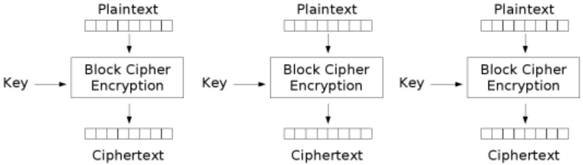
Block cipher modes

Block cipher modes

Block ciphers (like DES and AES) can be used directly only for a single block of plaintext. However, typically we want to encrypt plaintext that is much longer than a single block. How can we use a block cipher to do that securely?

ECB

Simplest way: Apply the encryption block by block
This is called *Electronic Codebook mode*, ECB.

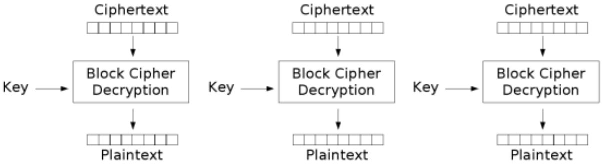


Electronic Codebook (ECB) mode encryption

Source: Wikipedia

ECB decryption

Decryption just does the operations in reverse, and uses the decrypt function of the block cipher.



Electronic Codebook (ECB) mode decryption

Source: Wikipedia

ECB is not secure

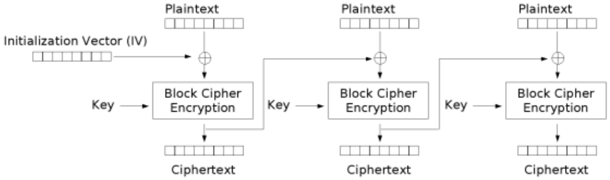
A good block mode should have the following properties:

1. [Security] Identical plaintexts shouldn't produce identical ciphertexts; and
   - Identical blocks within a plaintext shouldn't produce identical ciphertext blocks
2. [Security] There should be protection against deletion or insertion of blocks
3. [Recovery] Ciphertext transmission errors should affect only the the block containing the error
4. [Efficiency] It should be efficient (e.g., parallelisable)

ECB fails properties 1 and 2. It satisfies 3 and 4, but they are not as important as 1 and 2.

CBC

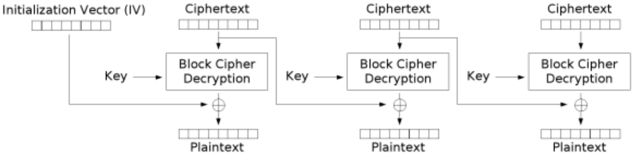
One solution: Add random “initialisation vector” (IV) to randomise the first block, and then use the current block to randomise the next block.



Cipher Block Chaining (CBC) mode encryption

Source: Wikipedia

The IV should be randomly chosen for each encryption. Note that you must store the IV with the ciphertext, otherwise it's not possible to decrypt. Thus, the IV is *random*, but not *secret*. Note that, since the ciphertext includes the IV, it is one block longer than the plaintext. So we have a small expansion in size.



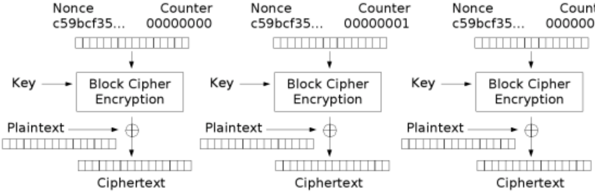
Cipher Block Chaining (CBC) mode decryption

Source: Wikipedia

Figure out which of properties 1, 2, 3, 4 hold for CBC.

Counter mode (CTR)

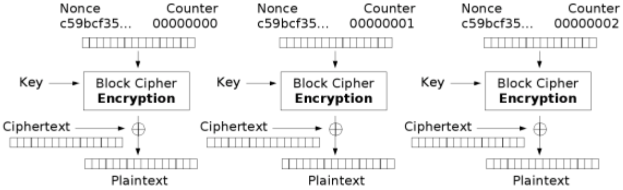
In counter mode, we don't chain the blocks together, but still we aim to make identical plaintext blocks have different ciphertext blocks. To encrypt a message, we chose a random nonce, and then set up a counter which is incremented for each block.



Counter (CTR) mode encryption

Source: Wikipedia

The nonce must be stored with or transmitted with the ciphertext blocks. Thus, similarly to CBC, CTR mode increases the size of the ciphtertext by one block.



Counter (CTR) mode decryption

Source: Wikipedia

Figure out which of properties 1, 2, 3, 4 hold for CTR.

Proper definition of security for Block Cipher Modes

We want to define whether a block mode is secure or not.

We could consider reusing the security definition for block cipher. However, that definition isn't well-suited.

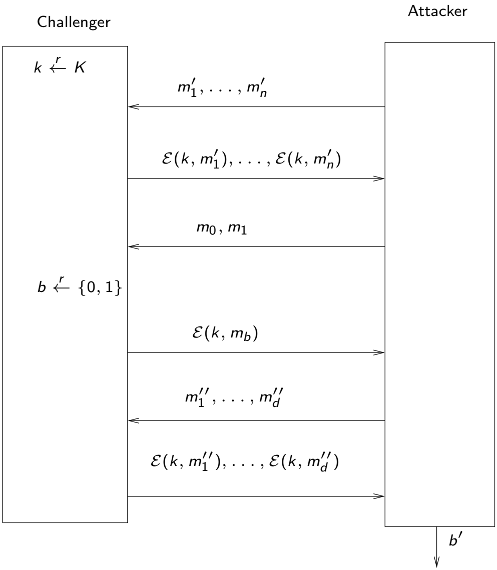
- The block cipher security game implicitly assumes that the block cipher, and the random permutation we are comparing it to, are both deterministic. With secure block cipher modes, the encryptions are randomised.
- In block cipher modes, a ciphertext bit depends only on some of the plaintext bits. A single bit change is not diffused over the entire ciphertext. So intuitively, we the block cipher security definition is too strong.

Definition

Let (E, D) be a pseudorandom permutation over (K, X), and $\mathcal{E}$ be a block cipher mode. We define the *indistinguishability under chosen-plaintext*-game between challenger and attacker as follows:

- The challenger generates a key $k \in K$ at random.
- The attacker performs a polynomial number of computations, possibly asking the challenger for the encryption by $\mathcal{E}$ of a polynomial number of arbitrary messages.
- The attacker submits two messages m_0 and m_1 to the challenger.
- The challenger selects a bit $b \in \{0, 1\}$ at random.
- The challenger returns the encryption $\mathcal{E}(k, m_b)$ to the attacker
- The attacker performs a polynomial number of computations, possibly asking the challenger for the encryption of a polynomial number of arbitrary messages.
- The attacker outputs a bit b'.

The attacker wins this game if $b' = b$.


Challenger
Attacker
$k \xleftarrow{r} K$
$m'_1, \ldots, m'_n$
$\mathcal{E}(k, m'_1), \ldots, \mathcal{E}(k, m'_n)$
m_0, m_1
$b \xleftarrow{r} \{0,1\}$
$\mathcal{E}(k, m_b)$
$m''_1, \ldots, m''_d$
$\mathcal{E}(k, m''_1), \ldots, \mathcal{E}(k, m''_d)$
b'

Intuitively, we call a block cipher mode secure if the attacker can only guess the bit b, ie wins the game half the time.

Definition

Let $Pr[b = b']$ be the probability that the attacker wins the IND-CPA-game, taken over all encryption keys of length n and all bits b. A block cipher mode satisfies *indistinguishability under chosen-plaintext attack* (IND-CPA) if

$$\left| Pr[b = b'] - \frac{1}{2} \right|$$

is negligible. Note that this probability depends on the size of K, since the security of the block cipher E depends on it.

ECB is not secure.

Let (E, D) be a secure block cipher, and let $\mathcal{E}$ be encryption using ECB mode.

The attacker can easily win the IND-CPA game. He can get the encryption of m_1 and m_2 in the first part (or even in the last part) of the game, and hence can easily distinguish which one the challenger chose.

Theorem

If (E, D) is a block cipher with key space X, the advantage of the attacker in the IND-CPA game for CBC is

$$\frac{2q^2L^2}{|X|} + 2Adv$$

where q is the number of messages encrypted with the same key k and L is the maximal length of each message, and Adv is the advantage of the attacker in the game for the secure block cipher.

For AES: must change key after using 2^{24} message of length 2^{24} each to obtain advantage of $\frac{1}{2^{32}}$

Theorem

If (E, D) is a block cipher with key space X, the advantage of the attacker in the IND-CPA game for counter mode is

$$\frac{2q^2L}{|X|} + 2Adv$$

where q is the number of messages encrypted with the same key k and L is the maximal length of each message, and Adv is the advantage of the attacker in the game for the secure block cipher.

For AES: must change key after using 2^{32} message of length 2^{32} each to obtain advantage of $\frac{1}{2^{32}}$.