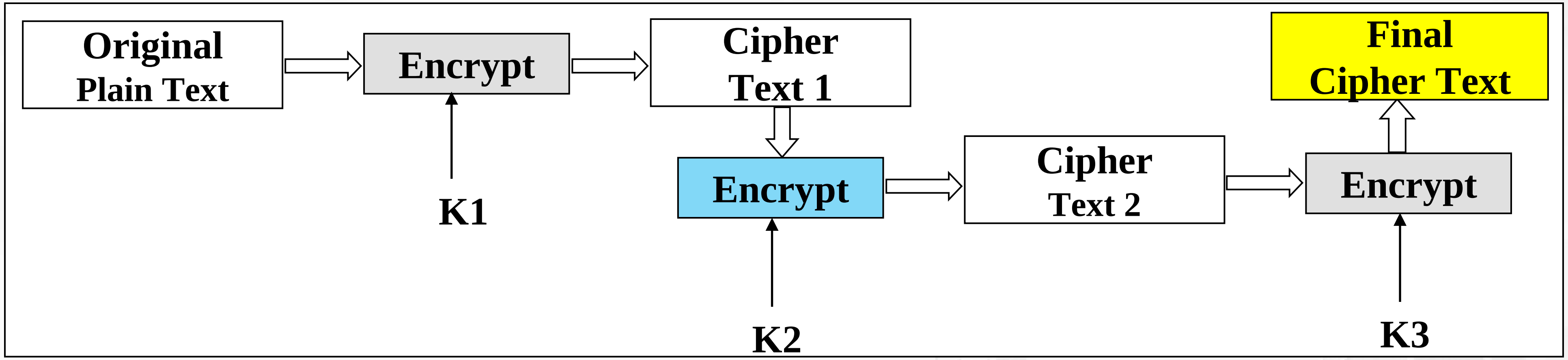


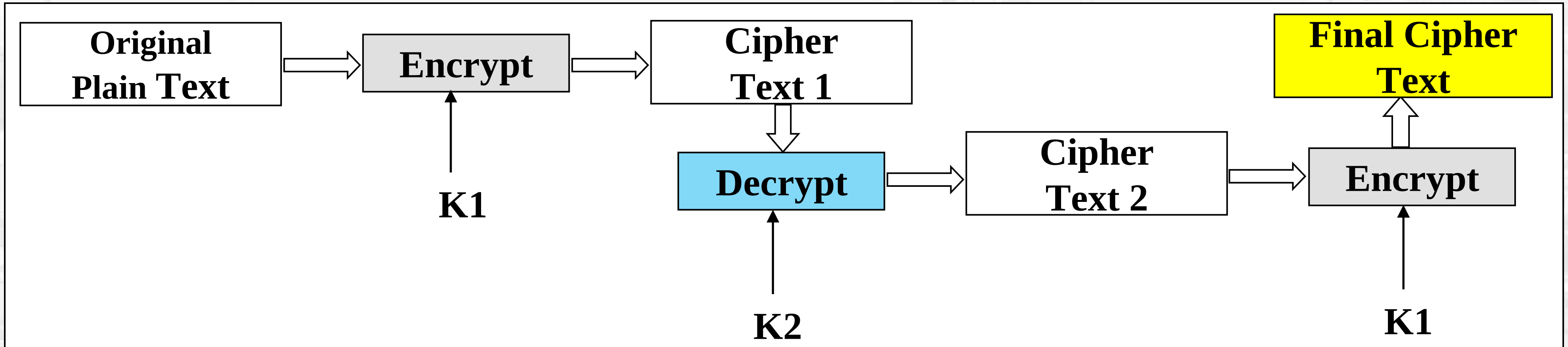




Triple DES

- Secure but more time for encryption





DES Weaknesses

Weaknesses in Cipher Design

S-boxes: At least three weaknesses are mentioned in the literature for S-boxes.

1. In S-box 4, the last three output bits can be derived in the same way as the first output bit by complementing some of the input bits.
2. Two specifically chosen inputs to an S-box array can create the same output.
3. It is possible to obtain the same output in a single round by changing bits in only three neighboring S-boxes.

D-boxes: One mystery and one weakness were found in the design of D-boxes:

1. It is not clear why the designers of DES used the initial and final permutations; these have no security benefits.
2. In the expansion permutation (inside the function), the first and fourth bits of every 4-bit series are repeated

Table 6.18 *Weak keys*

Keys before parities drop (64 bits)				*Actual key (56 bits)*	
0101	0101	0101	0101	0000000	0000000
1F1F	1F1F	0E0E	0E0E	0000000	FFFFFFF
E0E0	E0E0	F1F1	F1F1	FFFFFFF	0000000
FEFE	FEFE	FEFE	FEFE	FFFFFFF	FFFFFFF

Table 6.19 *Semi-weak keys*

First key in the pair				*Second key in the pair*			
01FE	01FE	01FE	01FE	FE01	FE01	FE01	FE01
1FE0	1FE0	0EF1	0EF1	E01F	E01F	F10E	F10E
01E0	01E1	01F1	01F1	E001	E001	F101	F101
1FFE	1FFE	0EFE	0EFE	FE1F	FE1F	FE0E	FE0E
011F	011F	010E	010E	1F01	1F01	0E01	0E01
E0FE	E0FE	F1FE	F1FE	FEE0	FEE0	FEF1	FEF1

Why A New Cipher?

❖ DES has the 56-bit key size and being too small.

❖ In January 1999, distributed.net and the Electronic Frontier Foundation collaborated to publicly break a DES key in 22 hours and 15 minutes

❖ **DES had outlived its usefulness:**
- Vulnerabilities were becoming known
- 56-bit key was too small
- Too slow in software implementations

❖ **NIST wanted increased trust in cipher:**
- Previous processes very closed
- DES suspected of having 'back doors'

Advanced Encryption Standard (AES)

❖ **Background**

- On January 2, 1997, NIST announced the initiation of the AES development.

The point stipulated that:

- The algorithm must be a symmetric block cipher
- Key lengths of 128, 192, and 256 bits must be supported
- Block length: 128, 192, and 256 bits
- Both software and hardware implementations must be possible
- Possible implementation on smart-cards
- Royalty-free

The finalists and their scores were as follows: 15 Ciphers submitted

- Rijndael (from Joan Deamen and Vincent Rijmen, 86 votes).
- Serpent (from Ross Anderson, Eli Biham, and Lars Knudsen, 56 votes).
- Twofish (from a team headed by Bruce Schneier, 31 votes).
- RC6 (from RSA Laboratories, 23 votes).
- MARS (from IBM, 13 votes)

- In **November 2001**, Rijndael became a U. S. Government standard published as Federal Information Processing Standard FIPS 197.
- It is not a Feistel cipher.
  - It works in parallel over the whole input block.

- ❖ The most powerful supercomputer in the world in 2017 was the Sunway TaihuLight in China. This beast is capable of a peak speed of 93.02 petaflops. This means that the most powerful computer in the world would still take some 885 quadrillion years to brute force a 128-bit AES key.

- ❖ The number of operations required to brute force a 256-bit cipher is 3.31 x 10^56. This is roughly equal to the number of atoms in the universe!

Rijndael’s Encryption Algorithm

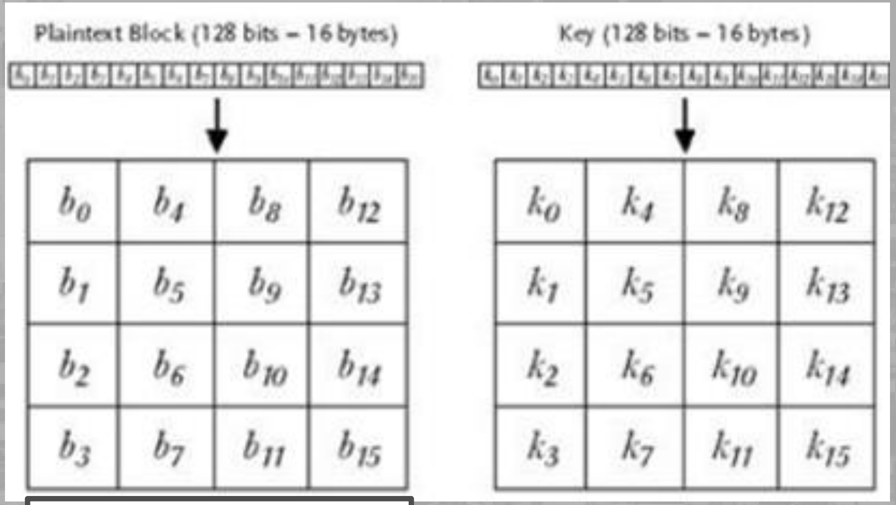
- The basic unit for processing in the AES algorithm is a byte.
- The AES algorithm’s operations are performed on a two-dimensional array of bytes called the State. It is referred as $s_{r,c}$
- Block = 128 bits = 16 byte = $b_0\ b_1\ b_2 \ldots\ldots\ldots\ldots\ldots b_{15}$
- Key = 128 bits = 16 byte = $k_0\ k_1\ k_2 \ldots\ldots\ldots\ldots\ldots k_{15}$
- The four bytes in each column of the State array stand as one word

State representation

$S_{0,0}$	$S_{0,1}$	$S_{0,2}$	$S_{0,3}$
$S_{1,0}$	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$
$S_{2,0}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$
$S_{3,0}$	$S_{3,1}$	$S_{3,2}$	$S_{3,3}$

$w_0 = s_{0,0}\ s_{1,0}\ s_{2,0}\ s_{3,0}$ $\quad w_2 = s_{0,2}\ s_{1,2}\ s_{2,2}\ s_{3,2}$

$w_1 = s_{0,1}\ s_{1,1}\ s_{2,1}\ s_{3,1}$ $\quad w_3 = s_{0,3}\ s_{1,3}\ s_{2,3}\ s_{3,3}$



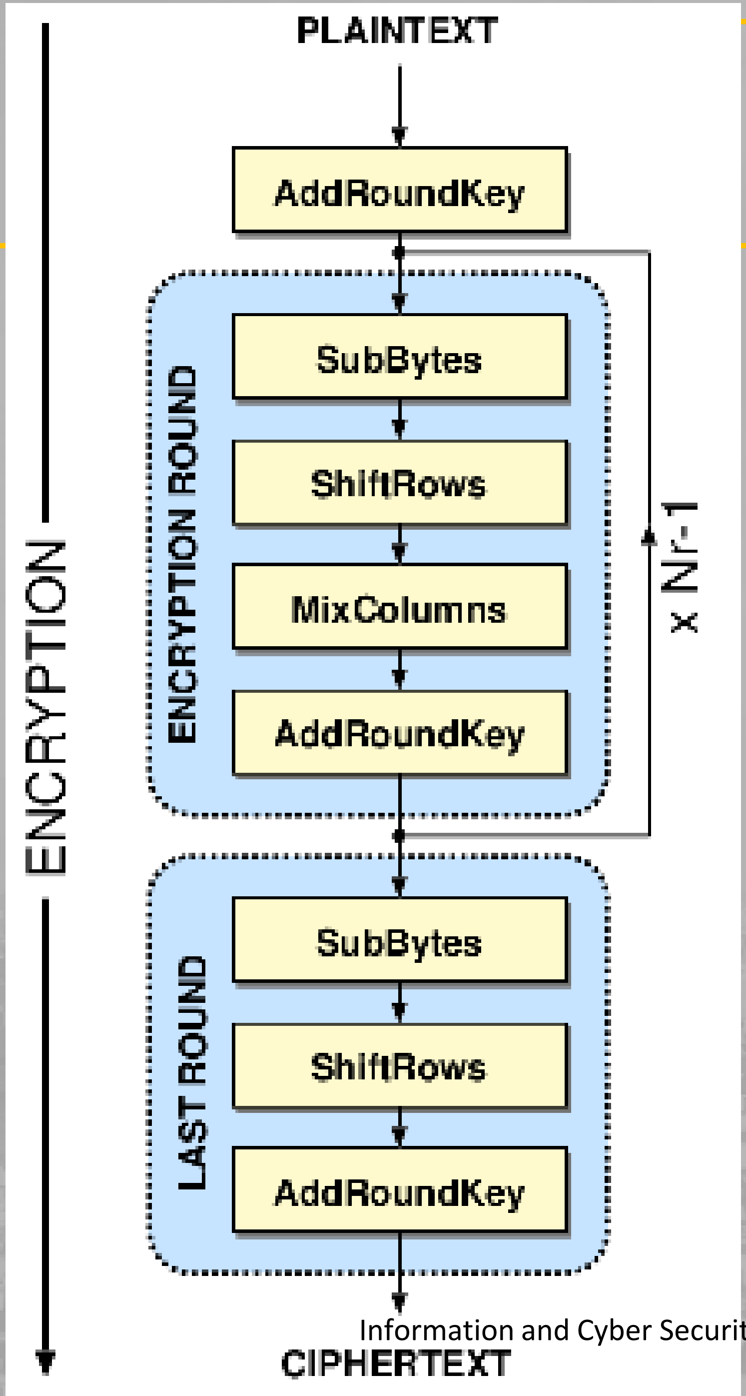
$w_0 \quad w_1 \quad w_2 \quad w_3$

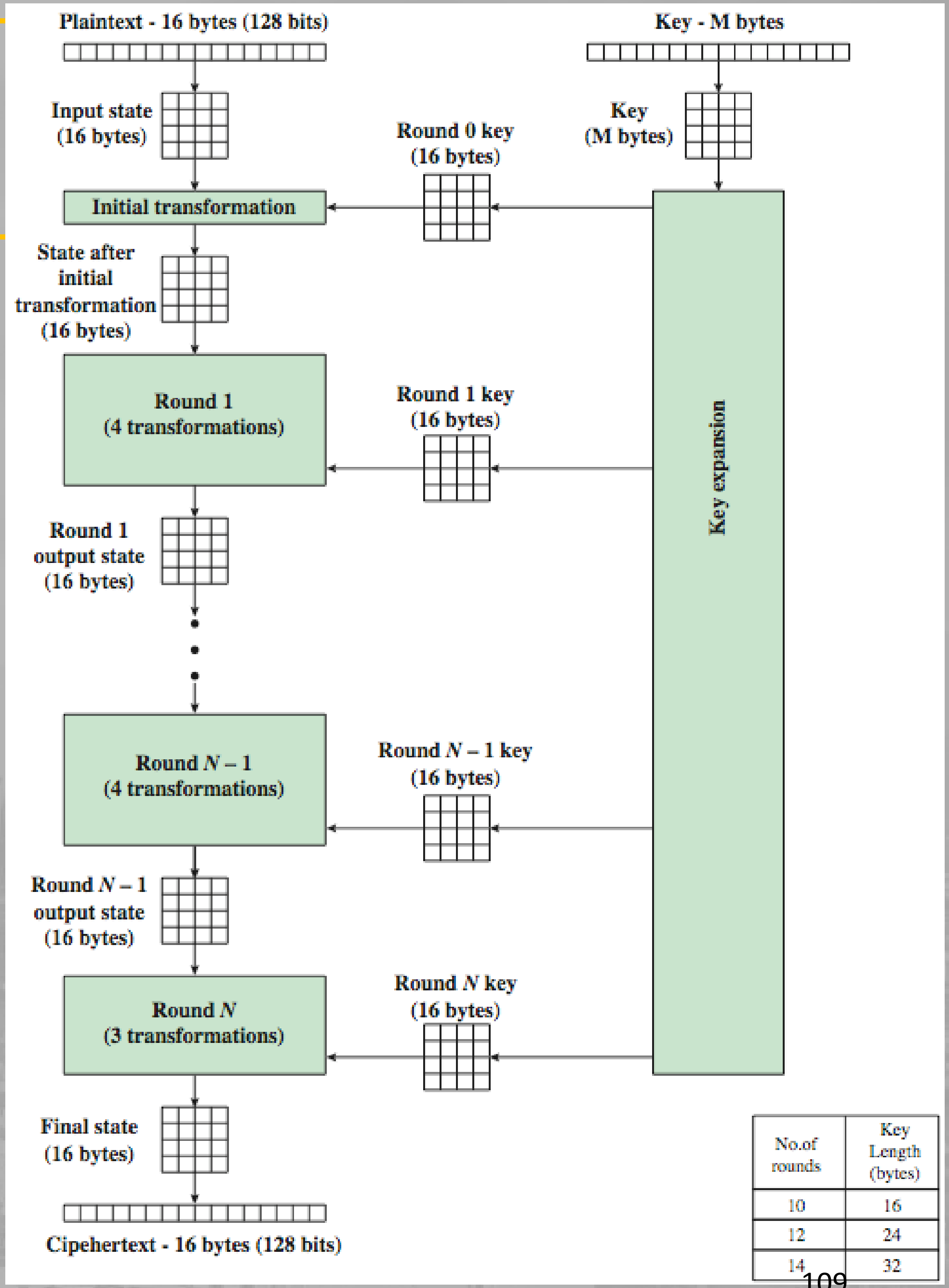
- Implemented as a 4 x 4 matrix, where each element in the matrix is one byte.
- Algorithm consists of an **initial round**, **Nr - 1** standard rounds where Nr is 10, 12, 14 depending on the block and key array sizes, and a **final round**.

Nr	Nb = 4	Nb = 6	Nb = 8
Nk = 4	10	12	14
Nk = 6	12	12	14
Nk = 8	14	14	14

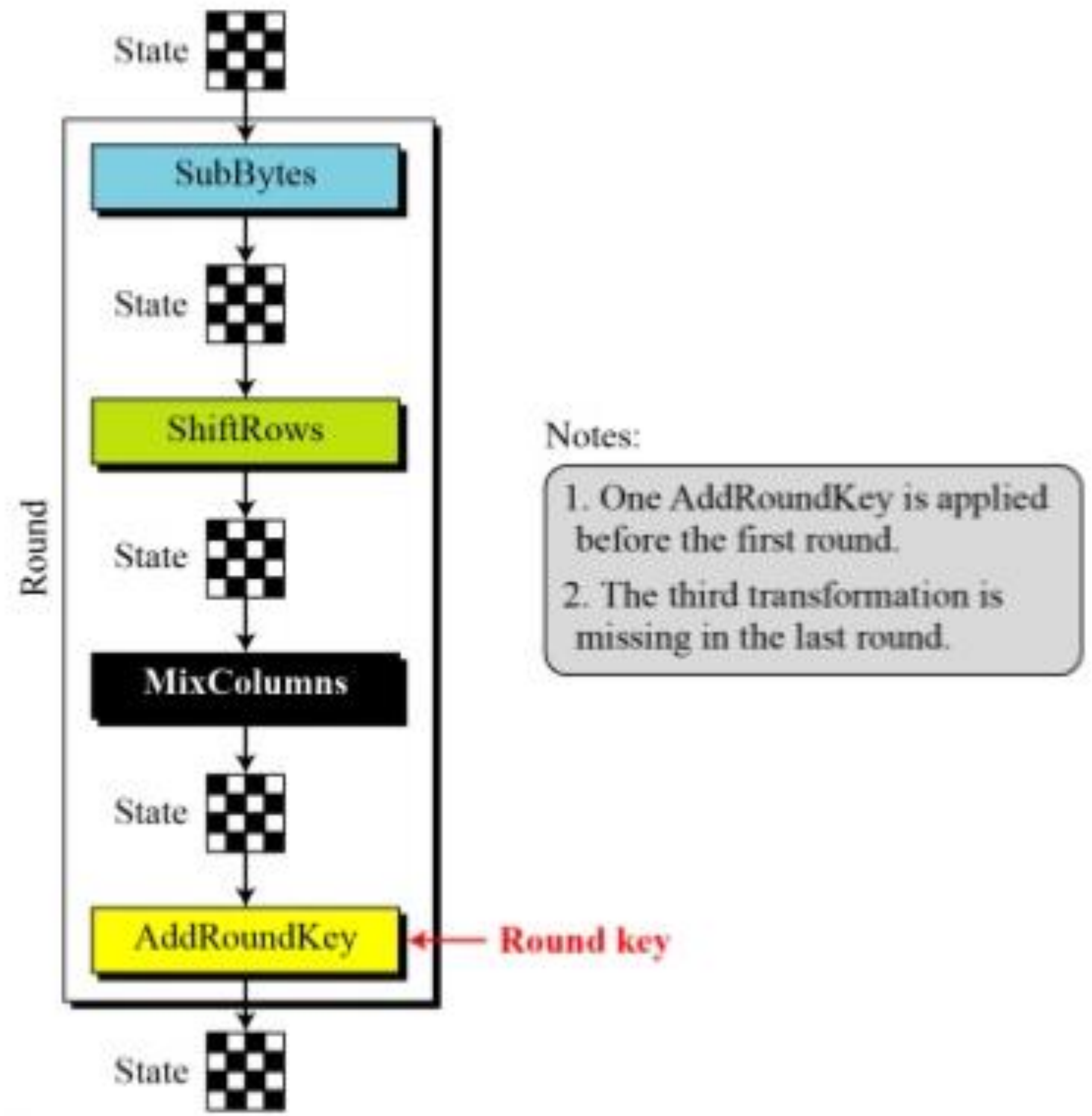
- **Possible Round Operations**
  - ByteSub – Substitution of Bytes
  - Shift Row – Shifts Rows
  - MixColumn – Multiplies Columns
  - AddRoundKey – XORs by Key

Encryption Algorithm





Structure of Each Round

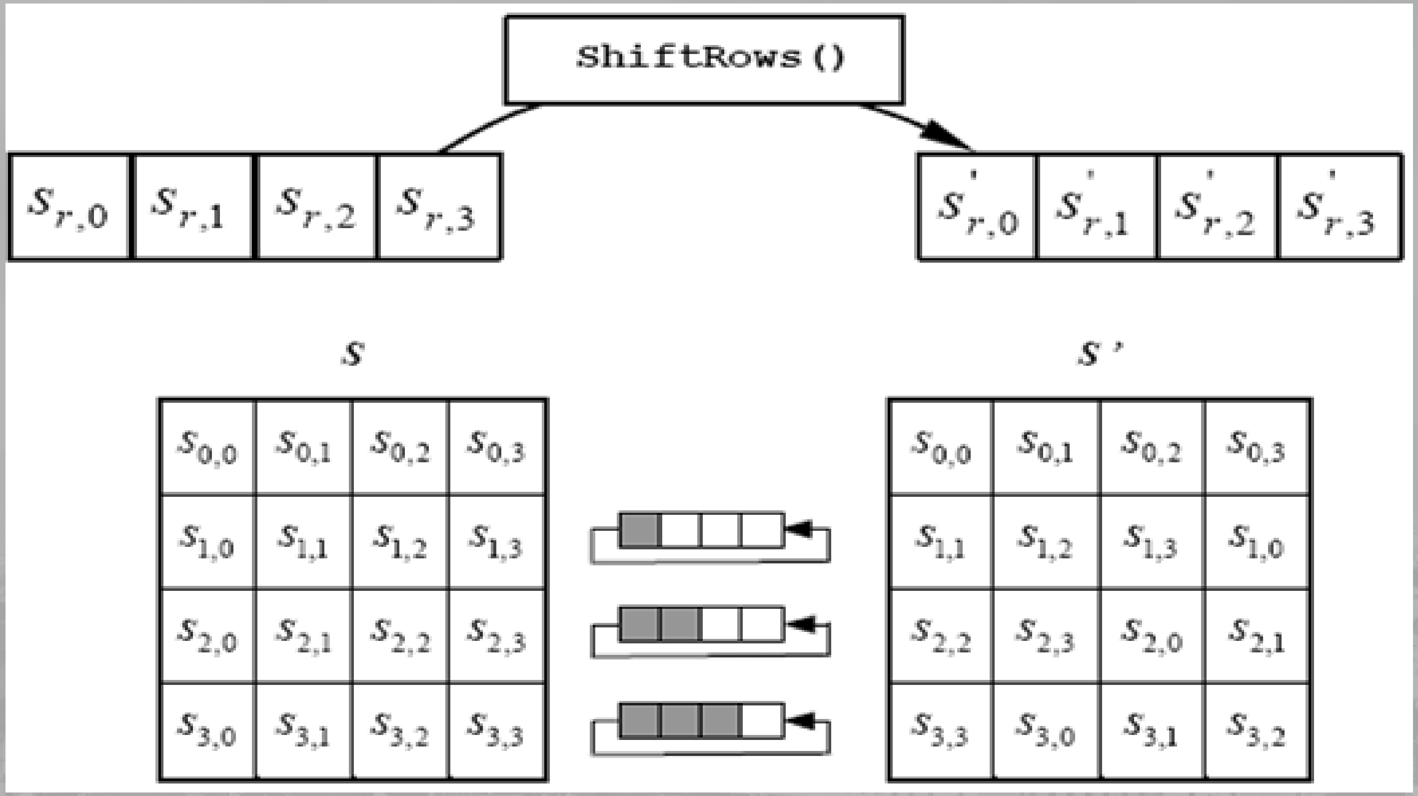


❖ SubBytes transformation

- 16 x 16 matrix whose entries are all distinct bytes.
- For example, if $s_{1,1} = \{53\}$, the result is {ed}.

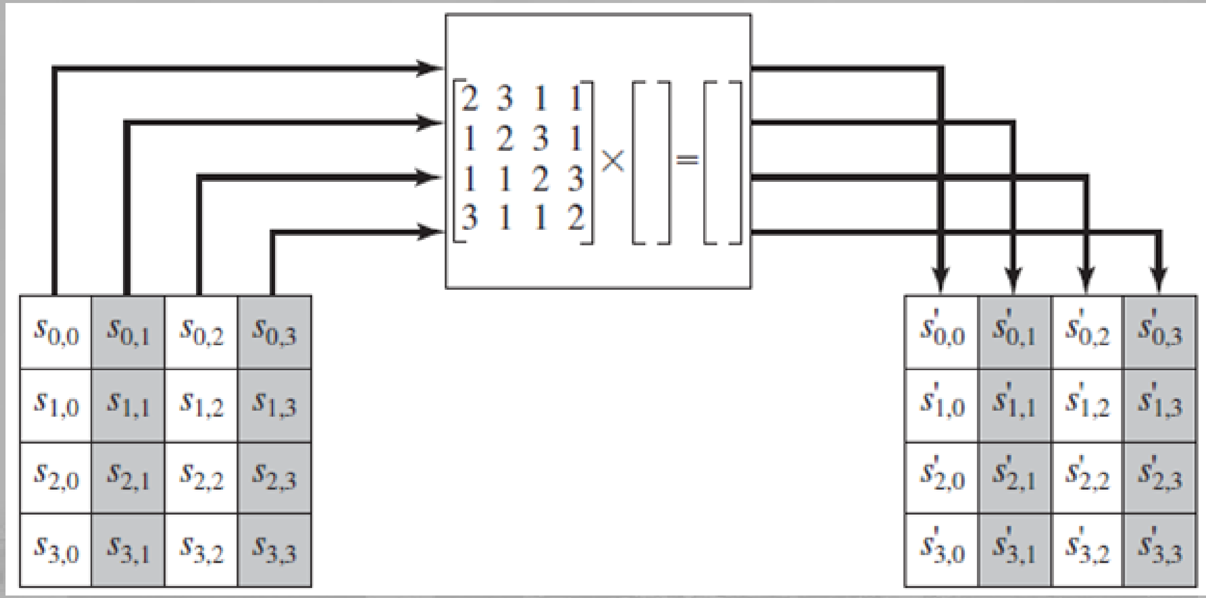
		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

❖ Shift-Rows transformation



❖ MixColumn transformation

- The mixcolumn transformation operates on the state column-by-column.



$$s'_{0,c} = (\{02\} \bullet s_{0,c}) \oplus (\{03\} \bullet s_{1,c}) \oplus s_{2,c} \oplus s_{3,c}$$
$$s'_{1,c} = s_{0,c} \oplus (\{02\} \bullet s_{1,c}) \oplus (\{03\} \bullet s_{2,c}) \oplus s_{3,c}$$
$$s'_{2,c} = s_{0,c} \oplus s_{1,c} \oplus (\{02\} \bullet s_{2,c}) \oplus (\{03\} \bullet s_{3,c})$$
$$s'_{3,c} = (\{03\} \bullet s_{0,c}) \oplus s_{1,c} \oplus s_{2,c} \oplus (\{02\} \bullet s_{3,c})$$

$$\begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} d4 & e0 & b8 & 1e \\ bf & 34 & 41 & 27 \\ 5d & 52 & 11 & 98 \\ 30 & ae & f1 & e5 \end{pmatrix}$$

$$02.\{d4\} \oplus 03.\{bf\} \oplus 01.\{5d\} \oplus 01.\{30\} = \{04\}$$

Multiplication

- A polynomial is irreducible if its only divisors are one and itself.
- For the AES algorithm, this irreducible polynomial is

$$m(x) = x^8 + x^4 + x^3 + x + 1 \quad \text{or} \quad \{01\}\{1b\} \text{ in hexadecimal notation}$$

E.g. $\{57\} \bullet \{83\} = \{c1\}$, because

$$(x^6 + x^4 + x^2 + x + 1)(x^7 + x + 1) = x^{13} + x^{11} + x^9 + x^8 + x^7 + x^7 + x^5 + x^3 + x^2 + x + x^6 + x^4 + x^2 + x + 1$$

$$= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1$$

$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \text{ modulo } x^8 + x^4 + x^3 + x + 1$$

$$= x^7 + x^6 + 1$$

❖ Add Round Key transformation

- In this operation, a Round Key is applied to the State by a simple bitwise XOR .

❖ Key schedule

- This consists of two components: the Key Expansion and the Round Key Selection.

The basic principle is the following:

- The total number of Round Key bits is equal to the block length multiplied by the number of rounds plus 1. (i.e. 128 x 11 = 1408, 1408/32 = 44)
- The Cipher Key is expanded into an Expanded Key.
- Round Keys are taken from this Expanded Key in the following way: the first Round Key consists of the first Nb words, the second one of the following Nb words, and so on.

Key = 2b 7e 15 16 28 ae d2 a6 ab f7 15 88 09 cf 4f 3c

Key expansion

The key expansion function depends on the value of Nk.

For Nk ≤ 6, we have:

```
KeyExpansion (byte Key [4*Nk], word W [Nb*(Nr+1)])
{
    for (i = 0; i < Nk; i++)
    {
            W[i] = (Key [4*i], Key [4*i+1], Key [4*i+2], Key [4*i+3])
    }
    for (i = Nk; i < Nb * (Nr + 1); i++)
    {       temp = W [i - 1];
            if (i % Nk = = 0)
            {
                    temp = SubByte (RotByte (temp)) XOR  Rcon [i / Nk]
            }
            W[i] = W[i - Nk]  XOR  temp
    }
}
```

- The round constants are independent of Nk and defined by:

  Rcon[i] = (RC[i], ‘00’, ‘00’, ‘00’)

  with RC[1] representing an element in GF(2^8) with a value of $x^{(i-1)}$ so that:

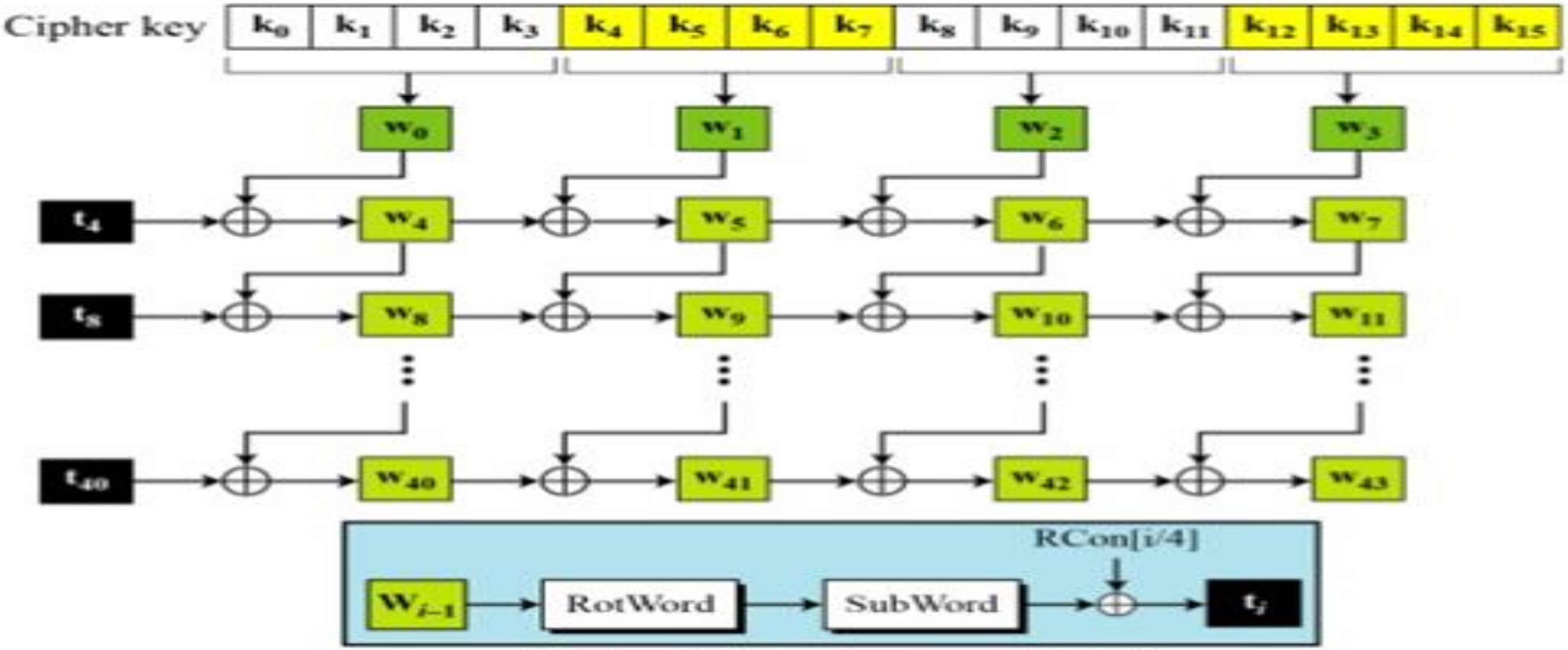
  RC[1] = 1 (i.e. ‘01’)

  RC[i] = x (i.e. ‘02’) • (RC[i-1]) = $x^{(i-1)}$

Table 7.4 *RCon constants*

Round	*Constant (RCon)*	*Round*	*Constant (RCon)*
1	$(\underline{01}\ 00\ 00\ 00)_{16}$	6	$(\underline{20}\ 00\ 00\ 00)_{16}$
2	$(\underline{02}\ 00\ 00\ 00)_{16}$	7	$(\underline{40}\ 00\ 00\ 00)_{16}$
3	$(\underline{04}\ 00\ 00\ 00)_{16}$	8	$(\underline{80}\ 00\ 00\ 00)_{16}$
4	$(\underline{08}\ 00\ 00\ 00)_{16}$	9	$(\underline{1B}\ 00\ 00\ 00)_{16}$
5	$(\underline{10}\ 00\ 00\ 00)_{16}$	10	$(\underline{36}\ 00\ 00\ 00)_{16}$

Key Expansion in AES-128



Making of t_i (temporary) words $i = 4\,N_r$

Key Expansion Examples

Cipher Key = 2b 7e 15 16 28 ae d2 a6 ab f7 15 88 09 cf 4f 3c

For **Nk = 4**, which results in

w_0 = 2b7e1516 w_1 = 28aed2a6 w_2 = abf71588 w_3 = 09cf 4f3c

i (dec)	temp	After RotWord()	After SubWord()	Rcon[i/Nk]	After XOR with Rcon	w[i−Nk]	w[i]= temp XOR w[i−Nk]
4	09cf4f3c	cf4f3c09	8a84eb01	01000000	8b84eb01	2b7e1516	a0fafe17
5	a0fafe17					28aed2a6	88542cb1
6	88542cb1					abf71588	23a33939
7	23a33939					09cf4f3c	2a6c7605
8	2a6c7605	6c76052a	50386be5	02000000	52386be5	a0fafe17	f2c295f2
9	f2c295f2					88542cb1	7a96b943
10	7a96b943					23a33939	5935807a
11	5935807a					2a6c7605	7359f67f
12	7359f67f	59f67f73	cb42d28f	04000000	cf42d28f	f2c295f2	3d80477d
13	3d80477d					7a96b943	4716fe3e
14	4716fe3e					5935807a	1e237e44
15	1e237e44					7359f67f	6d7a883b
16	6d7a883b	7a883b6d	dac4e23c	08000000	d2c4e23c	3d80477d	ef44a541
17	ef44a541					4716fe3e	a8525b7f
18	a8525b7f					1e237e44	b671253b
19	b671253b					6d7a883b	db0bad00
20	db0bad00	0bad00db	2b9563b9	10000000	3b9563b9	ef44a541	d4d1c6f8
21	d4d1c6f8					a8525b7f	7c839d87
22	7c839d87					b671253b	caf2b8bc
23	caf2b8bc					db0bad00	11f915bc

Example: Nb = 4 and Nk = 4

Input : 32 43 f6 a8 88 5a 30 8d 31 31 98 a2 e0 37 07 34

Key : 2b 7e 15 16 28 ae d2 a6 ab f7 15 88 09 cf 4f 3c

	32	88	31	e0														2b	28	ab	09
																		[illegible]	ae	f7	cf
																		[illegible]	d2	15	4f
																		[illegible]	a6	88	3c

The Round Key values a

Rou

Nun

	19	a0	9a	e9	d4	e0	b8	1e	d4	e0	b8	1e	04	e0	48	28		a0	88	23	2a
	3d	f4	c6	f8	27	bf	b4	41	bf	b4	41	27	66	cb	f8	06		fa	54	a3	6c
1	e3	e2	8d	48	11	98	5d	52	5d	52	11	98	81	19	d3	26	⊕	fe	2c	39	76
	be	2b	2a	08	ae	f1	e5	30	30	ae	f1	e5	e5	9a	7a	4c		17	b1	39	05

	a4	68	6b	02	49	45	7f	77	49	45	7f	77	58	1b	db	1b		f2	7a	59	73
	9c	9f	5b	6a	de	db	39	02	db	39	02	de	4d	4b	e7	6b		c2	96	35	59
2	7f	35	ea	50	d2	96	87	53	87	53	d2	96	ca	5a	ca	b0	⊕	95	b9	80	f6
	f2	2b	43	49	89	f1	1a	3b	3b	89	f1	1a	f1	ac	a8	e5		f2	43	7a	7f

	aa	61	82	68		ac	ef	13	45		ac	ef	13	45		75	20	53	bb		3d	47	1e	6d	
3	8f	dd	d2	32		73	c1	b5	23		c1	b5	23	73		ec	0b	c0	25	⊕	80	16	23	7a	=
	5f	e3	4a	46		cf	11	d6	5a		d6	5a	cf	11		09	63	cf	d0		47	fe	7e	88	
	03	ef	d2	9a		7b	df	b5	b8		b8	7b	df	b5		93	33	7c	dc		7d	3e	44	3b	

	48	67	4d	d6		52	85	e3	f6		52	85	e3	f6		0f	60	6f	5e		ef	a8	b6	db	
4	6c	1d	e3	5f		50	a4	11	cf		a4	11	cf	50		d6	31	c0	b3	⊕	44	52	71	0b	=
	4e	9d	b1	58		2f	5e	c8	6a		c8	6a	2f	5e		da	38	10	13		a5	5b	25	ad	
	ee	0d	38	e7		28	d7	07	94		94	28	d7	07		a9	bf	6b	01		41	7f	3b	00	

	e0	c8	d9	85		e1	e8	35	97		e1	e8	35	97		25	bd	b6	4c		d4	7c	ca	11	
5	92	63	b1	b8		4f	fb	c8	6c		fb	c8	6c	4f		d1	11	3a	4c	⊕	d1	83	f2	f9	=
	7f	63	35	be		d2	fb	96	ae		96	ae	d2	fb		a9	d1	33	c0		c6	9d	b8	15	
	e8	c0	50	01		9b	ba	53	7c		7c	9b	ba	53		ad	68	8e	b0		f8	87	bc	bc	

	f1	c1	7c	5d		a1	78	10	4c		a1	78	10	4c		4b	2c	33	37		6d	11	db	ca	
6	00	92	c8	b5		63	4f	e8	d5		4f	e8	d5	63		86	4a	9d	d2	⊕	88	0b	f9	00	=
	6f	4c	8b	d5		a8	29	3d	03		3d	03	a8	29		8d	89	f4	18		a3	3e	86	93	
	55	ef	32	0c		fc	df	23	fe		fe	fc	df	23		6d	80	e8	d8		7a	fd	41	fd	

	26	3d	e8	fd		f7	27	9b	54		f7	27	9b	54		14	46	27	34		4e	5f	84	4e	
7	0e	41	64	d2		ab	83	43	b5		83	43	b5	ab		15	16	46	2a	⊕	54	5f	a6	a6	=
	2e	b7	72	8b		31	a9	40	3d		40	3d	31	a9		b5	15	56	d8		f7	c9	4f	dc	
	17	7d	a9	25		f0	ff	d3	3f		3f	f0	ff	d3		bf	ec	d7	43		0e	f3	b2	4f	

	5a	19	a3	7a		be	d4	0a	da		be	d4	0a	da		00	b1	54	fa		ea	b5	31	7f	
8	41	49	e0	8c		83	3b	e1	64		3b	e1	64	83		51	c8	76	1b	⊕	d2	8d	2b	8d	=
	42	dc	19	04		2c	86	d4	f2		d4	f2	2c	86		2f	89	6d	99		73	ba	f5	29	
	b1	1f	65	0c		c8	c0	4d	fe		fe	c8	c0	4d		d1	ff	cd	ea		21	d2	60	2f	

	ea	04	65	85		87	f2	4d	97		87	f2	4d	97		47	40	a3	4c		ac	19	28	57	
9	83	45	5d	96		ec	6e	4c	90		6e	4c	90	ec		37	d4	70	9f	⊕	77	fa	d1	5c	=
	5c	33	98	b0		4a	c3	46	e7		46	e7	4a	c3		94	e4	3a	42		66	dc	29	00	
	f0	2d	ad	c5		8c	d8	95	a6		a6	8c	d8	95		ed	a5	a6	bc		f3	21	41	6e	

	eb	59	8b	1b		e9	cb	3d	af		e9	cb	3d	af							d0	c9	e1	b6	
10	40	2e	a1	c3		09	31	32	2e		31	32	2e	09						⊕	14	ee	3f	63	=
	f2	38	13	42		89	07	7d	2c		7d	2c	89	07							f9	25	0c	0c	
	1e	84	e7	d2		72	5f	94	b5		b5	72	5f	94							a8	89	c8	a6	

	39	02	dc	19
output	25	dc	11	6a
	84	09	85	0b
	1d	fb	97	32

- **Addition**
  - The addition is performed with the XOR operation (denoted by $\oplus$) - i.e., modulo 2 - so that

    $1 \oplus 1 = 0$, $1 \oplus 0 = 1$, and $0 \oplus 0 = 0$.

$(x^6 + x^4 + x^2 + x + 1) + (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2$ (polynomial notation)

$\{01010111\} \oplus \{10000011\} = \{11010100\}$ (binary notation)

$\{57\} \oplus \{83\} = \{d4\}$ (hexadecimal notation)

Multiplication

- A polynomial is irreducible if its only divisors are one and itself.
- For the AES algorithm, this irreducible polynomial is

$$m(x) = x^8 + x^4 + x^3 + x + 1 \quad \text{or} \quad \{01\}\{1b\} \text{ in hexadecimal notation}$$

E.g. {57} • {83} = {c1}, because

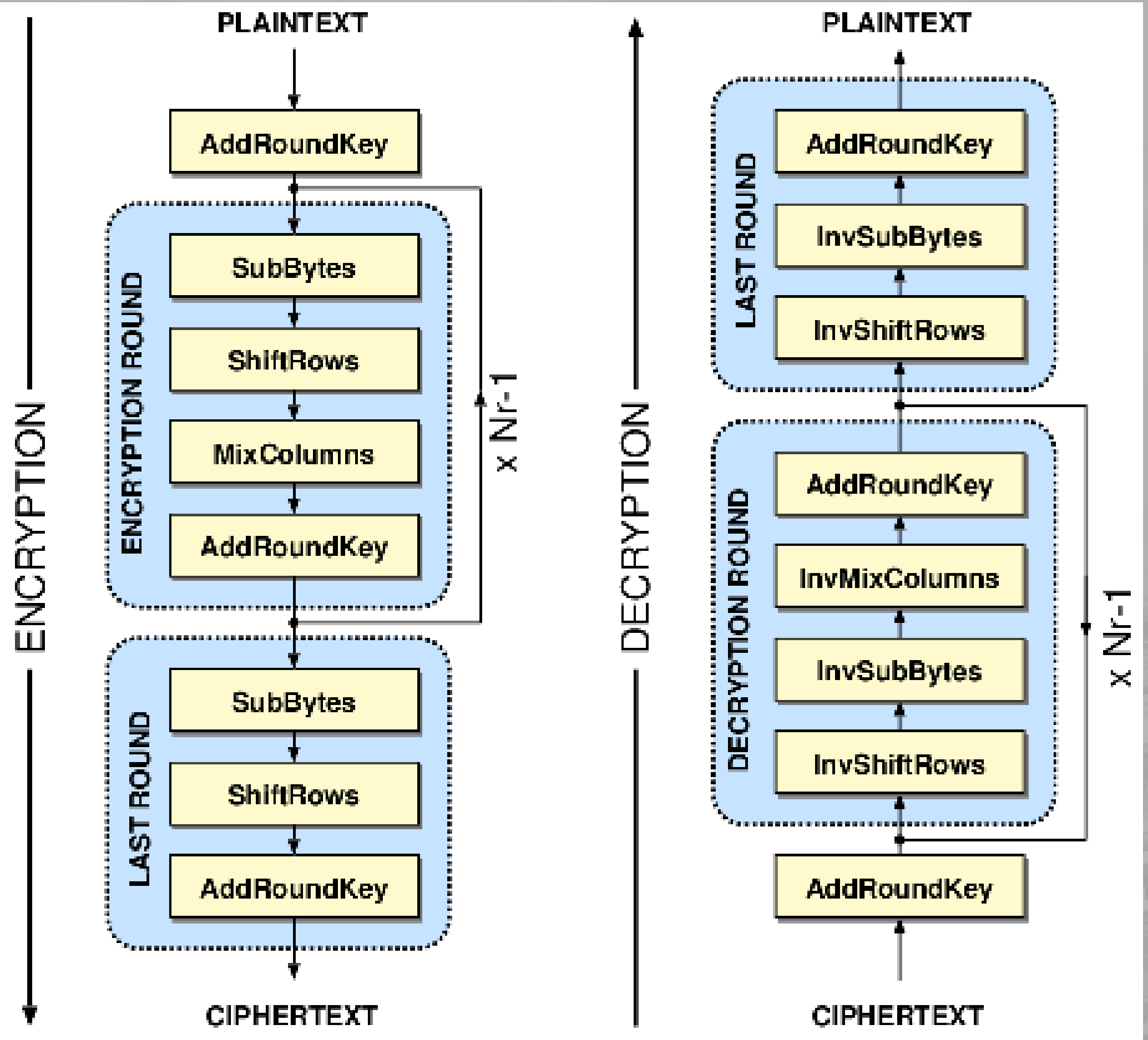
$$(x^6 + x^4 + x^2 + x + 1)(x^7 + x + 1) = x^{13} + x^{11} + x^9 + x^8 + x^7 + x^7 + x^5 + x^3 + x^2 + x + x^6 + x^4 + x^2 + x + 1$$

$$= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1$$

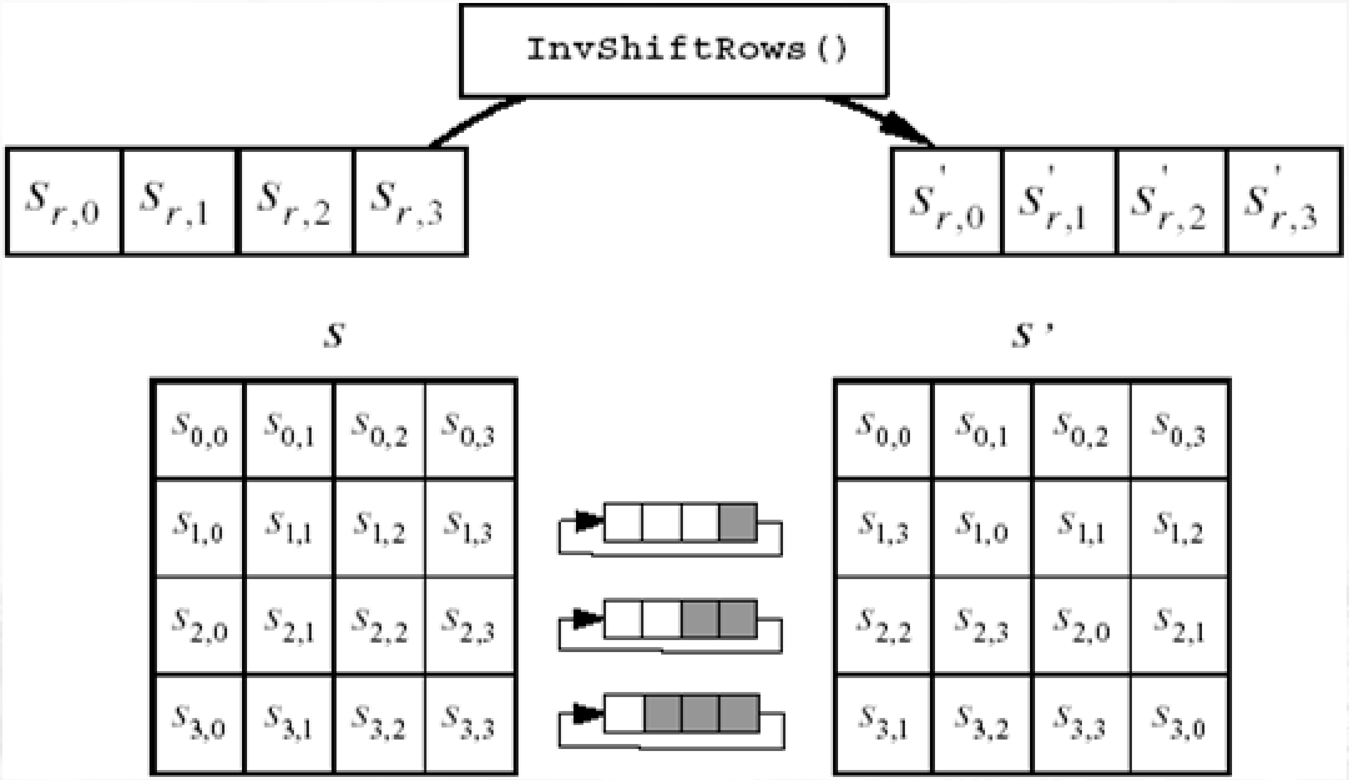
$$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \text{ modulo } x^8 + x^4 + x^3 + x + 1$$

$$= x^7 + x^6 + 1$$

❖ Rijndael's Decryption Algorithm



Inverse Shift-Rows transformation



❖ Inverse Sub-Bytes transformation

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
	1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
	2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
	3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
	4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
	5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
	6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
	7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
	8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	73
	9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
	a	47	f1	1a	71	1d	29	c5	89	6f	b7	62	0e	aa	18	be	1b
	b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
	c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
	d	60	51	7f	a9	19	b5	4a	0d	2d	e5	7a	9f	93	c9	9c	ef
	e	a0	e0	3b	4d	ae	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
	f	17	2b	04	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d

❖ Inverse Mix-column transformation

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \quad \text{for } 0 \le c < Nb.$$

$$s'_{0,c} = (\{0e\} \bullet s_{0,c}) \oplus (\{0b\} \bullet s_{1,c}) \oplus (\{0d\} \bullet s_{2,c}) \oplus (\{09\} \bullet s_{3,c})$$

$$s'_{1,c} = (\{09\} \bullet s_{0,c}) \oplus (\{0e\} \bullet s_{1,c}) \oplus (\{0b\} \bullet s_{2,c}) \oplus (\{0d\} \bullet s_{3,c})$$

$$s'_{2,c} = (\{0d\} \bullet s_{0,c}) \oplus (\{09\} \bullet s_{1,c}) \oplus (\{0e\} \bullet s_{2,c}) \oplus (\{0b\} \bullet s_{3,c})$$

$$s'_{3,c} = (\{0b\} \bullet s_{0,c}) \oplus (\{0d\} \bullet s_{1,c}) \oplus (\{09\} \bullet s_{2,c}) \oplus (\{0e\} \bullet s_{3,c})$$

Advantages

- Rijndael can be implemented to run at speeds unusually fast for a block cipher on a Pentium (Pro). There is a trade-off between table size/performance.

- Rijndael can be implemented on a Smart Card in a small amount of code, using a small amount of RAM and taking a small number of cycles. There is some ROM/performance trade-off.

- The round transformation is parallel by design, an important advantage in future processors and dedicated hardware.

- As the cipher does not make use of arithmetic operations, it has no bias towards big-or little endian processor architectures.

Limitations

- ❖ The inverse cipher is less suited to be implemented on a smart card than the cipher itself: it takes more code and cycles. (Still, compared with other ciphers, even the inverse is very fast)

- ❖ In software, the cipher and its inverse make use of different code and/or tables.

- ❖ In hardware, the inverse cipher can only partially re-use the circuitry that implements the cipher.

Comparison

BASIS FOR COMPARISON	DES (DATA ENCRYPTION STANDARD)	AES (ADVANCED ENCRYPTION STANDARD)
Basic	In DES, the data block is divided into two halves.	In AES, the entire data block is processed as a single matrix.
Principle	DES work on Feistel Cipher structure.	AES work on block Cipher structure.
Plaintext	Plaintext is of 64 bits	Plaintext can be of 128,192, or 256 bits
Key size	DES in comparison to AES has smaller key size.	AES has larger key size as compared to DES.
Rounds	16 rounds	10 rounds for 128-bit algo, 12 rounds for 192-bit algo 14 rounds for 256-bit algo
Rounds Names	Expansion Permutation, Xor, S-box, P-box, Xor and Swap.	Subbytes, Shiftrows, Mix columns, Addroundkeys.
Security	DES has a smaller key which is less secure.	AES has large secret key comparatively hence, more secure.
Speed	DES is comparatively slower.	AES is faster.

Thank You !!!!!!