Security and Protection of Systems

Lecture 05
Wireless & Mobile Systems Security

Dr. Zaher Haddad

Outlines

- Bluetooth Systems
- Near Field Communication Systems
- Radio Frequency Identification Systems
- WiFi Systems
- Mobile Systems
- Security Solutions

Outlines

- Bluetooth Systems
- Near Field Communication Systems
- Radio Frequency Identification Systems
- WiFi Systems
- Mobile Systems
- Security Solutions

3

Bluetooth Systems

- Bluetooth systems are applications that use Bluetooth technology that depends on short range radio frequency transmission.
- Examples: Wireless Mouse – Wireless Keyboard – PAN – BAN.
- Piconet is established when two devices come within range each others.
- Master: controlling all wireless traffic.
- Slave: Receive commands

1. Active Slave: Scanning transmissions
2. Parked Slave: Connecting without participation

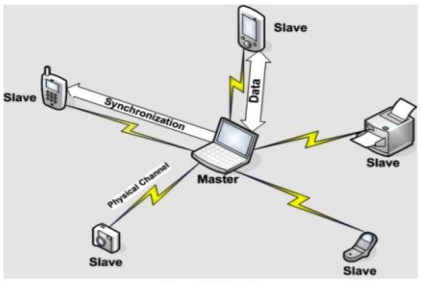


Fig. PICONET

Bluetooth Attacks

1- Bluejacking Attack:

- Sends undesirable packets to Bluetooth enabled devices. (text – image – sound – video ….)
- Consider more annoying than harmful
- No data stolen

2- Bluesnarfing Attack:

- Accessed unauthorized information from wireless device through a Bluetooth connection
- Copies information without owner knowledge such as coping emails and contact.

Outlines

- Bluetooth Systems
- Near Field Communication Systems
- Radio Frequency Identification Systems
- WiFi Systems
- Mobile Systems
- Security Solutions

Near Field Communications

- NFC is a technology used to establish communication between devices in close distance.
- Once two devices close to each other in the range less than 20 cm, can read information as well as transmit data.

Near Field Communications Attacks

1- Eavesdropping NFC:

- An unencrypted NFC communication can be intercepted and viewed
- Attacker must be extremely close to pick the NFC communication Therefore, users should remain aware of their surrounding while making services such as payments.

2- MitM NFC:

An attacker can intercept the NFC communication between devices and forge a facilitate response.

Note: Devices can be configured pairing so one device can only send and the other can only receive..

Outlines

- Bluetooth Systems
- Near Field Communication Systems
- Radio Frequency Identification Systems
- WiFi Systems
- Mobile Systems
- Security Solutions

9

Radio Frequency Identification (RFID)

- Commonly used to transmit information between employee identification badges, inventory tags, book labels, and other paper-based tags that can be detected by a proximity reader.
- Most RFID are passive
  - Because, it is a very small and did not require a power supply.
- RFID tags are susceptible to different attacks
- Contains some security enhancements over the previous version

RFID Attacks

1- Unauthorized tag access

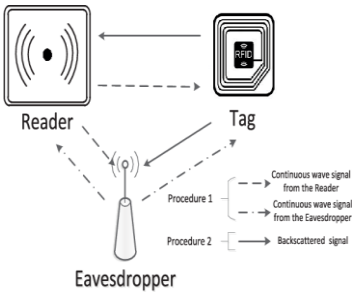
- A Rogue RFID reader can determine the inventory on a store shell to track the sales of specific items.
- Sales information could be used by a rival product manufactures to negotiate additional shell space of better product placement.

2- Fake tags

- Fake tags with some fictitious information are replaced instead of authenticated tags.
- Fictitious information that are given by fake tags gives illusion to user about product that not exist in the store

3- Eavesdropping Tags:

- Unauthorized user could track on transmission between RDIF user and readers.
- Reveal secret information to rivals



Outlines

- Bluetooth Systems
- Near Field Communication Systems
- Radio Frequency Identification Systems
- WiFi Systems
- Mobile Systems
- Security Solutions

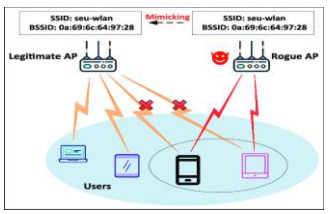
WiFi Systems

- Commonly used for local area networking of devices and Internet access, allowing nearby digital devices to exchange data by radio waves.
- Most WiFi attacks are
  1. Rogue access point
  2. Evil twins
  3. Intercepting wireless data

WiFi Attacks

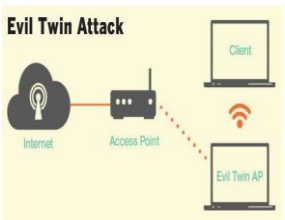
1- Rogue access point

- An unauthorized access point that allows an attacker to bypass network security configuration
- Usually setup by an insider (Employee)
- May be setup behind a firewall, opening the network to attack



2- Evil twins

- Access point setup by an attacker
- Attempts to mimic an authorized access point
- Attackers capture transmissions from users to evil twins access point



3- Intercepting wireless transmission

- An attacker can pick up the Radio frequency signal from an open misconfigured access point
- Using WiFi to read this data could yield significant information to an attacker regarding the wired enterprise network.
- Could be Replay, MitM, DoS attack

Outlines

- Bluetooth Systems
- Near Field Communication Systems
- Radio Frequency Identification Systems
- WiFi Systems
- Mobile Systems
- Security Solutions

15

Mobile devices

- Types: Tablet: Smartphones, Wearable, Laptop, Notebook
- Connectivity Methods: Cellular, Satellite, Infrared, ANT, USB connections
  - ANT- A proprietor wireless network technology used by sensors for communicating data
- Mobile devices may faces different types of security risks such as :
  1. Mobile device vulnerabilities
  2. Connection vulnerabilities

Mobile Device Vulnerabilities (1)

- Physical Security: A result of the loss or theft of mobile devices
- Limited firmware update: Apple iOS is closed and propriety architecture, Google does not create the hardware that Android run on.
- Location Tracking: Mobile devices with GPS capabilities support geolocation, Attacker can determine where user are and plan to steal information or inflict harm
- Unauthorized Recording: Mobile activated could be recorded during the service.

Connection Vulnerabilities (2)

- Tethering: Mobile device with an active internet connection can be used, therefore, an unauthorized mobile device may infect other tethered mobile devices or the corporate network.
- USB in-the-Go (OTG): Mobile device with a USB connection can act as either a host or a peripheral used for external media access, therefore, an infected computer could allow malware to be sent to the device.
- Connecting to the public network: Mobile devices must at time use public external networks for internet access, therefore, attacker can eavesdropping data transmission and view sensitive information

Securing Mobile Devices Considerations

1st Consideration: Configuring the device:

1. Disable unused feature: disable Bluetooth wireless data communication in order to prevent bluejacking and bluesnarfing.
2. Use strong authentication: Restrict unauthenticated users with a screen lock and require a strong passcode.

Securing Mobile Devices Considerations

2nd Consideration: Configuring device app security:

Significant loopholes in which mobile device data can be accessed through:

1. 1- Data-in-transit: carrier build surveillance capabilities into their networks, allows law enforcement agencies to collet data-in-transit, new mobile apps over the –the-top content
2. 2-Remote- data-at-Rest: Apple and google possess decryption key necessary to unlock data on their services. Courts routinely server order to apple and google to provide data stored on their servers, Users can choose to turn off backups to iCloud or Google servers.

Securing Mobile Devices Considerations

3rd Consideration: Using mobile management tools:

Storage segmentation: Separating business data from personal data

- Containerization: Separating storage into containers and managing each appropriately.
- Advantages : Helps companies avoid data ownership privacy issues and legal concern regarding a user's personal data, and allows companies to delete only business data when necessary without touching personal data.

Outlines

- Bluetooth Systems
- Near Field Communication Systems
- Radio Frequency Identification Systems
- WiFi Systems
- Mobile Systems
- Security Solutions

WiFi Protected Access (WPA)

- Could be implemented through firmware upgrades on wireless network interface cards.
- Implements the Temporal Key Integrity Protocol (TKIP).
  - Employs a per-packet key, meaning that it dynamically generates a new 128-bit key for each packet and thus prevents the types of attacks.
- Includes a Message Integrity Check (CRC), which is designed to prevent an attacker from altering and resending data packets (Integrity).
- Withstands Modification and replay attacks

Limitations of WPA

1. Older firmware and operating systems cannot be upgraded to support WPA.
2. WPA software must be present in computers, AP, and wireless adapters for WPA to function.
3. To connect multiple versions of WPA, the WPA and network clients must have the same configuration.
4. DoS and spoofing attacks can still happen even with WPA.
5. Transmission time is longer due to the additional packet size.

WPA 2

- More secure than WPA, because it uses a more advanced encryption key named "AES-CCMP."
- Helps wireless users in other ways, such as enabling them to disconnect from a wireless network and reconnect without having to go through a number of authentication process.
- Limitations
  1. Attackers have successfully compromised networks that use WPA and WPA2 encryption.
  2. Attackers can simply park near a location that broadcasts Wi-Fi and attempt to crack the network's security.
  3. WPA passwords are longer and more secure, but you should use WPA2 when you really need maximum protection.

WPA 3

- **In-transit Security**: Enhanced 128-bit encryption in WPA3-Personal and 192-bit encryption for WPA3-Enterprise implementations. Using higher bit encryption significantly decreases the odds of compromising the key.
- **Secure Authentication**: Uses a pre-shared key to join the network. Adds another layer of security or "handshake" called Simultaneous Authentication of Equals (SAE). The latest standard also introduces "forward secrecy", which protects the ongoing communication even if the pre-shared key used to authenticate is compromised.
- **Public Network Security**: Public networks such as in airports, malls and municipal networks are usually unencrypted or "Open." With WPA3, the communication over open network is automatically encrypted with a mechanism called Opportunistic Wireless Encryption (OWE). This prevents eavesdropping while connected on a public network.

Thanks

Security and Protection of Systems

Lecture 07
Introduction to Cryptography

Dr. Zaher Haddad

Outlines

- Introduction
- Cryptographic Transformations
- Stream and Block Cipher
- Feistel cipher
- Data Encryption Standard (DES)

Outlines

- Introduction
- Cryptographic Transformations
- Stream and Block Cipher
- Feistel cipher
- Data Encryption Standard (DES)

3

Terminologies

- Plaintext: An original message
- Ciphertext: The coded message
- Deciphering/decryption: Restoring the plaintext from the ciphertext
- Cryptography: The area of study of the many schemes used for encryption
- Cryptographic system/cipher: A scheme being use to convert from ciphertext to plaintext
- Cryptanalysis: Techniques used for deciphering a message without any knowledge of the enciphering details
- Cryptology: The areas of cryptography and cryptanalysis

4

Characteristics

1. The type of operations used for transforming plaintext to ciphertext
   - Substitution
   - Transposition
2. The number of keys used
   - Symmetric, single-key, secret-key, conventional encryption
   - Asymmetric, two-key, or public-key encryption
3. The way in which the plaintext is processed
   - Block cipher
   - Stream cipher

5

Encryption Scheme Security

- Unconditionally secure: No matter how much time an opponent has, it is impossible to decrypt the ciphertext simply because the required information is not there
- Computationally secure: The cost of breaking cipher exceeds the value of encrypted information, The time required to break the cipher exceeds the useful lifetime of information
- Diffusion: The statistical structure of plaintext is dissipated into long-range statistics of the ciphertext. This is achieved by having each plaintext digit affect the value of many ciphertext digits
- Confusion: Seeks to make the relationship between the statistics of the ciphertext and the value of the encryption key as complex as possible . Even if the attacker can get some handle on the statistics of the ciphertext, the way in which the key was used to produce that ciphertext is so complex as to make it difficult to deduce the key

6

Outlines

- Introduction
- Cryptographic Transformations
- Stream and Block Cipher
- Feistel cipher
- The Data Encryption Standard (DES)

7

Cryptographic Transformations

1- Substitutions: Each plaintext element or group of elements is uniquely replaced by a corresponding ciphertext element or group of elements

- Examples of Substitution techniques such as Caesar – Playfair – Vigenère.

2- Permutation (Transposition) : No elements are added or deleted or replaced in the sequence, rather the order in which the elements appear in the sequence is changed

- Examples of permutation techniques Such as Rail Fence Cipher, Row Transposition Cipher

8

1- Caesar Cipher

- Simplest and earliest known use of a substitution cipher used by Julius Caesar
- Replacing each letter of the alphabet with the letter standing three places further down the alphabet
- Mathematically: the general Caesar algorithm is:

$$C = E(k, p) = (p + k) \bmod 26 \quad , \quad P = D(k, C) = (C - k) \bmod 26$$

- Where k takes on a value in the range 1 to 25.
- Example: plain: meet me after the toga party

  cipher: PHHW PH DIWHU WKH WRJD SDUWB

9

2- Playfair Cipher

- A substitution cipher used replacing each couple of the plaintext by diagrams in 5x5 matrix as follows:

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

1. Create 5 x 5 matrix of letters using a keyword such as "MONARCHY"
2. Create a diagram by splitting plaintext to couple of letters

- In the case of the last letter is single, couple it by letter x.
- If a two letters are repeated, insert letter x in the middle.

10

2- Playfair Cipher

- If the letters were in different rows and columns, encrypts them by rectangle circular left/right swap
- If the letters were in the same rows, encrypts them by circular left swap.
- If the letters were in the same column, encrypts them by circular down swap.
- Reverse the last steps to decrypt the message

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

11

2- Playfair Cipher

- Example 1 : Plaintext: attack
- Diagram : at ta ck
- Repeating: no repeating
- at encrypted to RS
- Ta encrypted to SR
- Ck encrypted to DE.
- Ciphertext: RSSRDE

- Example 2 : Plaintext: balloon
- Diagram : ba ll oo n
- Repeating: yes: ba lx lo on
- ba same column encrypt to IB
- lx encrypted to SU
- lo encrypted to PM.
- on same row encrypted to NA
- Ciphertext: RSSRDE

12

3- Rail Fence Cipher

- Simplest transposition cipher
- Plaintext is written down as a sequence of diagonals and then read off as a sequence of rows
- To encipher message "meet me after the toga party" with a rail fence of depth 2, we would write:
- m e m a t r h t g p r y
- e t e f e t e o a a t
- Encrypted message is: MEMATRHTGPRYETEFETEOAAT

13

4- Row Transposition Cipher

- Is a more complex transposition, Write 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: 3 4 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

  Ciphertext: TTNAAPTMTSUOAODWCOIXKNLYPETZ

14

Outlines

- Introduction
- Cryptographic Transformations
- Stream and Block Cipher
- Feistel cipher
- The Data Encryption Standard (DES)

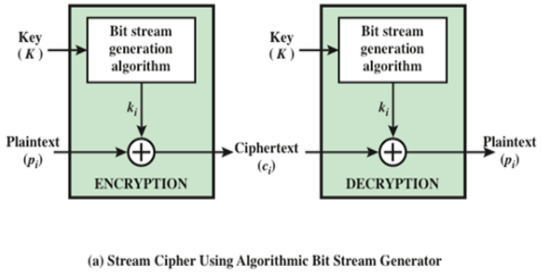
15

Stream Cipher

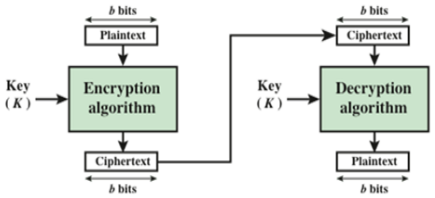
- Encrypts data stream one bit or one byte at a time
- In the ideal case, the keystream is as long as the plaintext bit stream
- Keystream must be provided to both users via some independent and secure channel; introduces insurmountable logistical problems if intended data traffic is very large
- bit-stream generator must be implemented as an algorithmic procedure so that the cryptographic bit stream can be produced by both users
- It must be computationally impractical to predict future portions of the bit stream based on previous portions of the bit stream
- The two users need only share the generating key and each can produce the keystream

Block Cipher

- A block of plaintext is treated as a whole and used to produce a ciphertext block of equal length
- Typically a block size of 64 or 128 bits is used
- As with a stream cipher, the two users share a symmetric encryption key
- The majority of network-based symmetric cryptographic applications make use of block ciphers



(a) Stream Cipher Using Algorithmic Bit Stream Generator



(b) Block Cipher

Block Cipher Design Principles

1- Number of Rounds: The greater number of rounds, the more difficult it is to perform cryptanalysis.

2- Design of Function F (Feistel): The more nonlinear F, the more difficult any type of cryptanalysis will be Strict avalanche criterion (SAC); O/P bits of an S-box should change with probability 1/2 when any single I/P bit is inverted and Bit independence criterion (BIC); O/P bits should change independently when any single I/P bit is inverted.

3- Key Schedule Algorithm: the key is used to generate one subkey for each round, In general, we would like to select subkeys to maximize the difficulty of deducing individual subkeys and the difficulty of working back to the main key, It is suggested that, at a minimum, the key schedule should guarantee key/ciphertext Strict Avalanche Criterion and Bit Independence Criterion

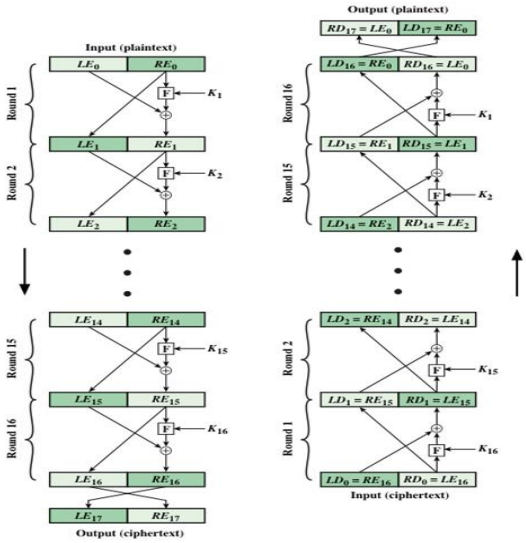
Outlines

- Introduction
- Cryptographic Transformations
- Stream and Block Cipher
- Feistel cipher
- The Data Encryption Standard (DES)

19

Feistel Cipher

- Feistel proposed the use of a cipher that alternates substitutions and permutations
- Is a practical application of a proposal by Claude Shannon to develop a product cipher that alternates confusion and diffusion functions
- Is the structure used by many significant symmetric block ciphers currently in use



Feistel Cipher Design Features

1. Block size: Larger block sizes mean greater security but reduced speed
2. Key size: Larger key size means greater security but may decrease speeds
3. Number of rounds: The essence of the Feistel cipher is that a single round offers inadequate security but that multiple rounds offer increasing security
4. Subkey generation algorithm: Greater complexity in this algorithm should lead to greater difficulty of cryptanalysis
5. Round function F: Greater complexity generally means greater resistance to cryptanalysis

Feistel Cipher Design Features

6. Fast software encryption/decryption: In many cases, encrypting is embedded in applications or utility functions, preclude a hardware implementation; accordingly, the speed of execution of the algorithm becomes a concern
7. Ease of analysis :If the algorithm can be concisely and clearly explained, it is easier to analyze that algorithm for cryptanalytic vulnerabilities and therefore develop a higher level of assurance as to its strength

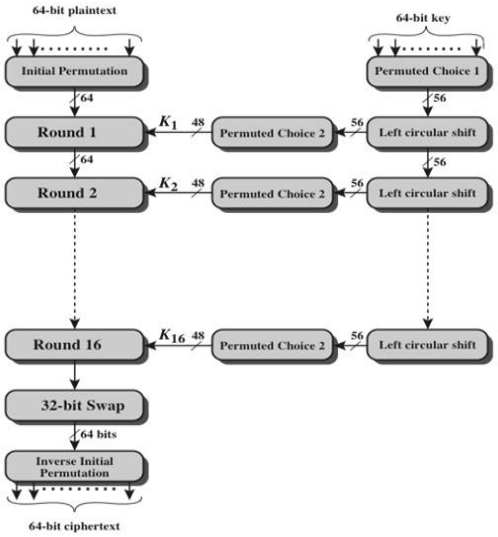
Outlines

- Introduction
- Cryptographic Transformations
- Stream and Block Cipher
- Feistel cipher
- The Data Encryption Standard (DES)

23

Data Encryption Standard (DES)

- Issued in 1977 by NIST as Federal Information Processing Standard
- The most widely used encryption scheme until introduction of AES in 2001
- Data are encrypted in 64-bit blocks using a 56-bit key
- The algorithm transforms 64-bit input in a series of steps into a 64-bit output
- The same steps, with the same key, are used to reverse the encryption



DES Example

Round	*Ki*	*Li*	*Ri*
IP		5a005a00	3cf03c0f
1	1e030f03080d2930	3cf03c0f	bad22845
2	0a31293432242318	bad22845	99e9b723
3	23072318201d0c1d	99e9b723	0bae3b9e
4	05261d3824311a20	0bae3b9e	42415649
5	3325340136002c25	42415649	18b3fa41
6	123a2d0d04262a1c	18b3fa41	9616fe23
7	021f120b1c130611	9616fe23	67117cf2
8	1c10372a2832002b	67117cf2	c11bfc09
9	04292a380c341f03	c11bfc09	887fbc6c
10	2703212607280403	887fbc6c	600f7e8b
11	2826390c31261504	600f7e8b	f596506e
12	12071c241a0a0f08	f596506e	738538b8
13	300935393c0d100b	738538b8	c6a62c4e
14	311e09231321182a	c6a62c4e	56b0bd75
15	283d3e0227072528	56b0bd75	75e8fd8f
16	2921080b13143025	75e8fd8f	25896490
IP–1		da02ce3a	89ecac3b

Avalanche Effect in DES: Change in Plaintext

Round		δ
	02468aceeca86420 12468aceeca86420	1
1	3cf03c0fbad22845 3cf03c0fbad32845	1
2	bad2284599e9b723 bad3284539a9b7a3	5
3	99e9b7230bae3b9e 39a9b7a3171cb8b3	18
4	0bae3b9e42415649 171cb8b3ccaca55e	34
5	4241564918b3fa41 ccaca55ed16c3653	37
6	18b3fa419616fe23 d16c3653cf402c68	33
7	9616fe2367117cf2 cf402c682b2cefbc	32
8	67117cf2c11bfc09 2b2cefbc99f91153	33

Round		δ
9	c11bfc09887fbc6c 99f911532eed7d94	32
10	887fbc6c600f7e8b 2eed7d94d0f23094	34
11	600f7e8bf596506e d0f23094455da9c4	37
12	f596506e738538b8 455da9c47f6e3cf3	31
13	738538b8c6a62c4e 7f6e3cf34bc1a8d9	29
14	c6a62c4e56b0bd75 4bc1a8d91e07d409	33
15	56b0bd7575e8fd8f 1e07d4091ce2e6dc	31
16	75e8fd8f25896490 1ce2e6dc365e5f59	32
IP–1	da02ce3a89ecac3b 057cde97d7683f2a	32

Avalanche Effect in DES: Change in Key

Round		δ
	02468aceeca86420 02468aceeca86420	0
1	3cf03c0fbad22845 3cf03c0f9ad628c5	3
2	bad2284599e9b723 9ad628c59939136b	11
3	99e9b7230bae3b9e 9939136b768067b7	25
4	0bae3b9e42415649 768067b75a8807c5	29
5	4241564918b3fa41 5a8807c5488dbe94	26
6	18b3fa419616fe23 488dbe94aba7fe53	26
7	9616fe2367117cf2 aba7fe53177d21e4	27
8	67117cf2c11bfc09 177d21e4548f1de4	32

Round		δ
9	c11bfc09887fbc6c 548f1de471f64dfd	34
10	887fbc6c600f7e8b 71f64dfd4279876c	36
11	600f7e8bf596506e 4279876c399fdc0d	32
12	f596506e738538b8 399fdc0d6d208dbb	28
13	738538b8c6a62c4e 6d208dbbb9bdeeaa	33
14	c6a62c4e56b0bd75 b9bdeeaad2c3a56f	30
15	56b0bd7575e8fd8f d2c3a56f2765c1fb	33
16	75e8fd8f25896490 2765c1fb01263dc4	30
IP–1	da02ce3a89ecac3b ee92b50606b62b0b	30

Thanks

28

Security and Protection of Systems

Lecture 08
Symmetric Cryptography

Dr. Zaher Haddad

Outlines

- Advanced Encryption Standard
- Double and Trible DES
- Modes of Operation

Outlines

- Advanced Encryption Standard
- Double and Trible DES
- Modes of Operation

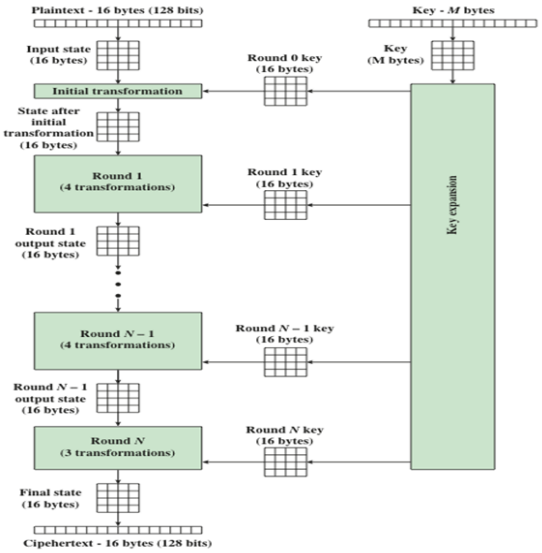
Advanced Encryption Standard (AES)

- The **AES**, also known by its original name **Rijndael** is a specification for the encryption of electronic data established by the U.S. (NIST) in 2001.
- Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.
- The algorithm described by AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data.

Key Size (words/bytes/bits)	4/16/128	6/24/192	8/32/256
Plaintext Block Size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Number of Rounds	10	12	14
Round Key Size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Expanded Key Size (words/bytes)	44/176	52/208	60/240

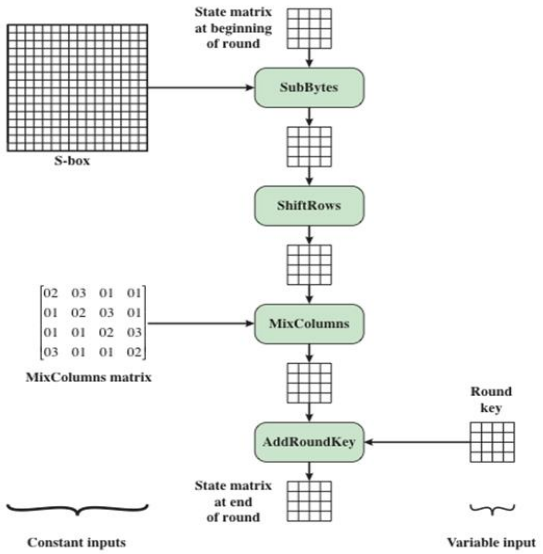
The overall structure of the AES encryption process.

- First N - 1 rounds consist of four transformation functions: SubBytes, ShiftRows, MixColumns, and AddRoundKey.
- Final round contains only three transformations, and a initial single transformation (AddRoundKey) before the first round
- Each transformation takes 4*4 matrices as input and produces a 4*4 matrix as output.
- The key expansion function generates N+1 round keys, each of which is a distinct 4* 4 matrix.



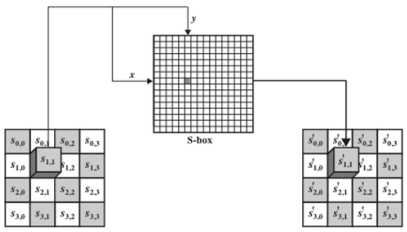
Single Round of AES

1. SubBytes – a non-linear substitution step where each byte is replaced with another according to a lookup table.
2. ShiftRows – a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
3. MixColumns – a linear mixing operation which operates on the columns of the state, combining the four bytes in each column.
4. AddRoundKey – each byte of the state is combined with a byte of the round key using bitwise xor.



AES Substitution

- Each byte in the *state* array is replaced with a SubByte sing an 8-bit substitution box.
- This operation provides the non-linearity in the cipher.
- The S-box used is derived from the multiplicative inverse over finite filed known to have good non-linearity properties.



- To avoid attacks based on simple algebraic properties, the S-box is constructed by combining the inverse function with an invertible affine transformation.
- The S-box is also chosen to avoid any fixed points and also any opposite fixed points.
- While performing the decryption, the InvSubBytes step (the inverse of SubBytes) is used, which requires first taking the inverse of the affine transformation and then finding the multiplicative inverse.

AES S-box

x \ y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
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

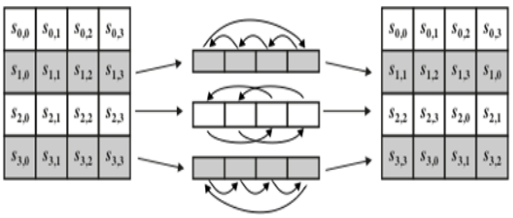
For example, the hexadecimal value {95} references row 9, column 5 of the S-box, which contains the value {2A}. Accordingly, the value {95} is mapped into the value {2A}.

AES Inverse S-box

		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

Shift Row Rationale

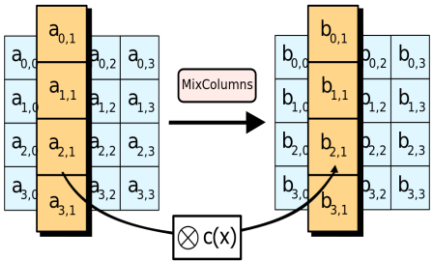
- Operates on the rows of the state; it cyclically shifts the bytes in each row by a certain offset.
- The first row is left unchanged.
- Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively.



- In this way, each column of the output state of the ShiftRows step is composed of bytes from each column of the input state.
- The importance of this step is to avoid the columns being encrypted independently, in which case AES would degenerate into four independent block ciphers.

Mix Columns Rationale

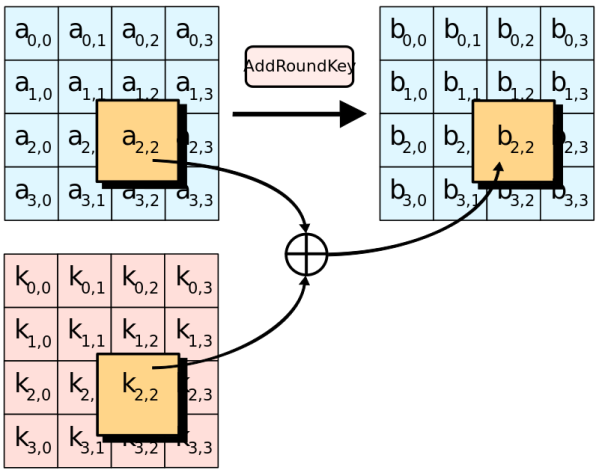
- In the MixColumns step, the four bytes of each column of the state are combined using an invertible linear transformation.
- The MixColumns function takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes.
- Together with ShiftRows, MixColumns provides diffusion in cipher



- During this operation, each column is transformed using a fixed matrix (matrix left-multiplied by column gives new value of column in the state)

AddRoundKey Transformation

- In the AddRoundKey step, the subkey is combined with the state. For each round, a subkey is derived from the main key using Rijndael's key schedule; each subkey is the same size as the state. The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR.



AES Security

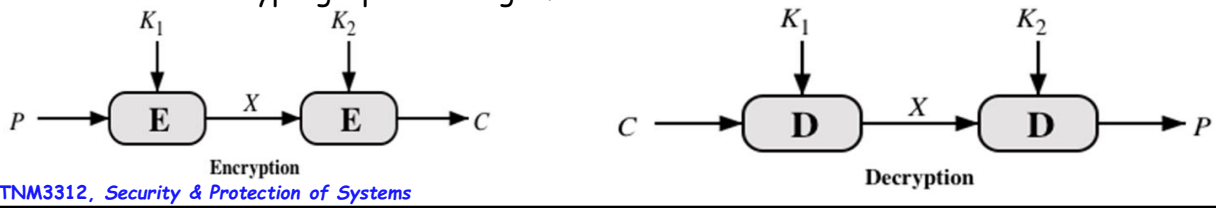
1. The design and strength of all key lengths of the AES algorithm (i.e., 128, 192 and 256) are sufficient to protect classified information up to the SECRET level.
2. TOP SECRET information will require use of either the 192 or 256 key lengths. The implementation of AES in products intended to protect national security systems and/or information must be reviewed and certified by NSA prior to their acquisition and use
3. AES has 10 rounds for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys.

Outlines

- Advanced Encryption Standard
- Double and Trible DES
- Modes of Operation

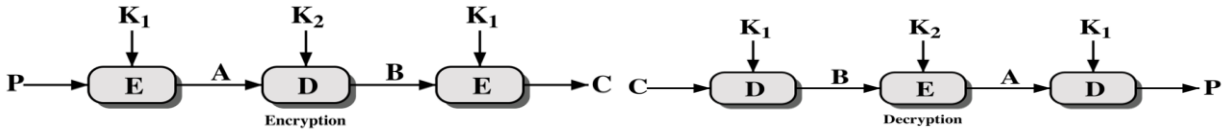
Double DES

- Because of its vulnerability to brute-force attack, symmetric cipher, has been largely replaced by stronger encryption schemes.
- Use multiple encryption with DES and multiple keys.
- The simplest form of multiple encryption has two encryption stages and two keys
- Given a plaintext P and two encryption keys K_1 and K_2 , ciphertext C is generated as $C = E(K_2 , E(K_1 , P))$, $P = D(K_1 , D(K_2 , C))$
- Double DES involves a key length of 56 * 2 = 112 bits, resulting in a dramatic increase in cryptographic strength.



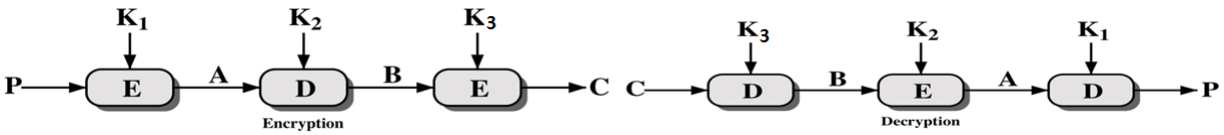
Triple-DES with Two-Keys

- Three stages of encryption with three different keys.
- This raises the cost of the meet-in-the-middle attack to 2^{112} , which is beyond what is practical now and far into the future.
  - $C = E(K_1, D(K_2, E(K_1, P)))$
  - $P = D(K_1, E(K_2, D(K_1, P)))$
- However, it has the drawback of requiring a key length of 56 * 3 = 168 bits, which may be somewhat unwieldy.



Triple DES with Three Keys

- Although the attacks just described appear impractical, anyone using two-key 3DES may feel some concern. Thus, many researchers now feel that three-key 3DES is the preferred alternative (e.g., [KALI96a]). Three-key 3DES has an effective key length of 168 bits and is defined as
  - $C = E(K_3, D(K_2, E(K_1, P)))$
  - $P = D(K_3, E(K_2, D(K_1, P)))$



Outlines

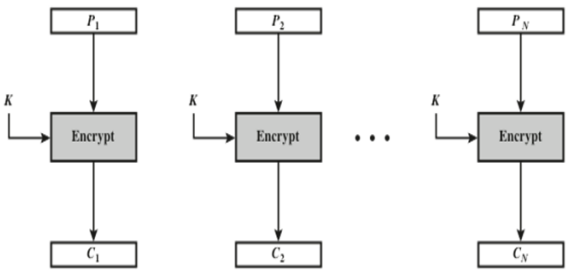
- Advanced Encryption Standard
- Double and Trible DES
- Modes of Operation

Modes of Operation

- A block cipher takes a fixed-length block b-bit and a key as input and produces a b-bit ciphertext block.
- If the amount of plaintext to be encrypted is greater than b-bit, then the block cipher can still be used by breaking the plaintext up into b-bit blocks.
- When n blocks are encrypted using the same key, a number of security issues arise.
- Mode of operation is a technique for enhancing the effect or adapting of a cryptographic algorithm, such as applying a block cipher to a sequence of data blocks or a data stream.
- Modes are intended to cover a wide of applications for which a block cipher could be used.
- These modes are intended for use with any symmetric block cipher (triple DES and AES).

1- Electronic Codebook (ECB) mode

- Plaintext is handled one block at a time and each block is encrypted using the same key.
- The term codebook is used because, for a given key, there is a unique ciphertext for every b-bit block of plaintext.

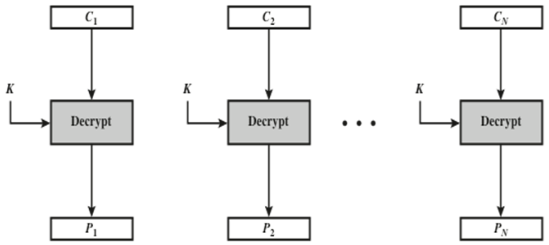


(a) Encryption

- Therefore, we can imagine a gigantic codebook in which there is an entry for every possible b-bit plaintext pattern showing its corresponding ciphertext.
- For a message longer than b-bit, the procedure is simply to break the message into b -bit blocks, padding the last block if necessary.

1- Electronic Codebook (ECB) mode

- Decryption is performed one block at a time, always using the same key.
- ECB method is ideal for a short amount of data, such as an encryption key , Thus, if you want to transmit a DES or AES key securely, ECB is the appropriate mode to use.

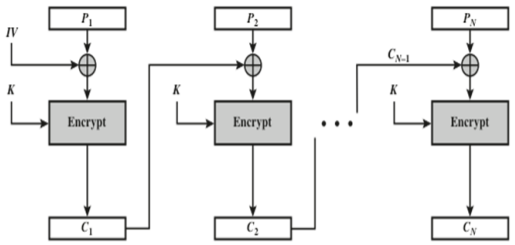


(b) Decryption

- The most significant characteristic of ECB is that if the same b -bit block of plaintext appears more than once in the message, it always produces the same ciphertext.
- For lengthy messages, the ECB mode may not be secure. If the message is highly structured, it may be possible for a cryptanalyst to exploit these regularities.

2- Cipher block chaining (CBC) mode

- CBC overcomes the security deficiencies of ECB, if the same plaintext block repeated, produces different ciphertext blocks.
- In CBC, the input to the encryption algorithm is the XOR of the current plaintext block and the preceding ciphertext block; the same key is used for each block.

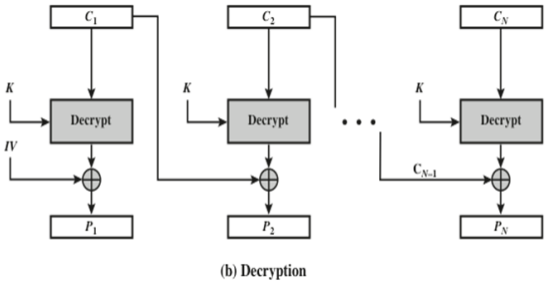


(a) Encryption

- In effect, we have chained together the processing of the sequence of plaintext blocks.
- The input to the encryption function for each plaintext block bears no fixed relationship to the plaintext block. , Therefore, repeating patterns of b bits are not exposed. As with the ECB mode, the CBC mode requires that the last block be padded to a full b bits if it is a partial block.

2- Cipher block chaining (CBC) mode

- On decryption, the IV is XORed with the output of the decryption algorithm to recover the first block of plaintext.
- IV is a data block that is the same size as the cipher block, the IV must be known to both the sender and receiver but be unpredictable by a third party.

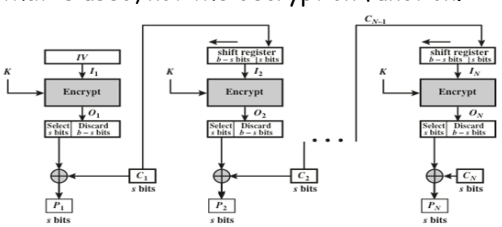


(b) Decryption

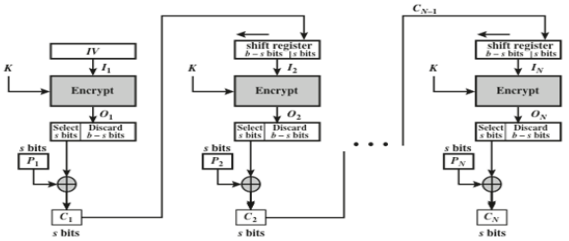
- In particular, for any given plaintext, it must not be possible to predict the IV that will be associated to the plaintext in advance of the generation of the IV.
- For maximum security, the IV should be protected against unauthorized changes.
- To produce the first block of ciphertext, an initialization vector (IV) is XORed with the first block of plaintext.

3- Cipher feedback (CFB) Mode

- The input to the encryption function is a b-bit shift register that is initially set to some IV.
- The leftmost s bits of the output of the encryption function are XORed with the first segment of plaintext P_1 to produce the first unit of ciphertext C_1 , which is then transmitted.
- The contents of the shift register are shifted left by s bits, and C_1 is placed in the rightmost s bits of the shift register.
- For decryption, the same scheme is used, except that the received ciphertext unit is XORed with the output of the encryption function to produce the plaintext unit. Note that it is the encryption function that is used, not the decryption function.



(b) Decryption



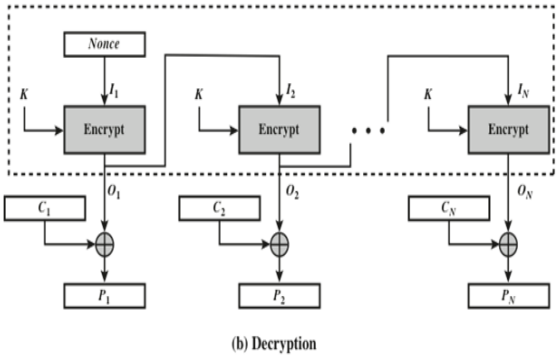
(a) Encryption

3- Cipher feedback (CFB) Mode

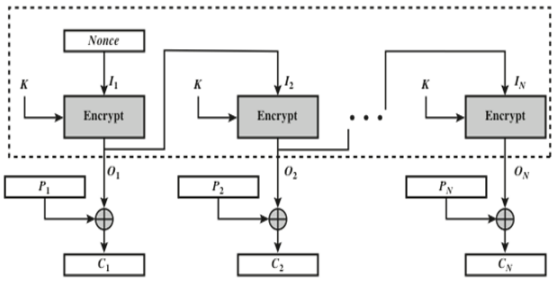
- CFB can be viewed as a stream cipher, it does not conform to the typical construction of a stream cipher.
- In a typical stream cipher, the cipher takes as input some initial value and a key and generates a stream of bits, which is then XORed with the plaintext bits. In the case of CFB, the stream of bits that is XORed with the plaintext also depends on the plaintext.
- In CFB encryption, like CBC encryption, the input block to each forward Cipher function (except the first) depends on the result of the previous forward Cipher function; therefore, multiple forward cipher operations cannot be performed in parallel.
- In CFB decryption, the required forward cipher operations can be performed in parallel if the input blocks are first constructed (in series) from the IV and the ciphertext.

4- Output Feedback (OFB) mode

- For OFB, the output of encryption is fed back to become input for encrypting the next block.
- In CFB, the output of XOR unit is fed back to become input for encrypting the next block.
- The other difference is that the OFB mode operates on full blocks of plaintext and ciphertext, whereas CFB operates on an s -bit subset.



(b) Decryption



(a) Encryption

4- Output Feedback (OFB) mode

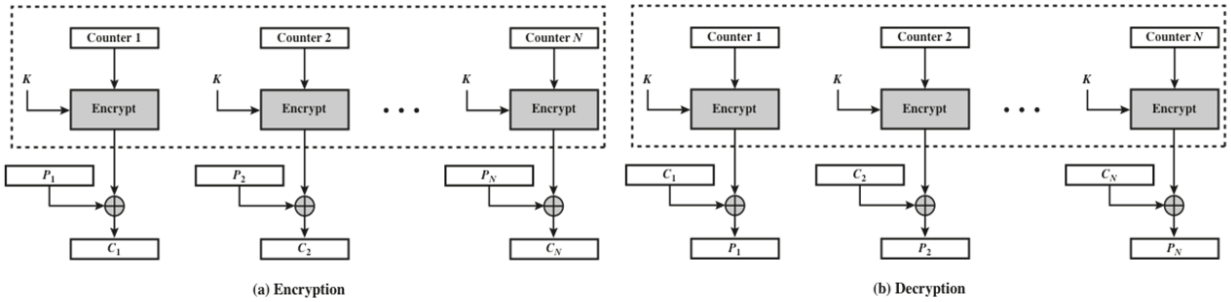
- IV must be a nonce; that is, the IV must be unique to each execution of the encryption operation. The reason for this is that the sequence of encryption output blocks, O_i, depends only on the key and the IV and does not depend on the plaintext.
- Therefore, for a given key and IV, the stream of output bits used to XOR with the stream of plaintext bits is fixed. If two different messages had an identical block of plaintext in the identical position, then an attacker would be able to determine that portion of the O_i stream.
- One advantage of the OFB method is that bit errors in transmission do not propagate.
- The disadvantage of OFB is that it is more vulnerable to a message stream modification attack than is CFB.

counter (CTR) mode

- Although interest in the CTR mode has increased recently with applications to ATM network security and IP sec.
- A counter equal to the plaintext block size is used.
- The only requirement is that the counter value must be different for each plaintext block that is encrypted.
- Typically, the counter is initialized to some value and then incremented by 1 for each subsequent block (modulo 2^b, where b is the block size).

counter (CTR) mode

- For encryption, the counter is encrypted and then XORed with the plaintext block to produce the ciphertext block; there is no chaining.
- For decryption, the same sequence of counter values is used, with each encrypted counter XORed with a ciphertext block to recover the corresponding plaintext block.
- Thus, the initial counter value must be made available for decryption.



Mode	Description	Typical Application
Electronic Codebook (ECB)	Each block of 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 block of plaintext and the preceding block 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 encryption output, and full blocks are used.	•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

Thanks

Security and Protection of Systems

Lecture 09
Asymmetric Cryptography

Dr. Zaher Haddad

Outlines

- Introduction
- RSA Algorithm
- Elgamal Algorithm
- Diffie-Hellman Algorithm

Outlines

- Introduction
- RSA Algorithm
- Elgamal Algorithm
- Diffie-Hellman Algorithm

Terminology

- **Asymmetric Keys:** Two related keys, a public(PK) and private (sk) , used to perform complementary operations, such as encryption, decryption or signature generation/verification .
- **Public key certificate:** a digital document issued and digitally signed by the sk of certification authority that binds the name of subscriber to a PK. The certificate indicates that the subscriber identified in the certificate has sole control and access the corresponding private key.
- **Public key(asymmetric) cryptographic algorithm:** a algorithm that uses two related keys. The two keys have the property that deriving sk from the PK is computationally infeasible.
- **Public key Infrastructure:** a set of policies, processes, server platforms, software and workstations used for the purpose of administration certificate sk, PK pairs, including the ability to issue, maintain, and revoke PK certificates.

Misconceptions Concerning Public-Key Encryption

Misconceptions Concerning Public-Key Encryption

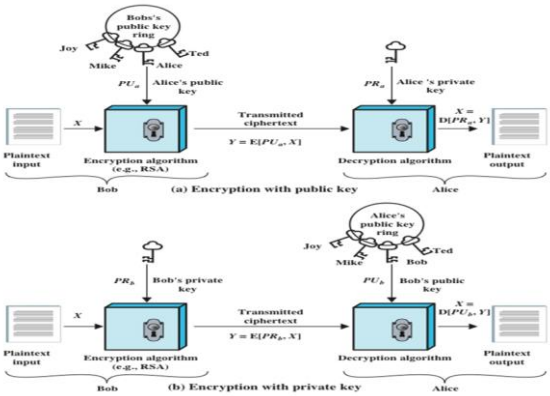
1. Public-key encryption is more secure than symmetric encryption
2. PKI is a general-purpose technique that has made symmetric encryption obsolete
3. There is a feeling that key distribution is trivial when using PKI, compared to the cumbersome handshaking involved with key distribution centers for symmetric encryption

- **Principles of Public-Key Cryptosystems :** evolved from an attempt to attack two of the most difficult problems associated with symmetric encryption:

1. Key distribution: How to have secure communications in general without having to trust a key distribution center (KDC) with your key
2. Digital signatures: How to verify that a message comes intact from the claimed sender

Public-Key Encryption/Decryption

1. Each user generates a pair of keys to be used for the encryption/decryption.
2. Each user places one of the two keys in a public register or accessible file (PK), the companion key is kept private (sk).
3. Each user maintains a collection of PKs obtained from others.
4. Bob sends Alice a confidential message by encrypting the message using Alice's PK.



5. Alice decrypts the message using her sk. No other recipient can decrypt the message because only Alice knows Alice's sk.
6. All participants have access to PK, and private keys are generated locally by each participant and therefore need never be distributed.

Outlines

- Introduction
- RSA Algorithm
- Elgamal Algorithm
- Diffie-Hellman Algorithm

Rivest-Shamir-Adleman (RSA) Algorithm

- by Ron Rivest, Adi Shamir, and Len Adleman at MIT and first published in 1978
- The Rivest-Shamir-Adleman (RSA) scheme has since that time reigned supreme as the most widely accepted and implemented general-purpose approach to public-key encryption.
- The RSA scheme is a cipher in which the plaintext and ciphertext are integers between 0 and n - 1 for some n .
- A typical size for n is 1024 bits, or 309 decimal digits.
- That is the probability of n is less than 2^{1024} .

RSA Algorithm

- Plaintext is encrypted in blocks, with each block having a binary value less than some number n .
- Block size must be less than or equal to $\log_2 (n) + 1$; in practice, the block size is i bits, where $2^i < n \leq 2^{i+1}$.
- Encryption / decryption are of the following form, for some plaintext block M and ciphertext block C.

$$C = M^e \mod n$$

$$M = C^d \mod n = (M^e)^d \mod n = M^{ed} \mod n$$

- Both sender and receiver must know the value of n . The sender knows the value of e , and only the receiver knows the value of d .
- Thus, this is a public-key encryption algorithm with a public key of $PU = \{e , n\}$ and a private key of $PR = \{d , n\}$.

Example of RSA Algorithm

Key Generation by Alice

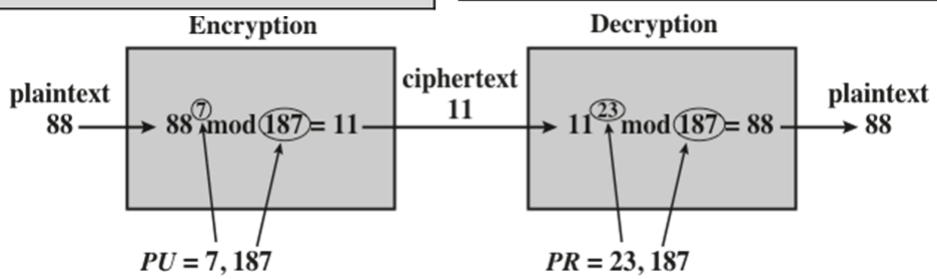
Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$d = e^{-1} \pmod{\phi(n)}$
Public key	$PU = \{e, n\}$
Private key	$PR = \{d, n\}$

Encryption by Bob with Alice's Public Key

Plaintext:	$M < n$
Ciphertext:	$C = M^e \bmod n$

Decryption by Alice with Alice's Private Key

Ciphertext:	C
Plaintext:	$M = C^d \bmod n$



Algorithm Requirements

- For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

1. It is possible to find values of e, d, n such that $M^{ed} \bmod n = M$ for all $M < n$
2. It is relatively easy to calculate $M^e \bmod n$ and $C^d \bmod n$ for all values of $M < n$
3. It is infeasible to determine d given e and n
   - *Why? Because of Factorization Problem*

The Security of RSA

1. Brute force: Involves trying all possible private keys
2. Mathematical attacks : There are several approaches, all equivalent in effort to factoring the product of two primes
3. Timing attacks: These depend on the running time of the decryption algorithm
4. Hardware fault-based attack: This involves inducing hardware faults in the processor that is generating digital signatures
5. Chosen ciphertext attacks: This type of attack exploits properties of the RSA algorithm

Outlines

- Introduction
- RSA Algorithm
- Elgamal Algorithm
- Diffie-Hellman Algorithm

ElGamal Cryptography

- Announced in 1984 by T. Elgamal
- Public-key scheme based on discrete logarithms closely related to the Diffie-Hellman technique
- Used in the digital signature standard (DSS) and the S/MIME e-mail standard
- Global elements are a prime number q and a which is a primitive root of q
- Security is based on the difficulty of computing discrete logarithms

ElGamal Cryptography

Global Public Elements	
q	prime number
α	$\alpha < q$ and α a primitive root of q

Key Generation by Alice	
Select private X_A	$X_A < q - 1$
Calculate Y_A	$Y_A = \alpha^{X_A} \bmod q$
Public key	$\{q, \alpha, Y_A\}$
Private key	X_A

ElGamal Cryptography

Encryption by Bob with Alice's Public Key	
Plaintext:	$M < q$
Select random integer k	$k < q$
Calculate K	$K = (Y_A)^k \bmod q$
Calculate C_1	$C_1 = \alpha^k \bmod q$
Calculate C_2	$C_2 = KM \bmod q$
Ciphertext:	(C_1, C_2)

Decryption by Alice with Alice's Private Key	
Ciphertext:	(C_1, C_2)
Calculate K	$K = (C_1)^{X_A} \bmod q$
Plaintext:	$M = (C_2 K^{-1}) \bmod q$

Outlines

- Introduction
- RSA Algorithm
- Elgamal Algorithm
- Diffie-Hellman Algorithm

Elliptic Curve Arithmetic

- Most of the products and standards that use public-key cryptography for encryption and digital signatures use RSA
  - The key length for secure RSA use has increased over recent years and this has put a heavier processing load on applications using RSA
- Elliptic curve cryptography (ECC) is showing up in standardization efforts including the IEEE P1363 Standard for Public-Key Cryptography
- Principal attraction of ECC is that it appears to offer equal security for a far smaller key size

Elliptic Curves Over GF(2^m)

- Use a cubic equation in which the variables and coefficients all take on values in GF(2^m) for some number m
- Calculations are performed using the rules of arithmetic in GF(2^m)
- The form of cubic equation appropriate for cryptographic applications for elliptic curves is somewhat different for GF(2^m) than for Z_p
  - It is understood that the variables x and y and the coefficients a and b are elements of GF(2^m) and that calculations are performed in GF(2^m)

Elliptic Curve Cryptography (ECC)

- Addition operation in ECC is the counterpart of modular multiplication in RSA
- Multiple addition is the counterpart of modular exponentiation
- To form a cryptographic system using elliptic curves, we need to find a "hard problem" corresponding to factoring the product of two primes or taking the discrete logarithm
  - $Q = kP$, where Q, P belong to a prime curve
  - Is "easy" to compute Q given k and P
  - But "hard" to find k given Q, and P
  - Known as the elliptic curve logarithm problem

Global Public Elements	
$E_q(a, b)$	elliptic curve with parameters a, b, and q, where q is a prime or an integer of the form 2^m
G	point on elliptic curve whose order is large value n

User A Key Generation	
Select private n_A	$n_A < n$
Calculate public P_A	$P_A = n_A \times G$

User B Key Generation	
Select private n_B	$n_B < n$
Calculate public P_B	$P_B = n_B \times G$

Calculation of Secret Key by User A
$K = n_A \times P_B$

Calculation of Secret Key by User B
$K = n_B \times P_A$

ECC Encryption/Decryption

- Must first encode any message m as a point on the elliptic curve P_m
- Select suitable curve and point G as in Diffie-Hellman
- Each user chooses a private key n_A and generates a public key $P_A = n_A * G$
- To encrypt and send message P_m to B, A chooses a random positive integer k and produces the ciphertext C_m consisting of the pair of points:

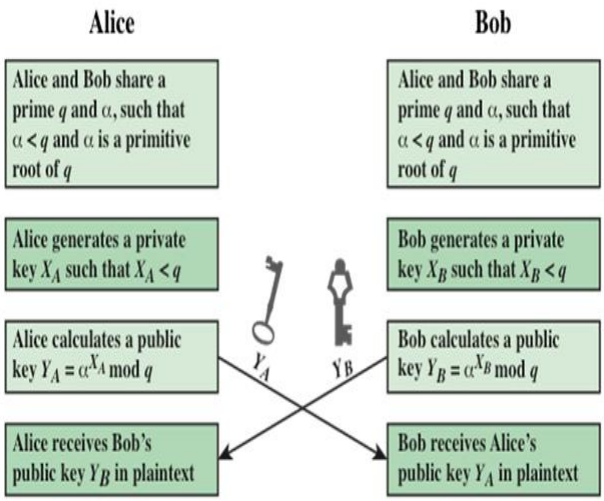
$$C_m = \{kG, P_m + kP_B\}$$

- To decrypt the ciphertext, B multiplies the first point in the pair by B's secret key and subtracts the result from the second point:

$$P_m + kP_B - n_B(kG) = P_m + k(n_BG) - n_B(kG) = P_m$$

Diffie-Hellman Key Exchange

- Purpose is to enable two users to securely exchange a key that can then be used for subsequent symmetric encryption of messages
- The algorithm itself is limited to the exchange of secret values
- Its effectiveness depends on the difficulty of computing discrete logarithms



23

Security of Elliptic Curve Cryptography

- Depends on the difficulty of the elliptic curve logarithm problem
- Fastest known technique is "Pollard rho method"
- Compared to factoring, can use much smaller key sizes than with RSA
- For equivalent key lengths computations are roughly equivalent
- Hence, for similar security ECC offers significant computational advantages

Symmetric and Asymmetric cryptography

Conventional Encryption	Public-Key Encryption
Needed to Work: 1. The same algorithm with the same key is used for encryption and decryption. 2. The sender and receiver must share the algorithm and the key. *Needed for Security:* 1. The key must be kept secret. 2. It must be impossible or at least impractical to decipher a message if the key is kept secret. 3. Knowledge of the algorithm plus samples of ciphertext must be insufficient to determine the key.	*Needed to Work:* 1. One algorithm is used for encryption and a related algorithm for decryption with a pair of keys, one for encryption and one for decryption. 2. The sender and receiver must each have one of the matched pair of keys (not the same one). *Needed for Security:* 1. One of the two keys must be kept secret. 2. It must be impossible or at least impractical to decipher a message if one of the keys is kept secret. 3. Knowledge of the algorithm plus one of the keys plus samples of ciphertext must be insufficient to determine the other key.

Comparable Key Sizes in Terms of Computational Effort for Cryptanalysis

Symmetric key algorithms	Diffie-Hellman, Digital Signature Algorithm	RSA (size of n in bits)	ECC (modulus size in bits)
80	$L = 1024$ $N = 160$	1024	160–223
112	$L = 2048$ $N = 224$	2048	224–255
128	$L = 3072$ $N = 256$	3072	256–383
192	$L = 7680$ $N = 384$	7680	384–511
256	$L = 15,360$ $N = 512$	15,360	512+

L = size of PK
N = size of sk

Algorithm	Encryption/Decryption	Digital Signature	Key Exchange
RSA	Yes	Yes	Yes
Elliptic Curve	Yes	Yes	Yes
Diffie-Hellman	No	No	Yes
DSS	No	Yes	No

Applications for PKI

Thanks