

# BLOCK CIPHERS, DES AND AES

TTM4135 - Lecture 4

Tjerand Silde

15.01.2026

# Motivation

- ▶ Block ciphers are the main bulk encryption algorithms used in commercial and military applications. They are considered both secure and efficient.
- ▶ Standardized block cipher AES and legacy cipher DES are widely deployed in real-world applications such as TLS and secure messaging.
- ▶ NIST's [AES algorithm validation list](#) includes over 13,500 validated implementations for e.g. USB drives, door controllers, media server encryption, disk encryption, Bluetooth devices, iPhone and more.

# Contents

## Block Cipher Principles

- Product Ciphers and Iterated Ciphers

- Feistel Ciphers

- Substitution-permutation networks

- Standard security properties

## DES

- History of DES

- DES algorithm

- Brute Force Attack on DES

- Double and triple DES

## AES

- AES History

- AES Algorithm

- Comparison of AES and DES

# Block ciphers

- ▶ Block ciphers are symmetric key ciphers in which each block of plaintext is encrypted with the same key.
- ▶ A *block* is a set of plaintext symbols of a fixed size. Typical block sizes for modern block ciphers are between 64 and 256 bits.
- ▶ In practice block ciphers are used in certain configurations called *modes of operation*. We look at modes in a later lecture.

# Notation in this lecture

- ▶  $P$ : Plaintext block (length  $n$  bits)
- ▶  $C$ : Ciphertext block (length  $n$  bits)
- ▶  $K$ : Key (length  $k$  bits)
- ▶  $C = E(P, K)$ : Encryption function
- ▶  $P = D(C, K)$ : Decryption function

# Criteria for block cipher design

In the 1940s Claude Shannon discussed two important encryption techniques.

- ▶ **Confusion:** This involves substitution to make the relationship between the key and ciphertext as complex as possible.
- ▶ **Diffusion:** This involves transformations that dissipate the statistical properties of the plaintext across the ciphertext.

Shannon proposed to use these techniques repeatedly using the concept of *product cipher*.

# Product cipher

- ▶ A product cipher is a cryptosystem in which the encryption function is formed by applying (or *composing*) several sub-encryption functions.
- ▶ Most block ciphers are the composition of simple functions  $f_i$  for  $i = 1, \dots, r$  where each  $f_i$  has a different key  $K_i$ .
- ▶ Thus we can write

$$C = E(P, K) = f_r(\dots (f_2(f_1(P, K_1), K_2) \dots), K_r)$$

# Iterated ciphers

Most modern block ciphers are in a special class of product ciphers known as *iterated ciphers*.

- ▶ The encryption process is divided into  $r$  similar *rounds*
- ▶ The sub-encryption functions are all the same function  $g$  called the *round function*
- ▶ Each key  $K_i$  is derived from the overall master key  $K$ . The keys  $K_i$  are called *round keys* or *subkeys* and are derived from  $K$  using a process called the *key schedule*

# Encryption in iterated ciphers

Given a plaintext block,  $P$ , a round function  $g$  and round keys  $K_1, K_2, \dots, K_r$ , the ciphertext block,  $C$ , is derived through  $r$  rounds as follows:

$$\begin{aligned}W_0 &= P \\W_1 &= g(W_0, K_1) \\W_2 &= g(W_1, K_2) \\&\vdots \\W_r &= g(W_{r-1}, K_r) \\C &= W_r\end{aligned}$$

## Decrypting iterated ciphers

- ▶ The round function  $g$  must have an inverse function  $g^{-1}$  with  $g^{-1}(g(W, K_i), K_i) = W$  for all keys  $K_i$  and blocks  $W$ .
- ▶ Decryption is then the reverse of encryption:

$$\begin{aligned}W_r &= C \\W_{r-1} &= g^{-1}(W_r, K_r) \\W_{r-2} &= g^{-1}(W_{r-1}, K_{r-1}) \\&\vdots \\W_0 &= g^{-1}(W_1, K_1) \\P &= W_0\end{aligned}$$

# Types of iterated cipher

Two widely used general block cipher designs are:

1. **Feistel ciphers:** an example is the Data Encryption Standard (DES)
2. **Substitution-Permutation Networks (SPNs):** an example is the Advanced Encryption Standard (AES)

# Feistel cipher

- ▶ Named after Horst Feistel, a cryptographer working for IBM who influenced the design of the Data Encryption Standard (DES)
- ▶ A Feistel cipher is an iterated cipher in which the round function swaps the two halves of the block and forms a new right hand half
- ▶ The process is sometimes called a *Feistel network* since it can be seen as a network which the two halves of the plaintext block travel through

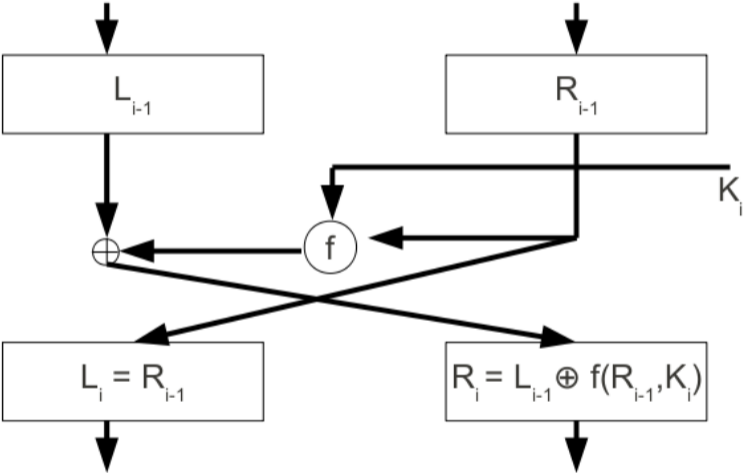
# Feistel encryption

1. Split plaintext block  $P = W_0$  into two halves:  $W_0 = (L_0, R_0)$ .
2. For each of the  $r$  rounds perform the following:

$$\begin{aligned}L_i &= R_{i-1} \\R_i &= L_{i-1} \oplus f(R_{i-1}, K_i)\end{aligned}$$

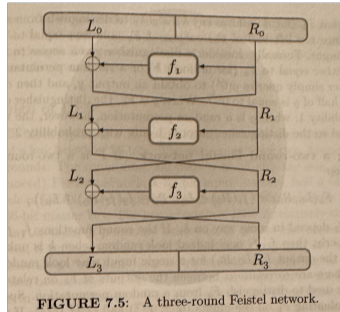
3. Ciphertext  $C = W_r$  is defined by  $C = (L_r, R_r)$ .

# Feistel ladder diagram



# Feistel ladder diagram

Source: Introduction to Modern Cryptography, Jonathan Katz and Yehuda Lindell, third edition



# Feistel decryption

1. Write the ciphertext block  $C$  as  $C = (L_r, R_r)$ .
2. For each of the  $r$  rounds perform the following:

$$\begin{aligned}L_{i-1} &= R_i \oplus f(L_i, K_i) \\ R_{i-1} &= L_i\end{aligned}$$

3. Finally the plaintext is  $P = (L_0, R_0)$ .
- ▶ We never have to invert the function  $f$  so we can always decrypt for *any* function  $f$ . However, choice of  $f$  is critical for security as it is the only **non-linear** part of the encryption function.

# SPNs

- ▶ A substitution-permutation network is an iterated cipher.
- ▶ The block length  $n$  must allow each block to be split into  $m$  sub-blocks of length  $l$  so that  $n = lm$ . Two permutations are defined.
- ▶ Permutation  $\pi_S$  operates on sub-blocks of size  $l$  bits:

$$\pi_S : \{0, 1\}^l \rightarrow \{0, 1\}^l$$

The permutation  $\pi_S$  is normally called an S-box (substitution box).

- ▶ Permutation  $\pi_P$  swaps the inputs from  $\{1, \dots, n\}$ . This is similar to the transposition cipher.

$$\pi_P : \{1, 2, \dots, n\} \rightarrow \{1, 2, \dots, n\}$$

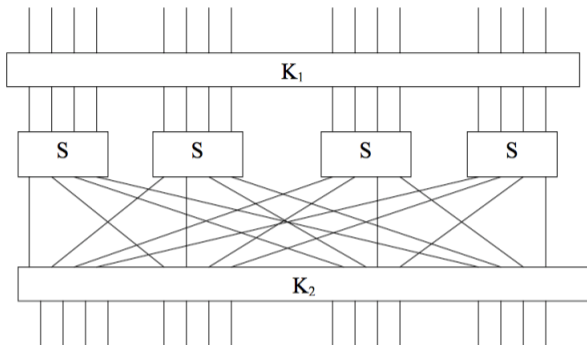
# Steps in SPN round function

The round function is defined by three steps:

1. The round key  $K_i$  is XORd with the current state block  $W_i$
2. Each sub-block is replaced (substituted) by application of  $\pi_S$
3. The whole block is permuted using  $\pi_P$

In the following picture the boxes marked  $S$  implement the permutation  $\pi_S$ . One complete round is shown with the start of a second one.

# SPN network



- ▶ The round key  $K_i$  is added (XOR) into the current block
- ▶ The same substitution,  $S$ , is applied to each sub-block
- ▶ The whole block is permuted at the bit level (transposition)

# One-round SPN

Source: Introduction to Modern Cryptography, Jonathan Katz and Yehuda Lindell, third edition

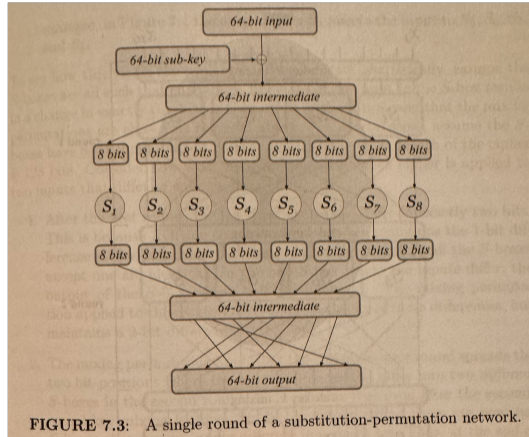
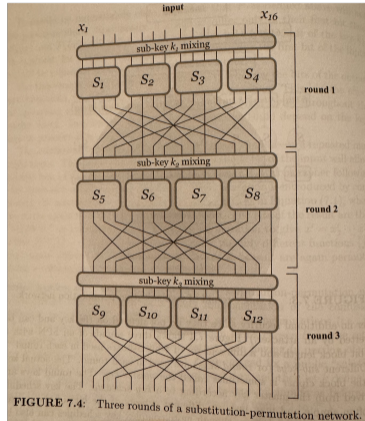


FIGURE 7.3: A single round of a substitution-permutation network.

# Three-round SPN

Source: Introduction to Modern Cryptography, Jonathan Katz and Yehuda Lindell, third edition



# Avalanche effects

- ▶ Good block ciphers typically exhibit *avalanche effects* in the ciphertext with respect to both key and plaintext.
- ▶ **Plaintext avalanche:** a small change in the plaintext should result in a large change in the resulting ciphertext. Ideally, changing one bit of the plaintext changes each of the bits in the output block with probability  $1/2$ .
- ▶ We can relate plaintext avalanche effect to Shannon's notion of diffusion.
- ▶ **Key avalanche:** a small change in the key (with the same plaintext) should result in a large change in the resulting ciphertext.
- ▶ We can relate key avalanche effect to Shannon's notion of confusion.

# Differential and Linear Cryptanalysis

- ▶ Differential cryptanalysis is a powerful technique first published in 1992. It is a chosen plaintext attack.
- ▶ It is based on the idea that the difference between two input plaintexts can be correlated to the difference between two output ciphertexts.
- ▶ Linear cryptanalysis is a known plaintext attack first published in 1993. It can be theoretically used to break DES and other schemes.
- ▶ Modern block ciphers are normally designed to be immune to both differential and linear cryptanalysis as a minimum security requirement.

# Contents

## Block Cipher Principles

Product Ciphers and Iterated Ciphers

Feistel Ciphers

Substitution-permutation networks

Standard security properties

## DES

History of DES

DES algorithm

Brute Force Attack on DES

Double and triple DES

## AES

AES History

AES Algorithm

Comparison of AES and DES



# Data Encryption Standard (DES)

- ▶ Designed by researchers from IBM and submitted to the NBS (National Bureau of Standards) in US in a call for a publicly available cipher.
- ▶ Approved in 1977 as the US standard for encryption.
- ▶ The encryption and decryption definitions are public property. The security of the DES algorithm resides in the difficulty of decryption without knowledge of the key (Kerckhoffs's principle).
- ▶ DES is a 16-round Feistel cipher with key length of 56 bits and data block length of 64 bits.

# Encryption operation

An input block of 64 bits denoted by  $P$ :

- Step 1** The 64 bits of  $P$  are permuted according to an initial fixed permutation, denoted by  $IP$ .
- Step 2** After the permutation, 16 rounds of a Feistel operation are applied, denoted by function  $f$ . A different 48 bit subkey is used for each round of the  $f$  function.
- Step 3** After the 16 round operations, a final fixed inverse permutation, denoted by  $IP^{-1}$ , is applied.

After Step 3, the output ciphertext block, denoted by  $C$ , has been formed.

# DES Feistel operation

For each round the following steps are followed:

**Step 1** Expand 32 bits to 48 bits

**Step 2** Bitwise modulo two add (XOR) 48 bits to 48 bit subkey for round

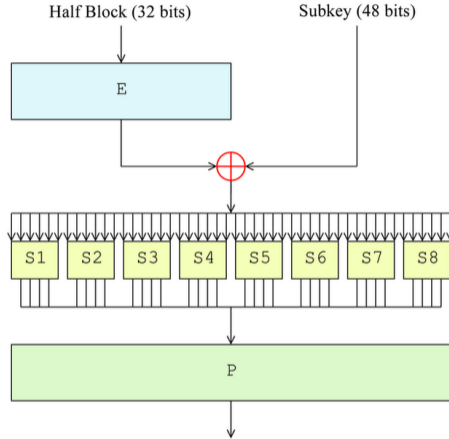
**Step 3** Break 48 bits into eight blocks of six bits each

**Step 4** Put block  $i$  into substitution table  $i$  resulting in block of length four

**Step 5** Apply permutation to resulting 32 bits



# Feistel f function used in DES



Picture courtesy of Wikimedia commons

## S-box example

Row No.	Column Number															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

- ▶ Suppose input block  $B$  is  $x_1x_2x_3x_4x_5x_6$
- ▶ Digits  $x_1$  and  $x_6$  define row number between 0 and 3
- ▶ Digits  $x_2x_3x_4x_5$  define column number between 0 and 15

# Key schedule

- ▶ Each of the sixteen rounds involves 48 bits of the 56 bit key.
- ▶ Each 48-bit subkey is defined by a series of permutations and shifts on the full 56-bit key

# Brute force attack

- ▶ A brute force attack on a block cipher consists of testing all possible  $2^k$  keys in order to find the key  $K$ .
- ▶ The right key can be identified by using a small number of ciphertext blocks, or by looking for low entropy in the decrypted plaintext.
- ▶ In the case of DES there are  $2^{56}$  keys to test so that, on the average, it would take  $2^{55}$  trial samples to find the right key.
- ▶ Right from its first publication the short size of the DES key was criticized.
- ▶ As technology evolved (i.e. computational power), this became insecure.

## Real world attacks

1997	<ul style="list-style-type: none"><li>• \$10,000 DES Challenge in February 1997</li><li>• Solved in June 1997</li><li>• Linked together thousands of computers over the Internet (parallel processing)</li></ul>
1998	<ul style="list-style-type: none"><li>• EFF DES Cracker built</li><li>• Cost less than \$250 000</li><li>• Less than three days to find 56-bit DES key</li><li>• Searched 88 billion keys per second</li></ul>
1999	<ul style="list-style-type: none"><li>• EFF DES Cracker plus distributed search</li><li>• 22 hours 15 minutes to find 56-bit DES key</li><li>• Searched 245 billion keys per second</li></ul>
2007	<ul style="list-style-type: none"><li>• Parallel FPGA-based machine COPACOBANA</li><li>• Cost \$10,000 to build</li><li>• Less than 1 week to find 56-bit DES key</li></ul>

# Double encryption

- ▶ Let  $K_1$  and  $K_2$  denote two keys of the block cipher. Then double encryption is defined by:

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

- ▶ If the key length of the original block cipher is  $k$  then exhaustive key attack requires  $2^{2k-1}$  trials on average.
- ▶ In fact there is a time-memory trade-off which reduces this using a meet-in-the-middle method.

# Meet-in-the-middle attack on double encryption

Suppose we have a double encrypted plaintext/ciphertext pair  $(P, C)$ :

**Step 1.** For each key, store  $C' = E(P, K)$  in memory.

**Step 2.** Check whether  $D(C, K') = C'$  for any key  $K'$ .

**Step 3.**  $K$  from Step 1 is  $K_1$  and  $K'$  from Step 2 is  $K_2$ .

**Step 4.** Check whether key values work for other  $(P, C)$  pairs.

This attack requires storage of one plaintext block for every possible key.

# Attack applied to double DES

- ▶ The attack requires:
  - ▶ storage of one plaintext block for every key
  - ▶ a single encryption for every key
  - ▶ a single decryption for every key
- ▶ For DES algorithm this would require storage of  $2^{56}$  64-bit blocks,  $2^{56}$  encryption operations and  $2^{56}$  decryption operations.
- ▶ Expensive, but much easier than brute force search through  $2^{111}$  keys.

# Triple encryption

- ▶ Much better security can be provided by using triple encryption.
- ▶ In general three keys  $K_1$ ,  $K_2$  and  $K_3$  are used. Encryption is defined by:

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

- ▶ This is secure from the above meet-in-the-middle attack.
  - ▶  $EDE$  for backward compatibility (if the keys are the same, i.e.  $K_1 = K_2 = K_3 = K$  this is the same as  $E$ )
  - ▶ If  $E$  has strong pseudorandom properties, so does  $D = E^{-1}$ .

# Standardized options

- ▶ The 1999 version of the DES standard specified three options:
  1. Use three independent keys  $K_1, K_2, K_3$ . The most secure.
  2. Use two keys with  $K_1 = K_3$ . Still secure enough.
  3. Use one key with  $K_1 = K_2 = K_3$ . Backward compatible with single key DES (vulnerable to brute-force key search).
  
- ▶ [NIST SP 800-131A](#), March 2019 states:
  - ▶ Two-key triple DES is allowed only for *legacy use* (decryption only).
  - ▶ Three-key triple DES remains allowed in existing applications only, and after 2023 only for legacy use.

# Contents

## Block Cipher Principles

- Product Ciphers and Iterated Ciphers

- Feistel Ciphers

- Substitution-permutation networks

- Standard security properties

## DES

- History of DES

- DES algorithm

- Brute Force Attack on DES

- Double and triple DES

## AES

- AES History

- AES Algorithm

- Comparison of AES and DES

# Advanced Encryption Standard (AES)

- ▶ Due to controversy over DES design, AES was designed in an open competition for everyone in the world to participate
- ▶ The process took several years with much public debate
- ▶ From 15 original submissions, 5 finalists were all widely believed secure
- ▶ Winner was Rijndael, designed by two Belgian cryptographers, Vincent Rijmen and Joan Daeman at KU Leuven

# AES overview

- ▶ Symmetric key block cipher
- ▶ Deals with 128-bit data block and 128-, 192- or 256-bit master key
- ▶ Number of rounds, NR, is 10, 12 or 14 (for 128-, 192-, 256-bit keys)
- ▶ Byte-based design, dealing with 8 bits at a time in internal operations
- ▶ The structure is essentially a substitution-permutation network:
  - ▶ initial round key addition
  - ▶ NR-1 rounds
  - ▶ final round

# State - matrix of bytes

Data block size = 16 bytes structured as follows:

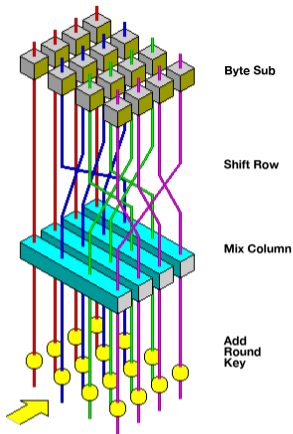
$a_{00}$	$a_{01}$	$a_{02}$	$a_{03}$
$a_{10}$	$a_{11}$	$a_{12}$	$a_{13}$
$a_{20}$	$a_{21}$	$a_{22}$	$a_{23}$
$a_{30}$	$a_{31}$	$a_{32}$	$a_{33}$

Mixture of finite field operations in  $GF(2^8)$  and bit string operations.

# Round transformation

Four basic operations:

1. ByteSub (non-linear substitution)
  2. ShiftRow (permutation)
  3. MixColumn (diffusion)
  4. AddRoundKey (XOR)
- 
- ▶ Essentially a substitution-permutation network ( $n = 128, l = 8$ )
  - ▶ S-box is look-up table but mathematically defined in  $GF(2^8)$



**Figure:** Illustration from

[https://en.wikipedia.org/wiki/Advanced\\_Encryption\\_Standard](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard)

## GF(8) example

The polynomial  $x^3 + x + 1$  is irreducible in  $\mathbb{Z}_2$ .

product mod $p(x)$	0	1	$x$	$x+1$	$x^2$	$x^2+1$	$x^2+x$	$x^2+x+1$
0	0	0	0	0	0	0	0	0
1	0	1	$x$	$x+1$	$x^2$	$x^2+1$	$x^2+x$	$x^2+x+1$
$x$	0	$x$	$x^2$	$x^2+x$	$x+1$	1	$x^2+x+1$	$x^2+1$
$x+1$	0	$x+1$	$x^2+x$	$x^2+1$	$x^2+x+1$	$x^2$	1	$x$
$x^2$	0	$x^2$	$x+1$	$x^2+x+1$	$x^2+x$	$x$	$x^2+1$	1
$x^2+1$	0	$x^2+1$	1	$x^2$	$x$	$x^2+x+1$	$x+1$	$x^2+x$
$x^2+x$	0	$x^2+x$	$x^2+x+1$	1	$x^2+1$	$x+1$	$x$	$x^2$
$x^2+x+1$	0	$x^2+x+1$	$x^2+1$	$x$	1	$x^2+x$	$x^2$	$x+1$

<http://homepages.math.uic.edu/~leon/mcs425-s08/handouts/field.pdf>

# Key schedule

- ▶ The master key input is 128 bits (or 192 bits or 256 bits).
- ▶ Each of the 10 (or 12 or 14 respectively) encryption and decryption rounds uses a 128-bit subkey.
- ▶ The number of subkeys required is one for each round (10/12/14) plus an initial subkey. Therefore, for a 128-bit key 11 subkeys are required.
- ▶ The key schedule derives the 128-bit subkeys from the 128-bit master key.

# AES security

- ▶ Some cracks have appeared but not significant breaks
- ▶ Attacks exist on reduced-round versions of AES
- ▶ *Related key attacks* exist. They require the attacker to obtain ciphertext encrypted with a key related to the actual key in a specified way.
- ▶ Most serious real attack so far reduces effective key size by around 2 bits.

# DES/AES comparison

- ▶ **Data block size:** DES - 64 bits; AES - 128 bits
- ▶ **Key size:** DES - 56 bits; AES - 128, 192 or 256 bits
- ▶ **Design structure:**
  - ▶ both are iterated ciphers
  - ▶ DES has a Feistel structure; AES is a SPN;
  - ▶ DES is bit-based; AES is byte-based
  - ▶ AES substantially faster in both hardware and software

# Conclusion

- ▶ Block ciphers are the workhorses of secure communications
- ▶ AES is the choice of today but triple-DES is still in use in older applications
- ▶ Designing good block ciphers is a difficult and time-consuming process and requires years of validation by experts
- ▶ In future lectures we will see how to use block ciphers as a building block for confidentiality and authentication

# Questions?