

Symmetric cryptography; modern encryption algorithms (DES, AES)

Prepared by Dr. Ayman Khaled

IT373

Information Security

Lecture 2



Course content

1. Overview of security concepts and definitions
2. **Symmetric cryptography; modern encryption algorithms (DES, AES)**
3. modern data encryption standards
4. block cipher cryptanalysis and usage
5. modern stream ciphers
6. Asymmetric cryptography; public key cryptography, key certificates and management
7. message authentication and hash functions
8. hash algorithms
9. digital signature and authentication protocols
10. electronic mail security, IP security, web security.

Agenda/Outline

- Cryptography
 - History
 - Modern Cryptology Origins
 - Classification of Cryptography Algorithms
- Symmetric Encryption
 - Requirements for Secure Symmetric Encryption
 - Symmetric Block Encryption Algorithms
 - Data Encryption Standard (D E S)
 - Advanced Encryption Standard (A E S)



Cryptography: Definition and Key Concepts

- Cryptography is the practice of securing communication and information by transforming it into an unreadable format for unauthorized users.



Cryptography

- Classified along three independent dimensions:
 - The type of operations used for transforming plaintext to ciphertext
 - Substitution – each element in the plaintext is mapped into another element
 - Transposition – elements in plaintext are rearranged
 - The number of keys used
 - Sender and receiver use same key – symmetric
 - Sender and receiver each use a different key - asymmetric
 - The way in which the plaintext is processed
 - Block cipher – processes input one block of elements at a time
 - Stream cipher – processes the input elements continuously



Cryptography: Definition and Key Concepts

- **Plaintext:** This is the original message or data that is fed into the algorithm as input.
- **Encryption algorithm:** The encryption algorithm performs various substitutions and transformations on the plaintext.
- **Secret key:** The secret key is also input to the algorithm. The exact substitutions and transformations performed by the algorithm depend on the key.
- **Ciphertext:** This is the scrambled message produced as output. It depends on the plaintext and the secret key. For a given message, two different keys will produce two different ciphertexts.
- **Decryption algorithm:** This is essentially the encryption algorithm run in reverse. It takes the ciphertext and the same secret key and produces the original plaintext.



History of Cryptography

- **Ancient Egypt (circa 1900 BC)**
- **Mesopotamia (circa 1500 BC)**
- **India (400 BC – 200 AD)**
- **Ancient Greece**
- **Roman Empire**



A Scytale, an early device for encryption.



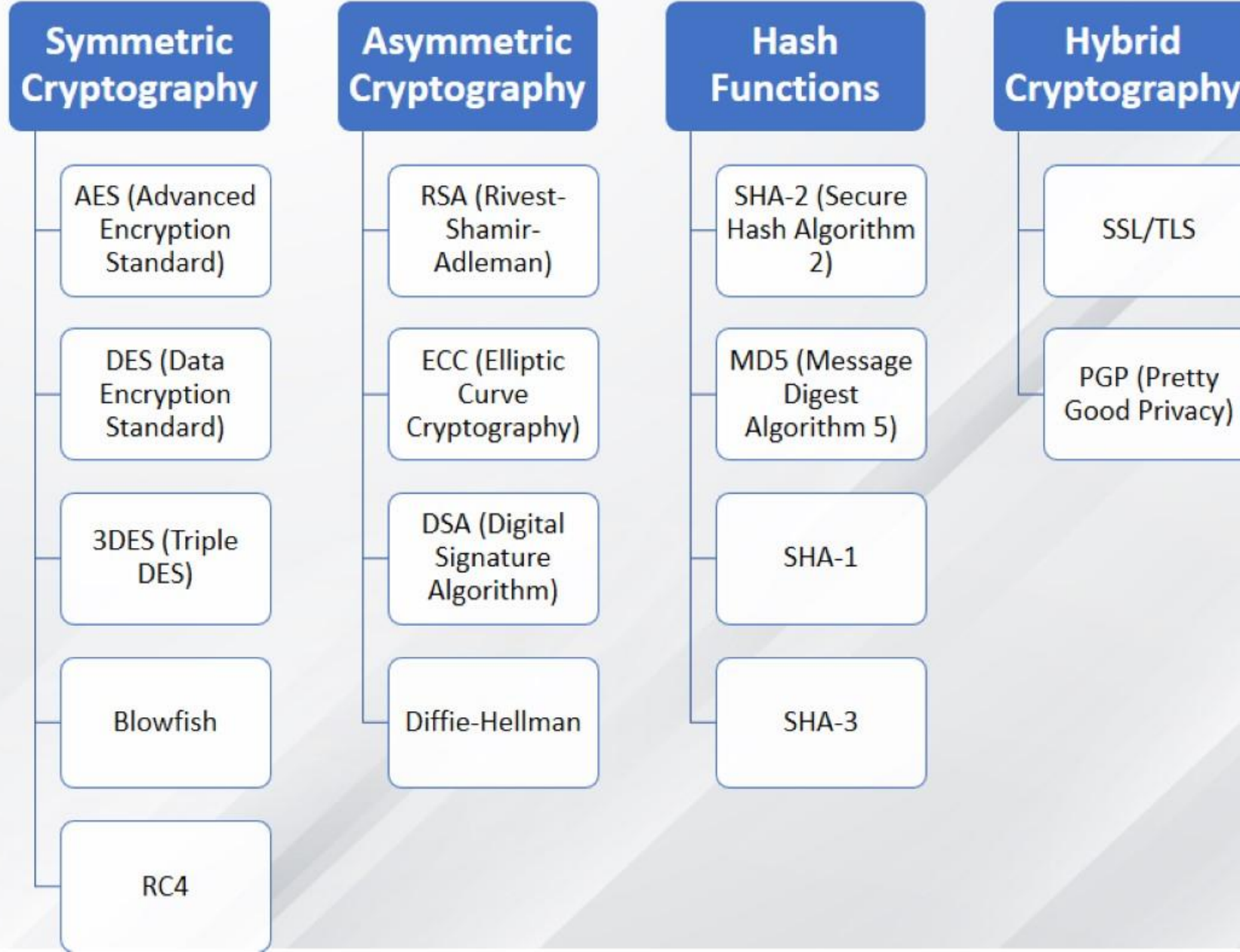
Modern Cryptology Origins

David Kahn (The Codebreakers) notes that cryptology systematically originated among the Arabs.

- **Al-Khalil (717–786)**
 - Authored *The Book of Cryptographic Messages*.
 - Introduced the first use of permutations and combinations for cryptanalysis.
- **Al-Kindi (c. 800 AD)**
 - Developed frequency analysis, the most significant cryptanalytic method until WWII.
 - Wrote *Risalah fi Istikhraj al-Mu'amma* (Manuscript for the Deciphering of Cryptographic Messages).
- **Ibn al-Durayhim (1312–1361)**
 - Described **substitution and transposition ciphers**.
 - Introduced an early **polyalphabetic cipher** and methods for cryptanalysis.



Classification of Cryptography Algorithms



Overview

- Symmetric encryption ensures **confidentiality** for transmitted or stored data.
- Commonly referred to as:
 - **Conventional encryption**
 - **Single-key encryption**
- Historical Use
 - Used by figures like Julius Caesar and during WWII by the German U-boat forces.
 - Still widely employed in diplomatic, military, and commercial communication.



Requirements for Secure Symmetric Encryption

Strong Encryption Algorithm

- Opponents knowing the algorithm and having access to ciphertexts should not be able to:
 - Decipher ciphertext.
 - Discover the key.
- Even with access to both ciphertext and the plaintext used to produce it, the key must remain secure.

Secure Key Management

- Both sender and receiver must securely obtain and protect the secret key.
- If the key is compromised, all communication using it becomes vulnerable.



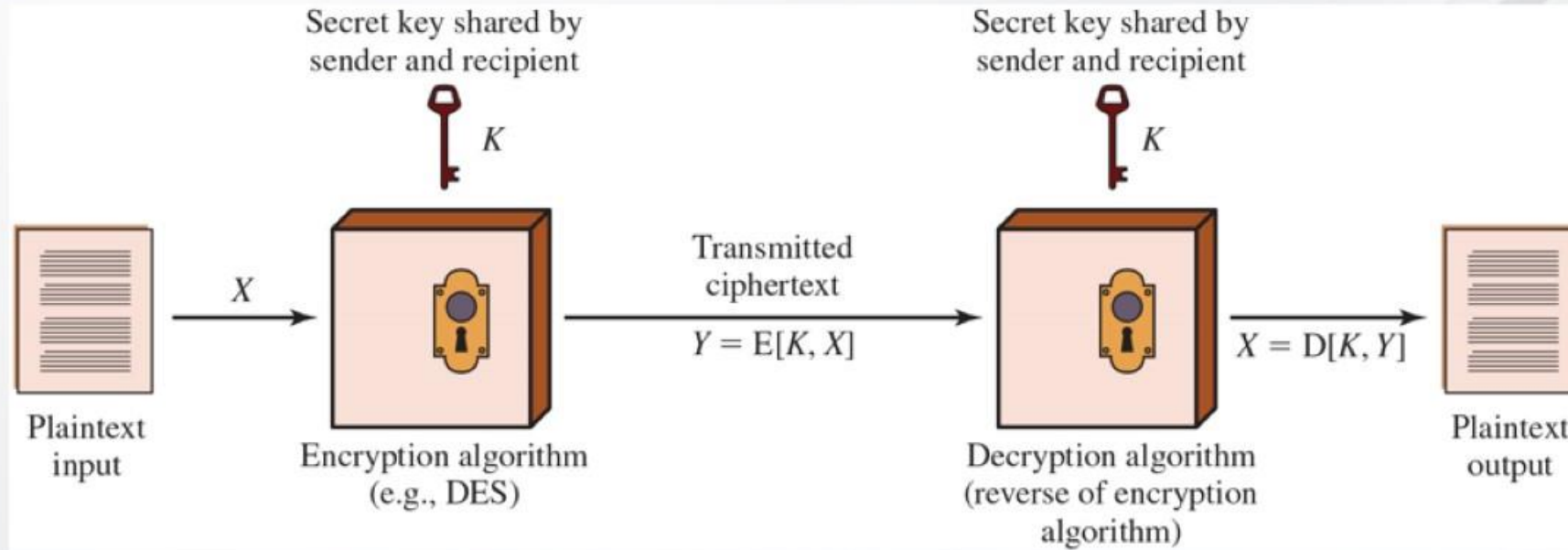
Symmetric Encryption: An Overview

- **Key Features:**
 - **Single Key:** A shared secret key between sender and receiver.
 - **Efficiency:** Symmetric encryption algorithms are generally faster and more efficient than asymmetric methods.
 - **Common Uses:** Secure communications, file encryption, and protecting sensitive data.



Figure 2.1

Simplified Model of Symmetric Encryption



Attacking Symmetric Encryption

Cryptanalytic Attacks

- Rely on:
 - Nature of the algorithm
 - Some knowledge of the general characteristics of the plaintext
 - Some sample plaintext-ciphertext pairs
- Exploits the characteristics of the algorithm to attempt to deduce a specific plaintext or the key being used
 - If successful all future and past messages encrypted with that key are compromised



Attacking Symmetric Encryption

Brute-Force Attacks

- Try all possible keys on some ciphertext until an intelligible translation into plaintext is obtained
 - On average half of all possible keys must be tried to achieve success



Symmetric Block Encryption Algorithms

- Symmetric encryption algorithms primarily consist of block ciphers.
- Block ciphers process plaintext in fixed-size blocks, producing an equal-sized ciphertext block for each plaintext block.
- Longer plaintexts are processed as a series of fixed-size blocks.
- Key Algorithms:
 - Data Encryption Standard (DES)
 - Triple DES (3DES)
 - Advanced Encryption Standard (AES)



Table 2.1

Comparison of Three Popular Symmetric Encryption Algorithms

	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



Data Encryption Standard (DES)

- For many years was the most widely used encryption scheme
 - Federal Information Processing Standards Unified Process (FIPS UB) 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



Data Encryption Standard (DES)

- Strength concerns:
 - Concerns about the algorithm itself
 - DES is the most studied encryption algorithm in existence
 - Concerns about the use of a 56-bit key
 - The speed of commercial off-the-shelf processors makes this key length woefully inadequate

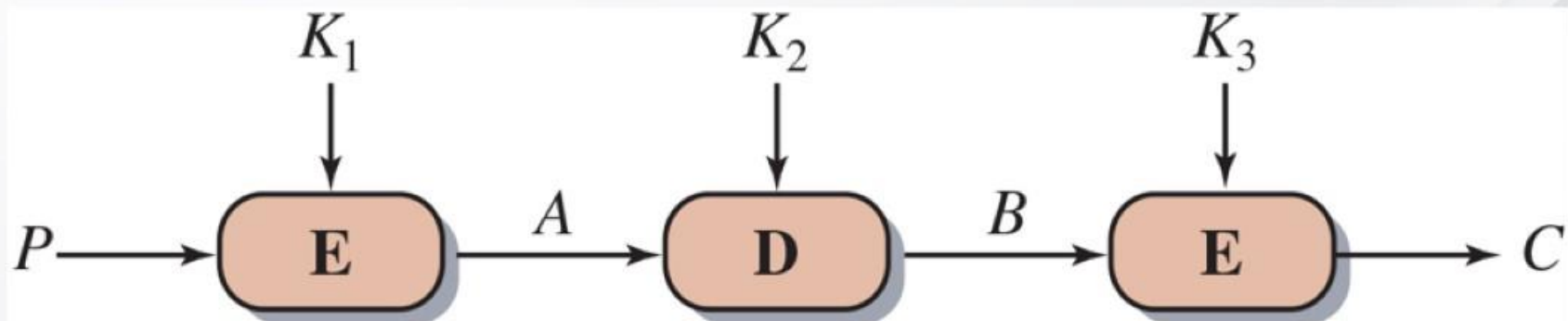


Triple DES (3DES)

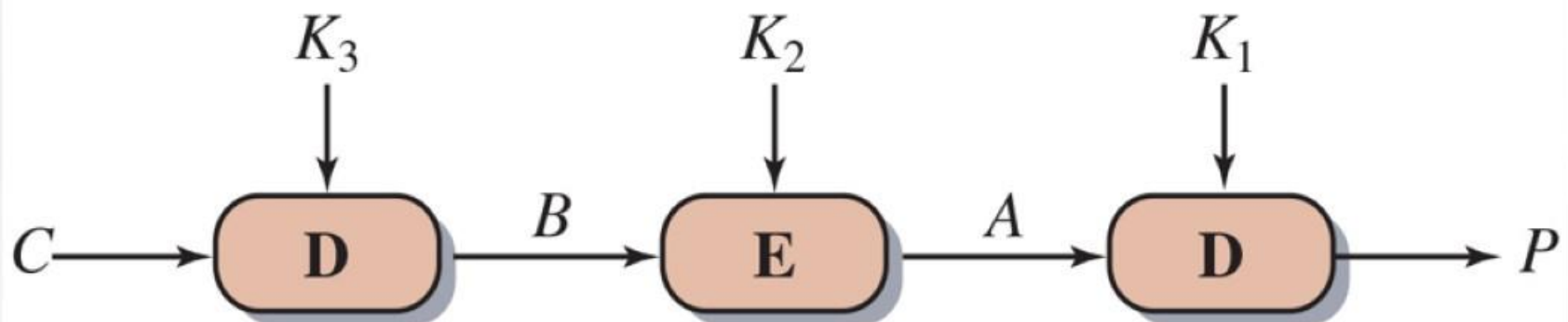
- Repeats basic DES algorithm three times using either two or three unique keys
- First standardized for use in financial applications in ANSI standard X9.17 in 1985
- Attractions:
 - 168-bit key length overcomes DES's vulnerability to brute-force attack
 - Underlying encryption algorithm is the same as in DES
- Drawbacks:
 - Algorithm is sluggish in software
 - Uses a 64-bit block size



Triple DES



(a) Encryption



(b) Decryption



Triple DES (3DES) Encryption

- 3DES (Triple DES): An extension of DES that applies the encryption algorithm three times using three keys.
 - Encryption Process:
 - Encrypt with K_1 : $E(K_1, P)$
 - Decrypt with K_2 : $D(K_2, E(K_1, P))$
 - Encrypt with K_3 : $E(K_3, D(K_2, E(K_1, P)))$
- Result: Ciphertext (C)
- Formula:
$$C = E(K_3, D(K_2, E(K_1, P)))$$
- Effective Key Length:
 - 168 bits with three distinct keys.



Triple DES (3DES) Decryption

- **Decryption Process:**

- Reverse the key order used in encryption.

- **Decryption Formula:**

$$P = D(K_1, E(K_2, D(K_3, C)))$$



Average Time Required for Exhaustive Key Search

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



Computationally Secure Encryption Schemes

- Encryption is computationally secure if:
 - Cost of breaking cipher exceeds value of information
 - Time required to break cipher exceeds the useful lifetime of the information
- Usually very difficult to estimate the amount of effort required to break
- Can estimate time/cost of a brute-force attack



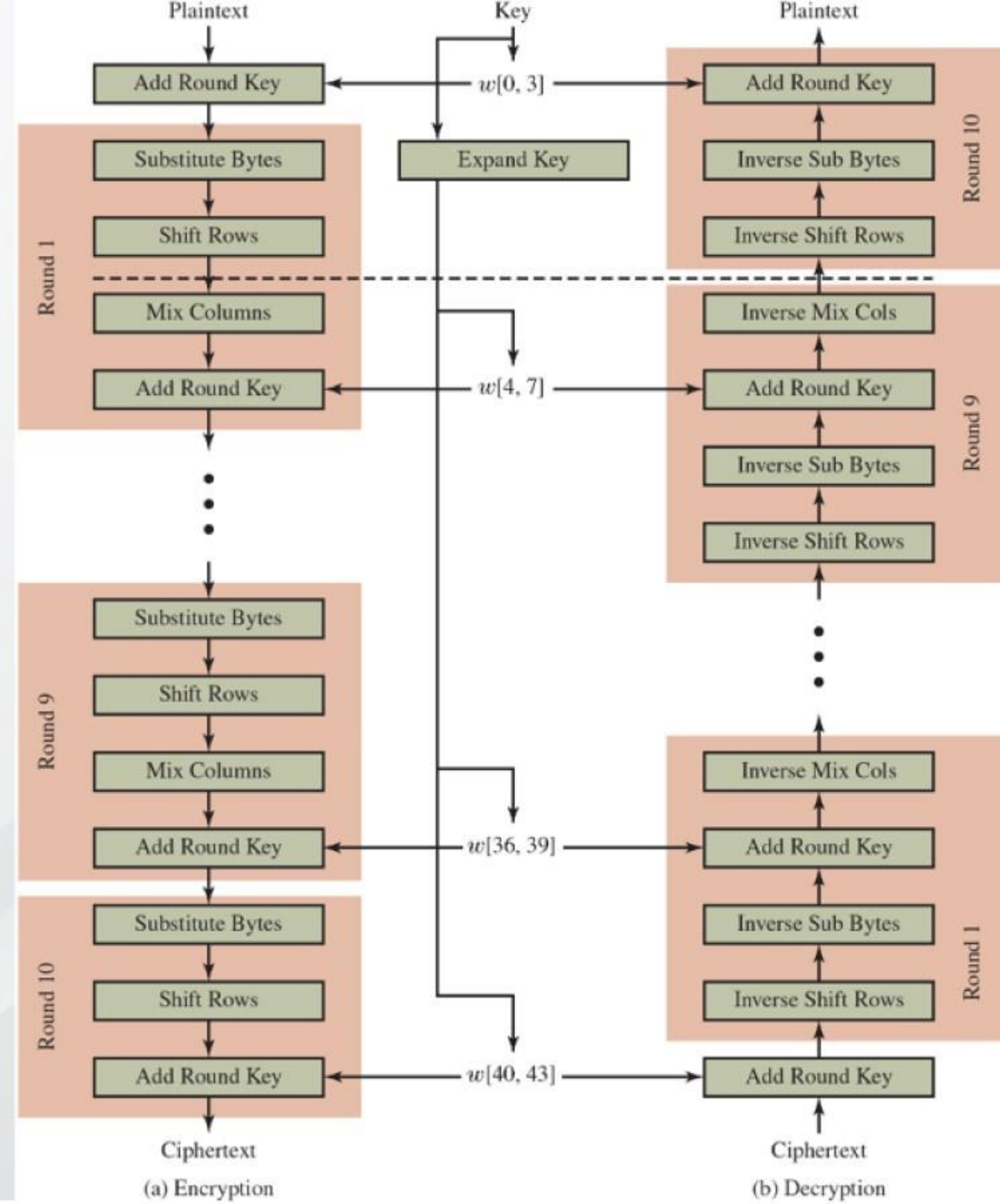
Advanced Encryption Standard (AES)

- Needed a replacement for 3DES
 - 3DES was not reasonable for long-term use
- NIST called for proposals for a new AES in 1997
 - Would have a security strength equal to or better than 3 DES
 - Would significantly improve efficiency
 - Symmetric block cipher
 - 128 bit data and 128/192/256 bit keys
- Selected Rijndael in November 2001
 - Published as FIPS 197



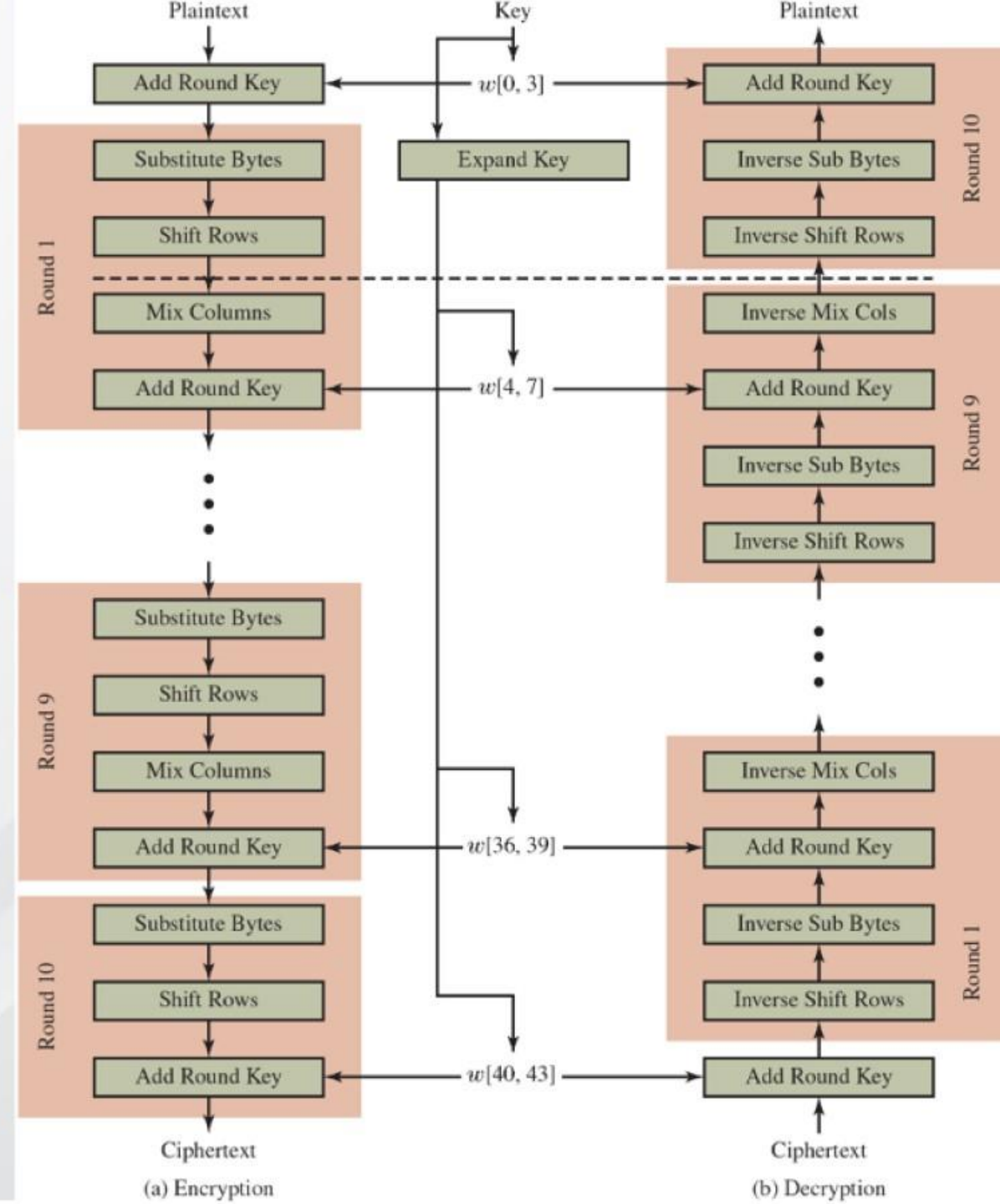
Overall structure of AES

- The structure is simple for both encryption and decryption.
- Begins with **Add Round Key** stage.
- Followed by **nine rounds**, each with four stages.
- Ends with a **tenth round** consisting of three stages.



Role of Add Round Key

- **Add Round Key** stage is the only one using the key.
- Cipher starts and ends with this stage for security.
- Other stages at the beginning or end can be reversed without the key, adding no security.



Importance of Additional Stages

- **Add Round Key** alone is not secure.
- The other three stages (Substitute Byte, Shift Row, Mix Columns) scramble bits.
- Combined, these operations provide security through:
 - **XOR encryption** (Add Round Key).
 - **Bit scrambling** (other three stages).
 - This scheme is efficient and highly secure.



XOR overview

P=1010

K=1101 $\oplus$

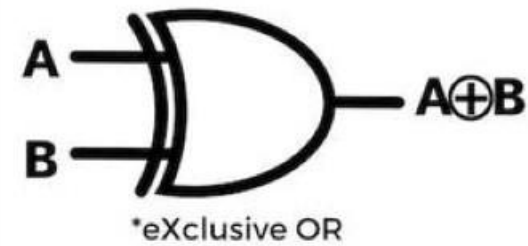


C= 0111

K=1101 $\oplus$

C= 0111

P=1010

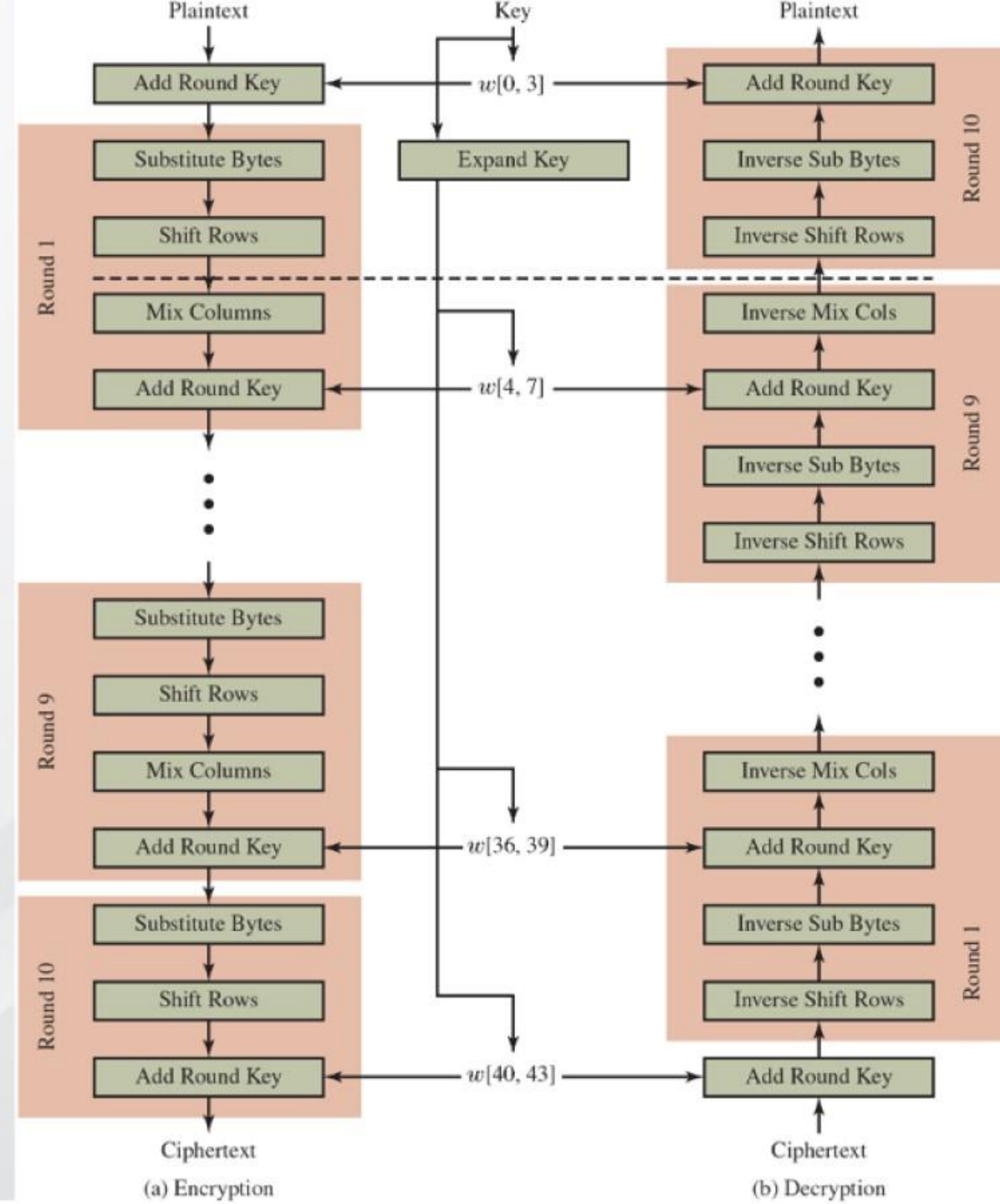


2 input XOR gate

A	B	A $\oplus$ B
0	0	0
0	1	1
1	0	1
1	1	0



Overall structure of AES



SubStitute Bytes Transformation

EA	04	65	85
83	45	5D	96
5C	33	98	B0
F0	2D	AD	C5

→

87	F2	4D	97
EC	6E	4C	90
4A	C3	46	E7
8C	D8	95	A6

Example Transformation

- Byte: 04 (hexadecimal)
- S-box Lookup:
 - Row: 0
 - Column: 4
- Substituted Value: F2

(a) S-box

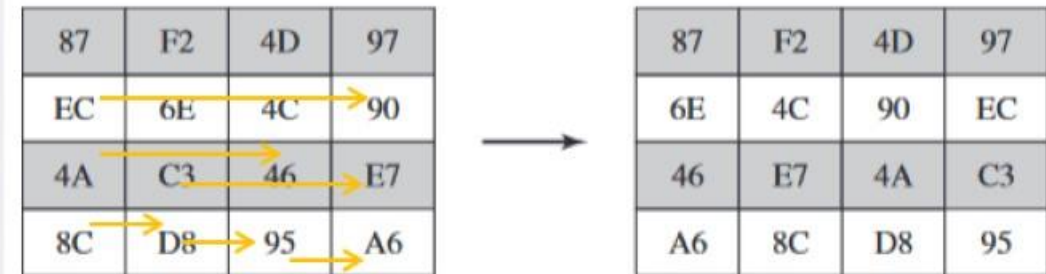
		y															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
x	0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
	1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
	2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
	3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
	4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	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 Row Transformation

- **Purpose:** Achieves diffusion by rearranging the data within the state matrix.
- **Mechanism:** Data is shifted in each row with wrapping.

Row Shifting Details:

- **First Row:** Remains unchanged.
- **Second Row:** Shifted left by **1** position.
- **Third Row:** Shifted left by **2** positions.
- **Fourth Row:** Shifted left by **3** positions.

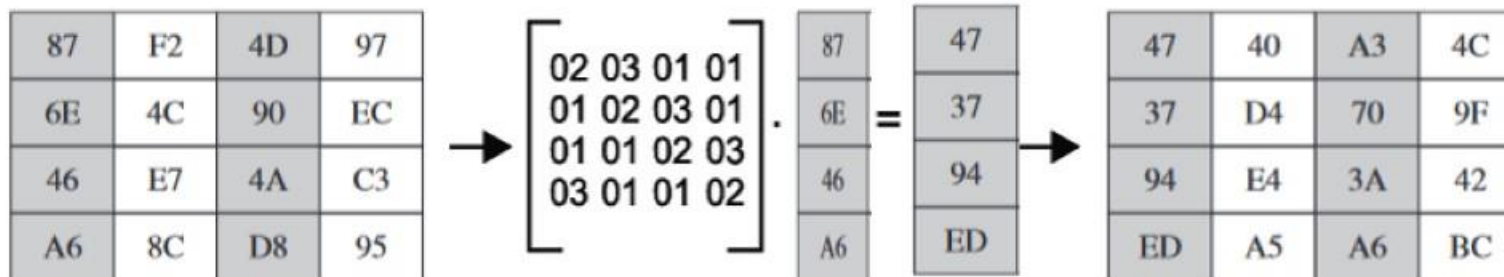


MixColumns Transformation

- **Purpose:** A transformation that operates on each column of the state individually.
- **Mechanism:** Each byte in a column is mapped to a new value based on all 4 bytes in that column..

Transformation Process:

- Utilizes equations over **finite fields** to compute new byte values.
- Ensures effective mixing of the bytes within each column.



Add Round Key Transformation in AES

- **Forward Transformation:**

The 128 bits of the State are bitwise XORed with the 128 bits of the round key, operating as a column-wise and byte-level transformation.

- **Inverse Transformation:**

The inverse process is identical, as XOR is its own inverse. This simple operation affects every bit of the State, contributing to AES security through round key complexity.

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

$\oplus$

AC	19	28	57
77	FA	D1	5C
66	DC	29	00
ED	A5	A6	BC

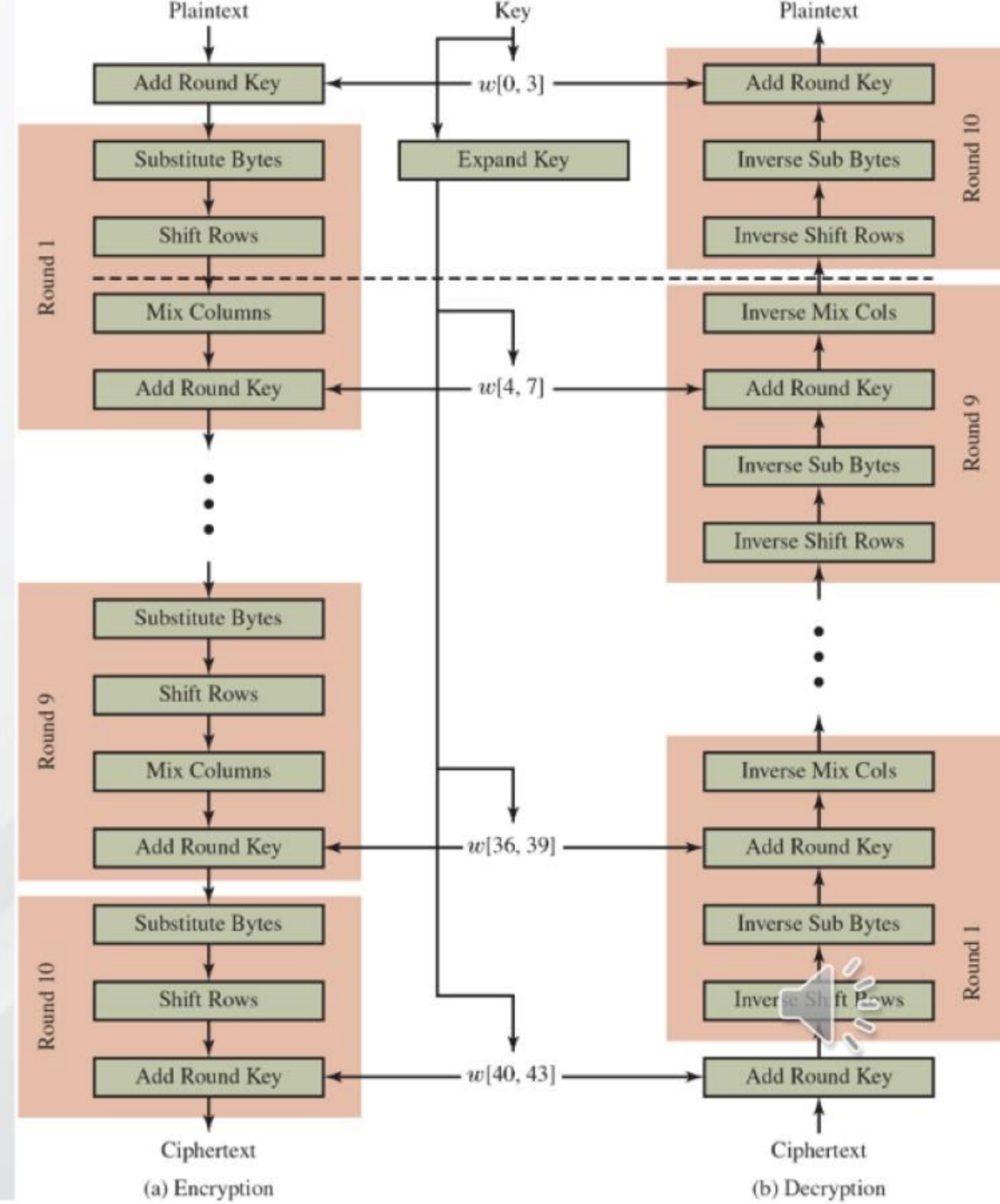
=

EB	59	8B	1B
40	2E	A1	C3
F2	38	13	42
1E	84	E7	D2



AES Key Expansion

The AES key expansion algorithm takes a 4-word (16-byte) key and produces a linear array of 44 words (176 bytes), providing sufficient keys for the initial Add Round Key stage and each of the 10 rounds.



Practical Security Issues

- Typically, symmetric encryption is applied to a unit of data larger than a single 64-bit or 128-bit block
- Electronic codebook (ECB) mode is the simplest approach to multiple-block encryption
 - Each block of plaintext is encrypted using the same key
 - Cryptanalysts may be able to exploit regularities in the plaintext
- Modes of operation
 - Alternative techniques developed to increase the security of symmetric block encryption for large sequences
 - Overcome the weaknesses of ECB



Advanced Encryption Standard(AES) is a symmetric encryption algorithm. AES is the industry standard as of now as it allows 128 bit, 192 bit and 256 bits encryption. Symmetric encryption is rapid as compared to asymmetric encryption and are used in systems such as database system. Following is an online tool to perform AES encryption and decryption of any plain-text or password.

The tool provides multiple modes of encryption and decryption such as **ECB, CBC, CTR and GCM mode**. **GCM** is considered more secure than CBC mode and is widely adopted for its performance.

For more info on AES encryption, visit [this explanation on AES Encryption](#). Below is the form to take the inputs for the encryption and decryption.

AES Encryption

Enter Plain Text to Encrypt

Enter plain text to be Encrypted

Select Cipher Mode of Encryption

CBC

Select Padding

PKCS5Padding

Enter IV (Optional)

Enter initialization vector

Key Size in Bits

128

Enter Secret Key

Enter secret key

Output Text Format ☒ Base64 ☐ Hex

Encrypt

AES Encrypted Output

Result goes here

AES Decryption

AES Encrypted Text

Enter text to be Decrypted

Select Cipher Mode of Decryption

CBC

Select Padding

PKCS5Padding

Enter IV Used During Encryption(Optional)

Enter initialization vector

Key Size in Bits

128

Enter Secret Key used for Encryption

Enter secret key

Output Text Format ☒ Plain-Text ☐ Base64

Decrypt

AES Decrypted Output

Decrypted result goes here

Any secret key value that you enter, or we generate is not stored on this site, this tool is provided via an HTTPS URL to ensure that any secret keys cannot be stolen.

If you appreciate this tool, then you can consider donating.

Summary

Cryptography

- History

- Modern Cryptology Origins

- Classification of Cryptography Algorithms

Symmetric Encryption

- Requirements for Secure Symmetric Encryption

- Symmetric Block Encryption Algorithms

- Data Encryption Standard (D E S)

- Advanced Encryption Standard (A E S)





Thank You

