TOPIC: 6.857

DATE: 3/6/17

FILE UNDER:

PAGE: L08.1

Admin:

Pset #2 due 3/13

Projects!

Today:

Block ciphers:

✓ DES

✓ AES

Modes of operation

✓ Ideal cipher (ECB, CTR, CBC)

Desai's "UFE" mode

Readings:

Ferguson Ch 3,

Paar Ch 3,4

Katz Ch 5

Wikipedia: "Block cipher modes of operation"

"ciphertext stealing"

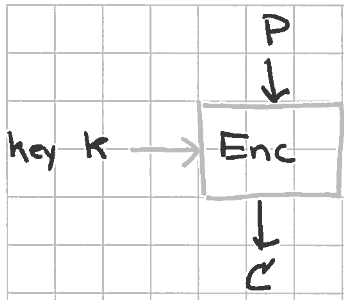
TOPIC:

DATE:

FILE UNDER:

PAGE: L5.2

Block ciphers:



plaintext block

ciphertext block

<u>fixed-length</u> P, C, K

DES: $|P| = |C| = 64$ bits $|K| = 56$ bits

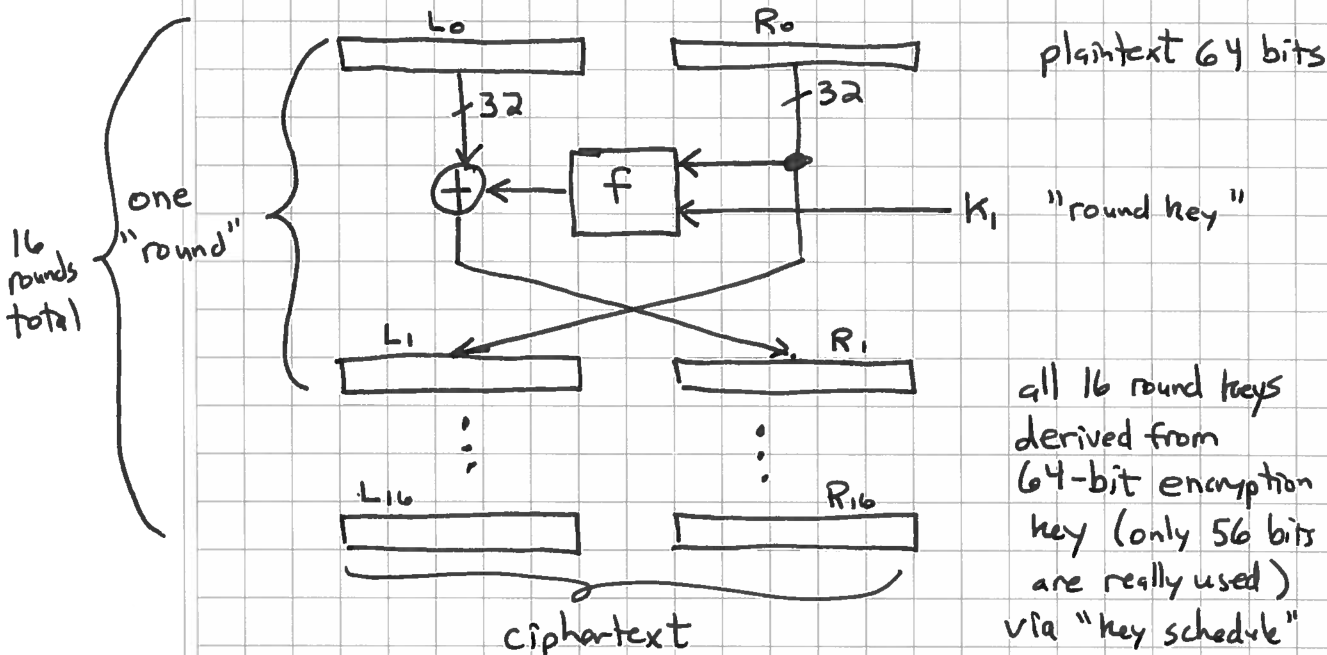
AES: $|P| = |C| = 128$ bits $|K| = 128, 192, 256$ bits

Use a "mode of operation" to handle variable-length input.

DES

"Data Encryption Standard"
Standarized in 1976. Now deprecated in favor of AES.

"Feistel structure":



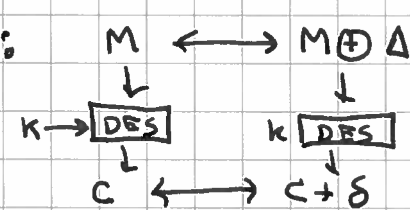
all 16 round keys derived from 64-bit encryption key (only 56 bits are really used) via "key schedule"

<u>Note</u>: Invertible for <u>any</u> f and <u>any</u> key schedule.

f uses 8 "S-boxes" mapping 6-bits ⇒ 4 bits nonlinearly.

Key is <u>too</u> <u>short</u>! (Breakable now quite easily by brute-force)

Subject to <u>differential attacks</u>:



2^{47} chosen pairs (Biham/Shamir)

Subject to <u>linear</u> <u>attacks</u>:

e.g. if $M_3 \oplus M_{15} \oplus C_2 \oplus K_{14} = 0$ (eqn on bits)
with prob $p = 1/2 + \varepsilon$

then need $1/\varepsilon^2$ samples to break (Matsui, 2^{43} PT/CT pairs)

TOPIC:

DATE:

FILE UNDER:

AES

"Advanced Encryption Standard" (U.S. govt)

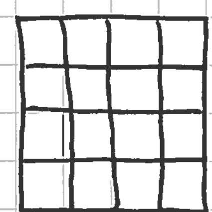
Replaces DES

AES "contest" 1997–1999:
15 algorithms submitted: RC6, Mars, Twofish, Rijndael, ...
winner = Rijndael (by Joan Daemen & Vincent Rijmen, (Belgians))

Specs: 128-bit plaintext/ciphertext blocks
128, 192, or 256-bit key
10, 12, or 14 rounds (dep. on key length)

Byte-oriented design (some math done in Galois field $GF(2^8)$)

View input as 4x4 byte array:



$4 \times 4 \times 8 = 128$

For version with 128-bit keys, 10 rounds:

- Derive 11 "round keys", each 128 bits (4x4x byte)
- In each round:
  1. XOR round key
  2. Substitute bytes (lookup table)
  3. Rotate rows (by different amts)
  4. Mix each column (by linear opn)

(last round has another round key xored in instead of mix-column)

- Output final state

See readings for details.
There are very fast implementations. Also Intel has put supporting hardware into its CPU's.

Security: Good; perhaps # rounds should be a bit larger...

For practical purposes, can treat AES as <u>ideal block cipher</u>:

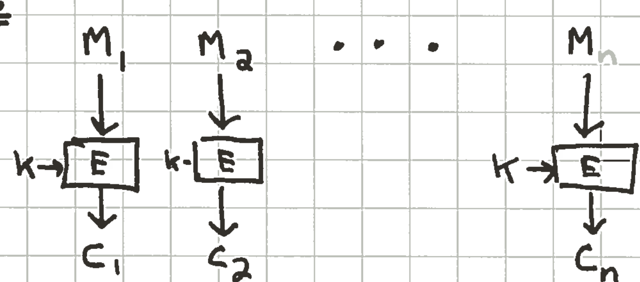
[For each key, mapping Enc(K, •) is a random independent permutation of $\{0,1\}^{128}$ to itself.

Modes of Operation:

How to encrypt variable-length messages? (using AES)

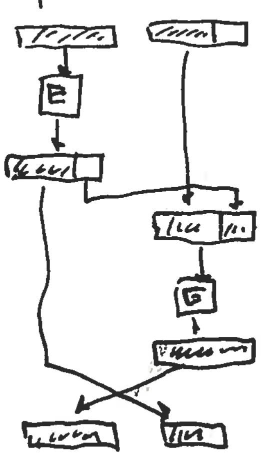
"ECB" = "Electronic code book"
"CTR" = "Counter mode"
"CBC" = "Cipher-block chaining" (& CBC-MAC)
"CFB" = "Cipher feedback"
... (others...)

ECB:



← divide data into b-bit blocks, where b = input block size

ciphertext stealing



To handle data that is not a multiple of b bits in length:
- Append a "1" bit (<u>always</u>)
- Append enough "0" bits to make length a multiple of b bits.

This gives <u>invertible</u> (1-1) "padding" operation.
Pad before encryption; unpad after decryption.

ECB preserves many patterns: repeated message blocks
⇒ repeated ciphertext blocks

ECB really only good for encrypting <u>random</u> data (e.g. keys)

TOPIC:

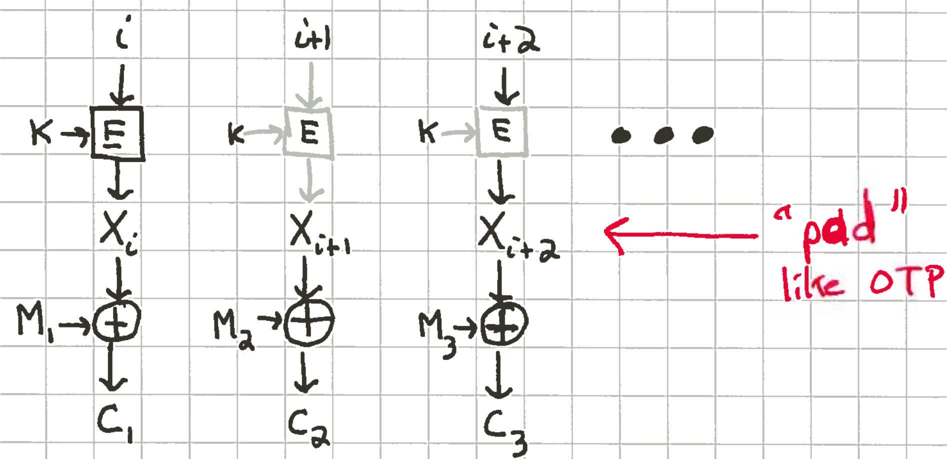
DATE:

FILE UNDER:

PAGE: L5.6

CTR (Counter mode):

Generate a PR (pseudorandom) sequence by encrypting i, $i+1$, ...
XOR with message to obtain ciphertext.



Initial counter value can be transmitted first:

$i, C_1, C_2, \ldots$

Of course, no counter value should be re-used!