Chosen plaintext attack

Chosen plaintext attack

> You choose the messages to be encrypted

CPA and deterministic ciphers

Claim

> No deterministic cipher is CPA secure

Intuition

- It leaks that two identical ciphertexts encode the same message

Proof Idea
Let we iterate the game of semantic security but we use the same key

- Adversary queries i pairs (m_{i0}, m_{i1})
- Challenger picks a message and returns the encryption
- The adversary must't be able to distinguish which ciphertext was encrypted

Attack

- Let the adversary query $(m, m) \to c$ and (m, m')
- if at the 2nd query he gets c back then m was encrypted, otherwise it was m'

CPA security

Task

- Make ciphers CPA secure
- Let E, D be the encryption, decryption algorithms

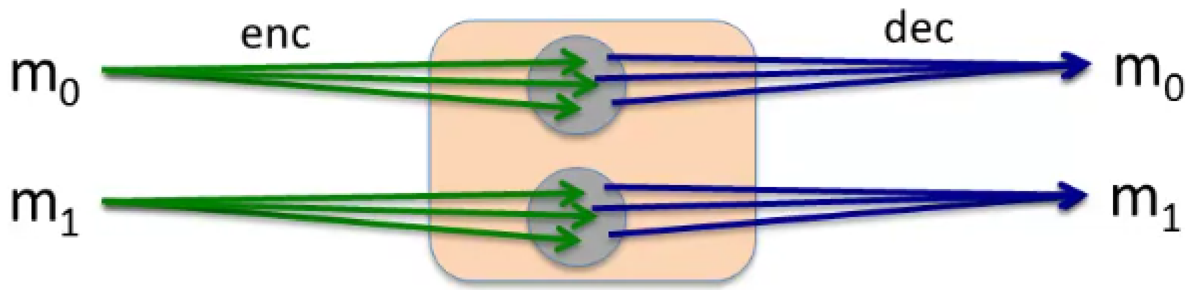
Stateful encryption

> Encryption/decryption can be **stateful**, meaning that every call to E or D willactually modify the value of k.

Randomized encryption

> Each time a plaintext is encrypted, the E algorithm chooses fresh, independent randomness specifc to that encryption.

- The main challenge in designing a randomized encryption method is to incorporate randomness into each ciphertext in such a way that decryption is still possible



- Every encryption goes to 1 different point in the “ball” each time

Ex:

- $F: K \times R \longrightarrow M$ be a secure PRF
- $E(k,m) = (r, F(k,r) \oplus m)$ for a random $r \in R$

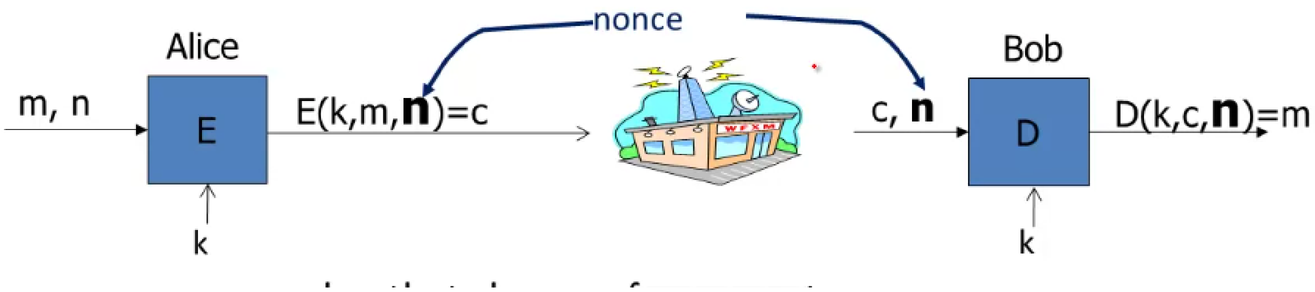
Mode of operation example:

- CBC mode

Nonce-based encryption

> We have 3 inputs $E(k,m,n)$

- A “nonce” stands for “number used only once”
  - and it refers to an extra argument that is passed to the E and D algorithms
  - A nonce does not need to be chosen randomly;
  - it does not need to be secret;
  - the pair k, n must be **different for every message**



Ex:

- Counter mode
  - Start pick a starting number then increment it for each message
  - You can send the nonce along the message
  - The parties can keep the counter