5.2.4 Design Considerations and Security

- After 2 rounds full diffusion is achieved, i.e., if one byte of input is changed, all bytes of the output are changed
- S-Box is constructed as $x \mapsto x^{-1}$ in $\mathbb{F}_{2^8}$. Advantages:
  - Simple, algebraic, highly non-linear
  - Resisting differential and linear cryptanalysis
  - No suspicion of trapdoor built in
- ShiftRow to resist the attacks: truncated differential and Square attack
- MixColumn causes diffusion among the bytes
- Key Schedule to avoid advantages from knowing parts of the key
- Presently no better attacks than exhaustive search known against AES-128. (Not entirely true $\rightsquigarrow 2^{126,1}$) Faster attacks are known if the number of rounds is ~~[illegible]~~ less than 7.
- Attacks against AES-192 and AES-256 of complexity 2^{119} are known (see Schneier) A Related-key cryptanalysis is used.
- Even more: refining this attack on AES-256 leads to a complexity of $2^{99.5}$

5.3 Other Block Ciphers

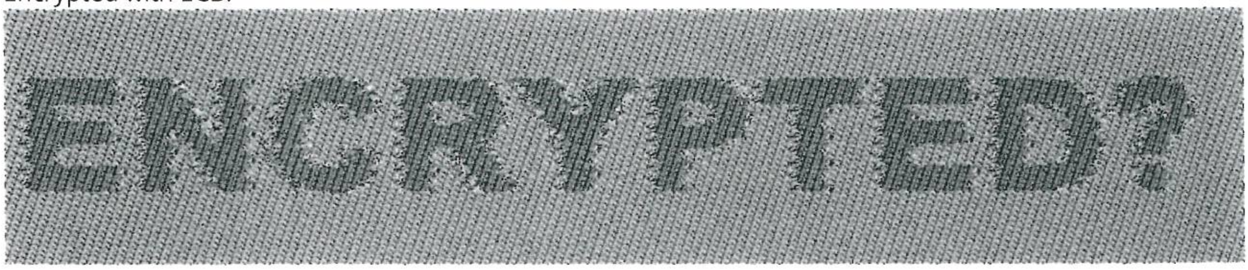
- IDEA – International Data Encryption Alg.
  Designed by Massey et al., 1990
  IDEA was part of PGP (pretty good privacy)
  Block length of 64 bits, key length 128 bits
  IDEA is secure, best known attack is exhaustive search
  Patented in Europe (1991), USA (1993), but for non-commercial applications it is free

- RC5 (Rivest et al. '94)
- Blowfish (Schneier '93)
- Serpent (Andersen, Biham, Knudsen '98)

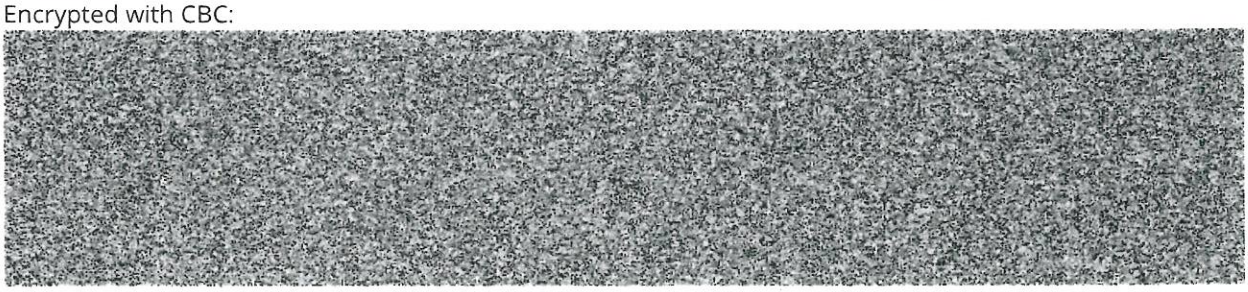
E

Input:

Encrypted with ECB:



Encrypted with CBC:



5.4 Modes of operation

Let E_K denote a blockcipher operating on blocks of fixed length using key K. 5 modes of operation were standardized in Dec. 1980

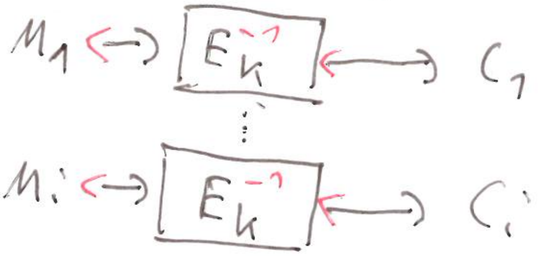
5.4.1 ECB (Electronic codebook mode)

Direct use of E_K.

Given: Plaintext blocks: $M_1, M_2, M_3, \ldots$

Encryption: $C_i = E_K(M_i)$

Decryption: $M_i = E_K^{-1}(C_i)$

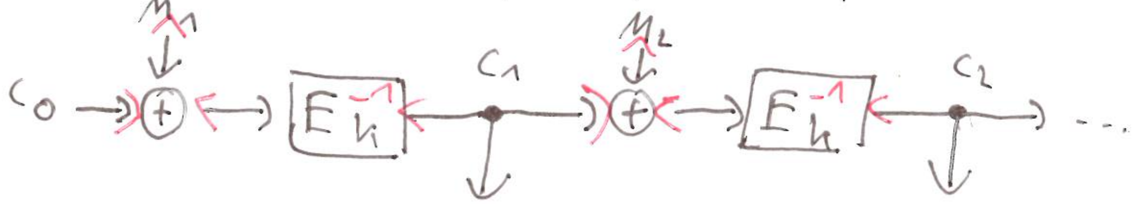


5.4.2 CBC (cipher blockchaining mode)

$$\left.\begin{array}{ll} \text{Given: Plaintext blocks} & : M_1, M_2, M_3, \ldots \\ \qquad\ \ \text{Key} & : K \\ \text{Non-secret Initial Vector} & : C_0 \end{array}\right\} (*)$$

Encryption: $C_i = E_K(M_i \oplus C_{i-1}) \qquad i = 1, 2, \ldots$

Decryption: $M_i = E_K^{-1}(C_i) \oplus C_{i-1}$

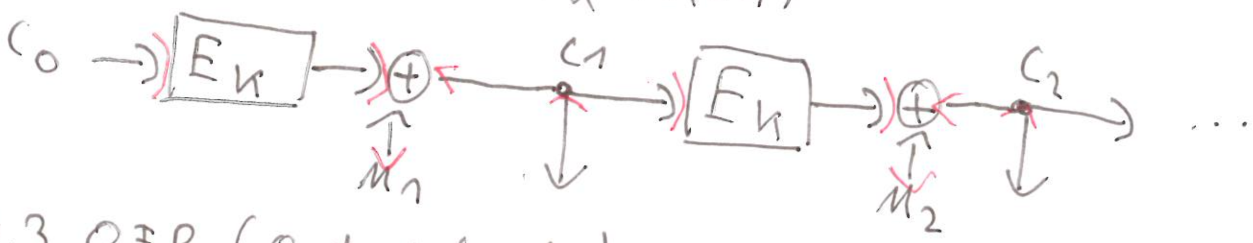


5.4.4 CFB (cipher feedback mode)

Given : (*)

Encryption : $C_i = E_K(C_{i-1}) \oplus M_i$

Decryption : $M_i = C_i \oplus E_K(C_{i-1})$



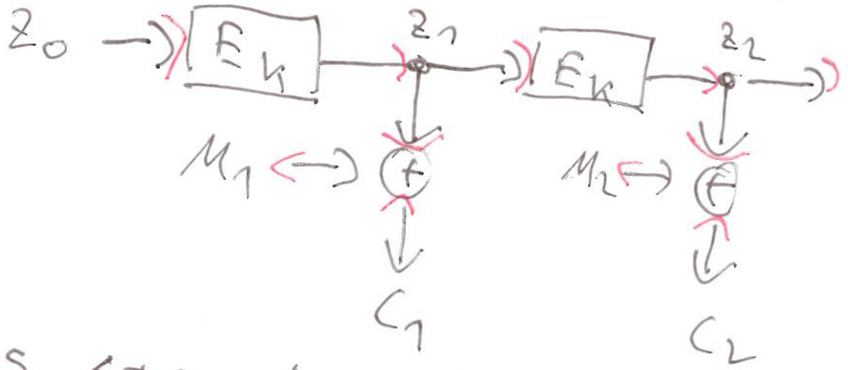
5.4.3 OFB (Output feedback mode)

Given : (*), $z_0 = C_0$

Encryption : $z_i = E_K(z_{i-1})$, $C_i = M_i \oplus z_i$

Decryption : $z_i = E_K(z_{i-1})$, $M_i = C_i \oplus z_i$

A key stream is generated and +-ored with the message

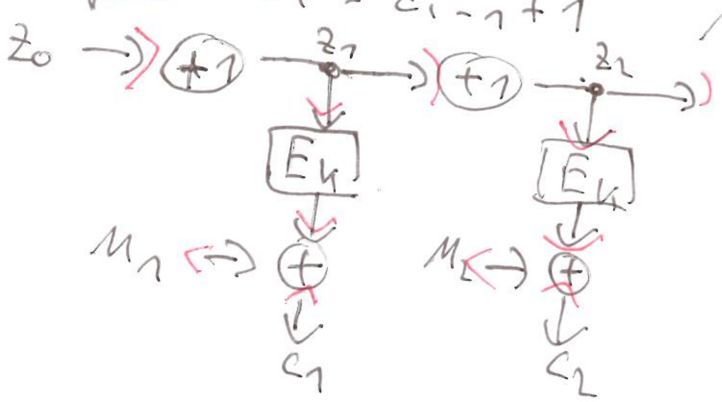


5.4.5 CTR (counter mode)

Given: (*), $z_0 = C_0$ (interpreted as some integer)

Encryption : $z_i = z_{i-1} + 1$, $C_i = E_K(z_i) \oplus M_i$

Decryption : $z_i = z_{i-1} + 1$ $\quad M_i = E_K(z_i) \oplus C_i$



In ECB, OFB, CTR: changing one plaintext block does not affect other cipher blocks.

Example: MAC – Message authentication code

In CBC and CFB modes, changing any plaintext block affects all subsequent ciphertext blocks. Appropriate for generating a MAC

- Append C_n to the message $(M_1, \dots, M_n)$
  If O/E tampers with the message, C_n does not fit any more
- The authorized receiver, knowing K, can easily verify C_n, hence, the integrity & authenticity of $(M_1, \dots, M_n)$

Example: Storing passwords

Direct plaintext storing of passwords is insecure. Hence,

- User types (name, password)
- System generates a key $K = K(\text{name, password})$ and stores $(\text{name}, BC_K(\text{password}))$
- When logging in, system compares $(\text{name}, BC_K(\text{password}))$ with the stored value.

Knowledge of $(\text{name}, BC_K(\text{password}))$ is useless for an intruder.