Block Cipher

Bahan Kuliah Keamanan Data

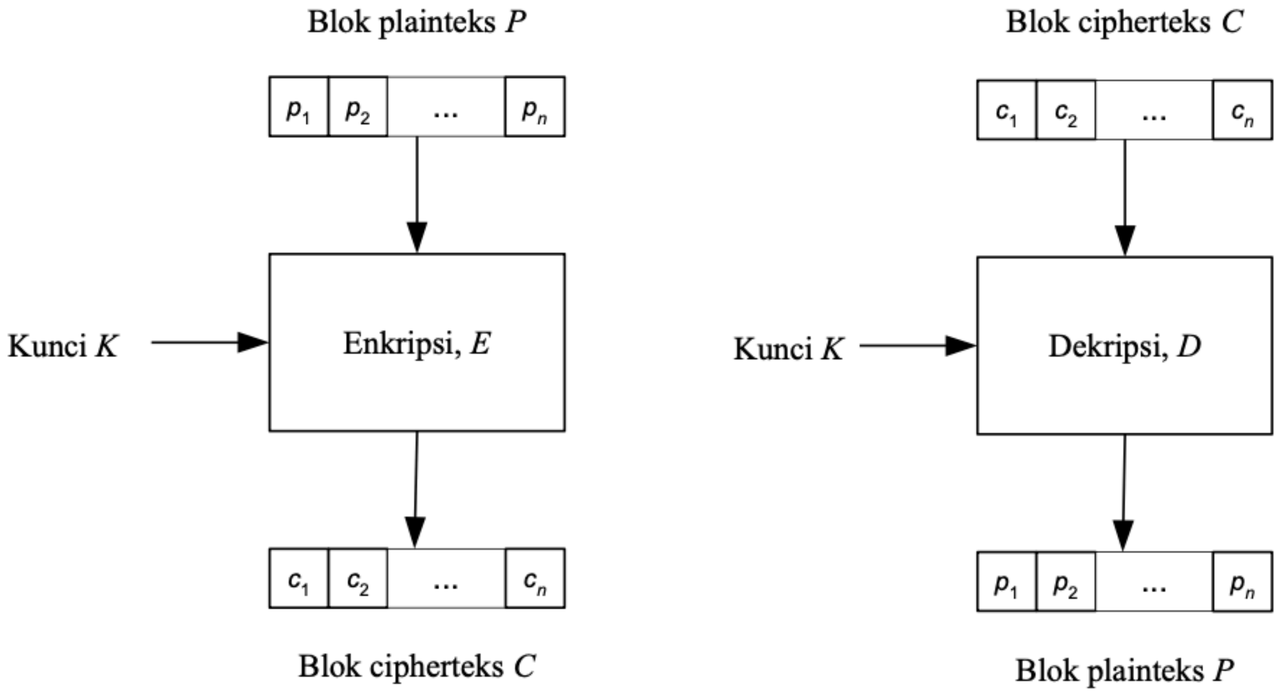
Sevi **Nurafni**

Fakultas Sains dan Teknologi
Universitas Koperasi Indonesia 2025

- Bit-bit plainteks dibagi menjadi blok-blok bit dengan panjang sama, misalnya 64 bit.
- Panjang blok cipherteks = panjang blok plainteks.
- Enkripsi dilakukan terhadap blok plainteks dengan bit-bit kunci
- Panjang kunci eksternal (yang diberikan oleh user) tidak harus sama dengan panjang blok plainteks

- Blok plainteks (P) berikuran n bit:
- $P = (p_1, p_2, \ldots, p_n), \qquad p_i \in \{0,1\}$
- Blok cipherteks (C) berukuran n bit:
- $C = (c_1, c_2, \ldots, c_n), . \qquad c_i \in \{0,1\}$

Skema enkripsi dan dekripsi pada cipher blok



Mode Operasi Cipher Blok

- Mode operasi: berkaitan dengan cara blok dioperasikan sebelum dienkripsi/dekripsi oleh fungsi E dan D.

- Ada 5 mode operasi cipher blok:
  - Electronic Code Book (ECB)
  - Cipher Block Chaining (CBC)
  - Cipher Feedback (CFB)
  - Output Feedback (OFB)
  - Counter Mode

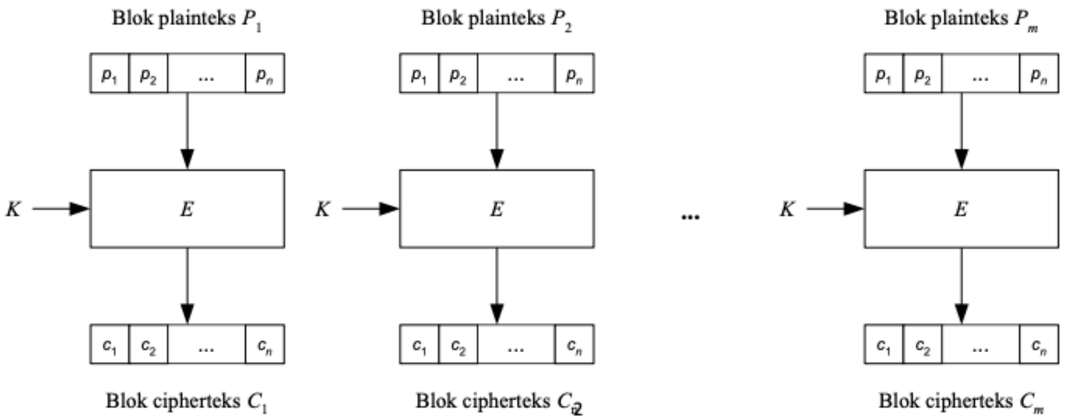
Electronic Code Book (ECB)

- Setiap blok plainteks P_i dienkripsi secara individual dan independen dari blok lainnya menjadi blok cipherteks C_i

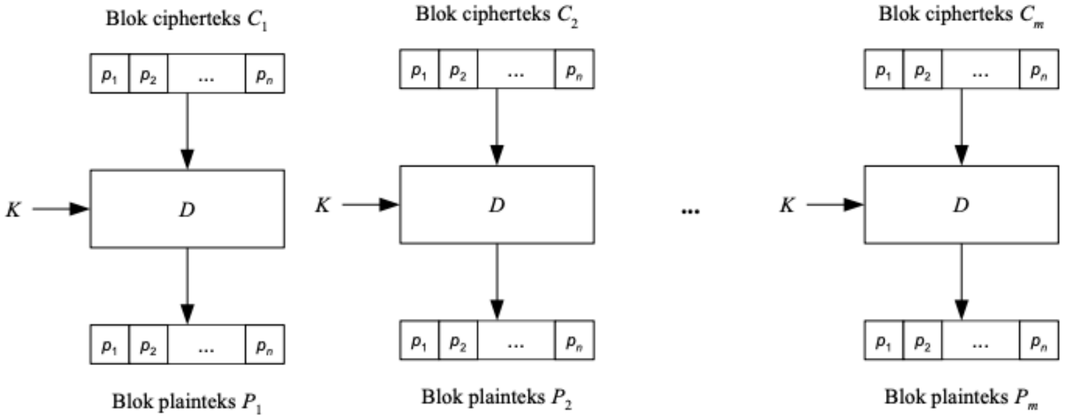
- Enkripsi: $C_i = E_k(P_i)$
- Dekripsi: $P_i = D_k(C_i)$

- Yang dalam hal ini, P_i dan C_i masing-masing blok plainteks dan cipherteks ke-i

Mode ECB



(a) Enkripsi



Contoh

- Plainteks: `10100010001110101001`
- Bagi plainteks menjadi blok-blok 4-bit:
- `1010 0010 0011 1010 1001 (HEX: A23A9)`
- Kunci (juga 4-bit): `1011`

- Misalkan fungsi enkripsi E yang sederhana algoritmanya sebagai berikut:
- XOR-kan blok plainteks P_i dengan K
- Geser secara wrapping bit-bit dari hasil langkah 1 posisi ke kiri

$$E_k(P) = (P \oplus K) << 1$$

$$E_K(P) = (P \oplus K) << 1$$

Enkripsi:

	1010	0010	0011	1010	1001	
	1011	1011	1011	1011	1011	⊕
Hasil *XOR*:	0001	1001	1000	0001	0010	
Geser 1 bit ke kiri:	0010	0011	0001	0010	0100	
Dalam notasi HEX:	2	3	1	2	4	

Jadi, hasil enkripsi plainteks

`10100010001110101001` (`A23A9` dalam notasi HEX)

adalah

`00100011000100100100` (`23124` dalam notasi HEX)

- Pada mode ECB, blok plainteks yang sama selalu dienkripsi menjadi blok cipherteks yang sama.

	1010	0010	0011	1010	1001	
	1011	1011	1011	1011	1011	⊕
Hasil *XOR*:	0001	1001	1000	0001	0010	
Geser 1 bit ke kiri:	0010	0011	0001	0010	0100	
Dalam notasi HEX:	2	3	1	2	4	

- Pada contoh di atas, blok `1010` muncul dua kali dan selalu dienkripsi menjadi `0010`

- Karena setiap blok plainteks yang sama selalu dienkripsi menjadi blok cipherteks yang sama, maka secara teoritis dimungkinkan membuat buku kode plainteks dan cipherteks yang berkorespinden (asal kata “code book” di dalam ECB).
- Enkripsi/deksripsi dilakukan dengan look up table

Plainteks	Cipherteks
0000	0100
0001	1001
0010	1010
…	…
1111	1010

- Untuk setiap kunci K yang berbeda, dibuat buku kode yang berbeda pula. Jika panjang kunci n bit, maka terdapat 2^n buku kode.

- Jumlah entri (baris) di dalam setiap buku kode untuk adalah 2^n

- Namun, semakin besar ukuran blok, semakin besar pula ukuran buku kodenya.

- Misalkan jika blok berukuran 64 bit, maka buku kode tersiri dari 2^{64} buah kode (entry), yang berarti terlalu besar untuk disimpan. Lagipula, setiap kunci mempunyai buku kode yang berbeda.

- Jika panjang plainteks tidak habis dibagi dengan ukuran blok, maka blok terakhir berukuran lebih pendek daripada blok-blok lainnya.

- Untuk itu, kita tambahkan bit-bit padding untuk menutupi kekurangan bit blok

- Misalnya ditambahkan 0 bit semua, atau 1 bit semua, atau bit 0 dan 1 berselang-seling

Kelebihan Mode ECB

1. Karena tiap blok plainteks dienkripsi secara independen, maka kita tidak perlu mengenkripsi pesan secara sekuensial,
2. Kesalahan 1 atau lebih bit pada blok chiperteks hanya mempengaruhi cipherteks yang bersangkutan pada proses dekripsi,

Kelemahan Mode ECB

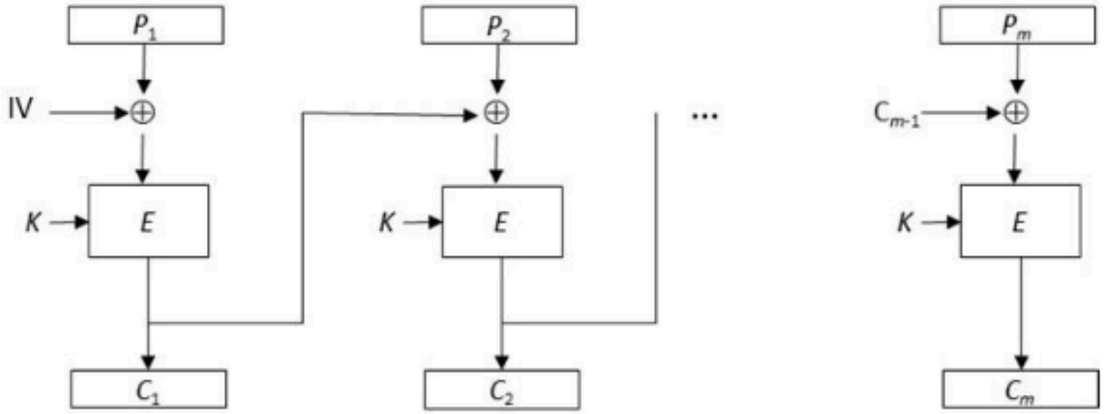
1. Karena plainteks sering mengandung bagian yang berulang (sehingga terdapat blok-blok plainteks yang sama), maka enkripsinya menghasulkan blok cipherteks yang sama pula,
2. Pihak lawan dapat memanipulasi cipherteks untuk mengelabui penerima pesan

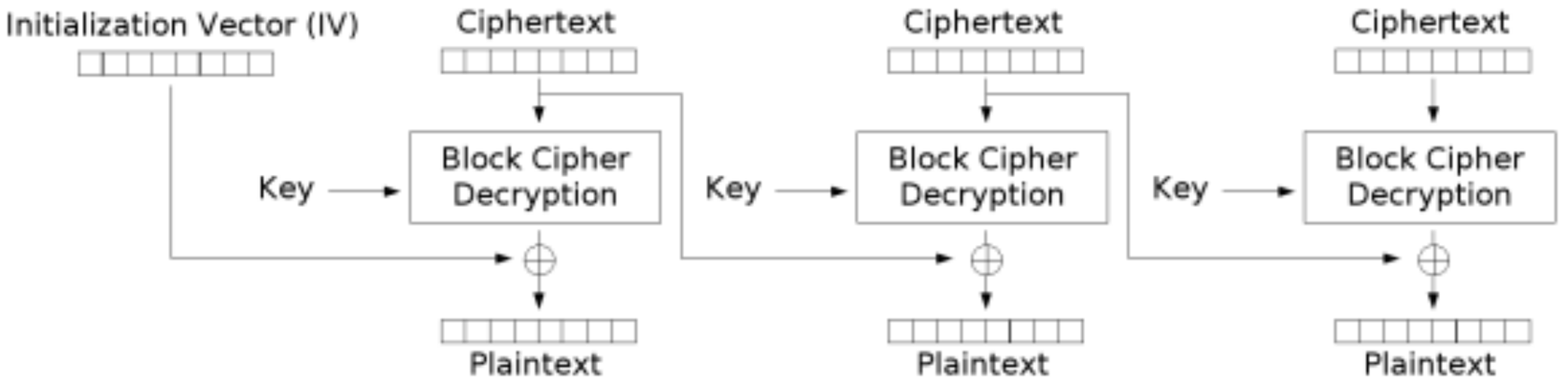
Cipher Block Chaining (CBC)

- Tujuan: membuat ketergantungan antar blok

- Setiap blok cipherteks bergantung tidak hanya pada blok plainteksnya tetapi juga pada seluruh blok plainteks sebelumnya.

- Hasil enkripsi blok sebelumnya di-umpan balikkan ke dalam enkripsi blok yang *current*.


P_1
P_2
P_m
IV
C_{m-1}
...
K
E
K
E
K
E
C_1
C_2
C_m


Initialization Vector (IV)
Ciphertext
Ciphertext
Ciphertext
Key
Block Cipher Decryption
Key
Block Cipher Decryption
Key
Block Cipher Decryption
Plaintext
Plaintext
Plaintext

- Enkripsi blok pertama memerlukan blok semu (C_0) yang disebut IV (Initialization Vector)

- IV dapat diberikan oleh pengguna atau dibangkitkan secara acak oleh program

- Pada dekripsi, blok plainteks diperoleh dengan cara meng-*XOR*-kan IV dengan hasil dekripsi terhadap blok chiperteks pertama

Contoh

```
10100010001110101001
```

Bagi plainteks menjadi blok-blok yang berukuran 4 bit:

```
1010   0010   0011   1010   1001
```

atau dalam notasi HEX adalah `A23A9`.

Misalkan kunci (K) yang digunakan adalah (panjangnya juga 4 bit)

```
1011
```

atau dalam notasi HEX adalah `B`. Sedangkan IV yang digunakan seluruhnya bit 0 (Jadi, C_0 = 0000)

Fungsi enkripsi E yang digunakan sama seperti sebelumnya: XOR-kan blok plainteks P_i dengan K, kemudian geser secara *wrapping* bit-bit dari $P_i \oplus K$ satu posisi ke kiri.

$$E_K(P) = (P \oplus K) << 1$$

Contoh

C_1 diperoleh sebagai berikut:

$P_1 \oplus C_0 = 1010 \oplus 0000 = 1010$

Enkripsikan hasil ini dengan fungsi E sbb:

$1010 \oplus K = 1010 \oplus 1011 = 0001$

Geser (*wrapping*) hasil ini satu bit ke kiri: 0010

Jadi, $C_1 = 0010$ (atau 2 dalam HEX)

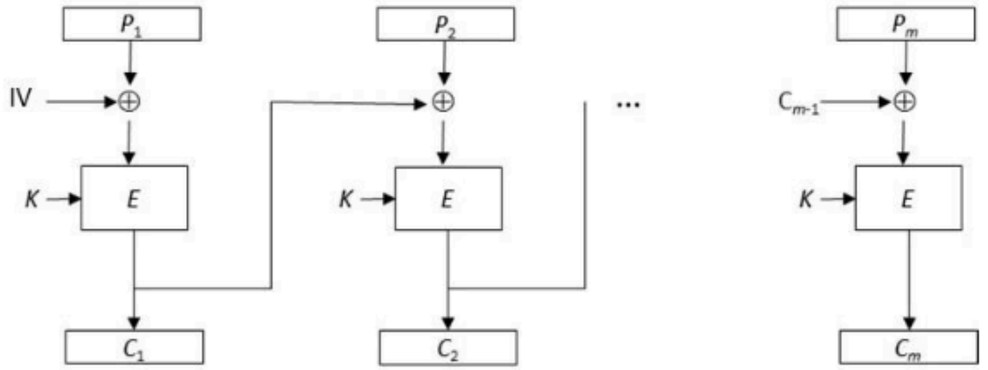
C_2 diperoleh sebagai berikut:

$P_2 \oplus C_1 = 0010 \oplus 0010 = 0000$

$0000 \oplus K = 0000 \oplus 1011 = 1011$

Geser (*wrapping*) hasil ini satu bit ke kiri: 0111

Jadi, $C_2 = 0111$ (atau 7 dalam HEX)



C_3 diperoleh sebagai berikut:

$P_3 \oplus C_2 = 0011 \oplus 0111 = 0100$

$0100 \oplus K = 0100 \oplus 1011 = 1111$

Geser (*wrapping*) hasil ini satu bit ke kiri: 1111

Jadi, $C_3 = 1111$ (atau F dalam HEX)

- Begitu seterusnya sehingga plainteks dan cipherteks hasilnya adalah:
  - Pesan (plainteks) A23A9
  - Cipherteks (mode ECB) 23124
  - Cipherteks (mode CBC) 27FDF

Kelebihan Mode CBC

1. Blok-blok plianteks yang sama tidak selalu menghasilkan blok-blok cipherteks yang sama

Kelemahan Mode CBC

1. Kesalahan satu bit pada sebuah blok plainteks akan meghasilkan kesalahan pada blok cipherteks yang berkoresponded dan kesalah tersebut merambat ke semua blok cipherteks berikutnya.
2. Kesalahan satu bit pada blok cipherteks hanya mempengaruhi blok plianteks yang berkoresponden dan satu bit pada blok plainteks berikutnya (pada posisi bit yang berkoresponden pula)

Tugas

Kelemahan Mode CBC

1. Cipher-Feedback (CFB)
2. Output-Feedback (OFB)
3. Counter Mode

SELAMAT BELAJAR