Symmetric Encryption: Modes of Operation, Semantic Security

Yan Huang

Credits: Vitaly Shmatikov (Cornell Tech)
Encrypting a Large Message

- So, we’ve got a good block cipher, but our plaintext is larger than 128-bit block size

- Modes of Operation
  - **Electronic Code Book (ECB) mode**
    - Split plaintext into blocks, encrypt each one separately using the block cipher
  - **Cipher Block Chaining (CBC) mode**
    - Split plaintext into blocks, XOR each block with the result of encrypting previous blocks
  - Also various counter modes, feedback modes, etc.
ECB Mode

- Identical blocks of plaintext produce identical blocks of ciphertext
- No integrity checks: can mix and match blocks
Information Leakage in ECB Mode

[Wikipedia]

Encrypt in ECB mode
Adobe Passwords Stolen (2013)

- 153 million account passwords
  - 56 million of them unique
- Encrypted using 3DES in ECB mode rather than hashed

Password hints
CBC Mode: Encryption

- Identical blocks of plaintext encrypted differently
- Last cipherblock depends on entire plaintext
  - Still does not guarantee integrity

Initialization vector (random)

Sent with ciphertext (preferably encrypted)
CBC Mode: Decryption

Initialization vector

plaintext

ciphertext

decrypt
decrypt
decrypt
decrypt

key

key

key

key
ECB vs. CBC

Similar plaintext blocks produce similar ciphertext blocks (not good!)

[AES in ECB mode]

[AES in CBC mode]

[Picture due to Bart Preneel]
Choosing the Initialization Vector

- Key used only once
  - No IV needed (can use IV=0)
- Key used multiple times
  - Best: fresh, random IV for every message
CBC and Electronic Voting

[Found in the source code for Diebold voting machines:

\[
\text{DesCBCEncrypt((des\_c\_block*)\,tmp, (des\_c\_block*)\,record.m\_Data, totalSize, DESKEY, NULL, DES\_ENCRYPT)\\}
\]
CTR (Counter Mode)

- Still does not guarantee integrity
- Fragile if counter repeats
When Is a Cipher “Secure”?

- Hard to recover plaintext from ciphertext?
  - What if attacker learns only some bits of the plaintext? Some function of the bits? Some partial information about the plaintext?

- Fixed mapping from plaintexts to ciphertexts?
  - What if attacker sees two identical ciphertexts and infers that the corresponding plaintexts are identical?
  - What if attacker guesses the plaintext – can he verify his guess?
  - Implication: encryption must be randomized or stateful
How Can a Cipher Be Attacked?

- Attackers knows ciphertext and encryption algorithm
  - What else does the attacker know? Depends on the application in which the cipher is used!
- Known-plaintext attack (stronger)
  - Knows some plaintext-ciphertext pairs
- Chosen-plaintext attack (even stronger)
  - Can obtain ciphertext for any plaintext of his choice
- Chosen-ciphertext attack (very strong)
  - Can decrypt any ciphertext except the target
  - Sometimes very realistic
Chosen-Plaintext Attack

Crook #1 changes his PIN to a number of his choice

PIN is encrypted and transmitted to bank

cipher(key,PIN)

Crook #2 eavesdrops on the wire and learns ciphertext corresponding to chosen plaintext PIN

... repeat for any PIN value
Very Informal Intuition

- Security against Chosen-Plaintext attack
  - Ciphertext leaks no information about the plaintext
  - Even if the attacker correctly guesses the plaintext, he cannot verify his guess
  - Every ciphertext is unique, encrypting the same message twice produces completely different ciphertexts

- Security against chosen-ciphertext attack
  - Integrity protection – it is not possible to change the plaintext by modifying the ciphertext
How to formalize CPA-Security?

Can you recall Shannon’s definition of perfect security?
Shannon’s perfect secrecy

Let \((E, D)\) be a cipher over \((K, M, C)\)

\((E, D)\) has perfect secrecy if \(\forall m_0, m_1 \in M, |m_0| = |m_1|\)

\[\{ E(k, m_0) \} = \{ E(k, m_1) \}\] where \(k \leftarrow K\).

Does this help to define CPA-Security?
The Chosen-Plaintext Game

1. $k \leftarrow \text{KeyGen}(1^n)$. $b \leftarrow \{0,1\}$. Give $\text{Enc}(k, \cdot)$ to $A$.

2. $A$ chooses as many plaintexts as he wants, and receives the corresponding ciphertexts via $\text{Enc}(k, \cdot)$.

3. $A$ picks two plaintexts $M_0$ and $M_1$. (Picking plaintexts for which $A$ previously learned ciphertexts is allowed!)

4. $A$ receives the ciphertext of $M_b$, and continues to have accesses to $\text{Enc}(k, \cdot)$.

5. $A$ outputs $b'$.

$A$ wins if $b' = b$. 
CPA Secure (one-time key)

For all efficient adversary $\mathcal{A}$, 

$$| \Pr[ b=b' ] - 1/2 |$$ 

is “negligible”.

For $i=1,...,q$, **CPA query:**

$$m_{i,0}, m_{i,1} \in M : |m_{i,0}| = |m_{i,1}|$$

$$c_i \leftarrow E(k, m_{i,b})$$

$$m_0, m_1 \in M : |m_0| = |m_1|$$

$$c \leftarrow E(k, m_b)$$

$$m'_{i,0}, m'_{i,1} \in M : |m'_{i,0}| = |m'_{i,1}|$$

$$c'_i \leftarrow E(k, m'_{i,b})$$

$b' \in \{0,1\}$
Alternative Definition of CPA-Security (one-time key)

For \( b \leftarrow \{0,1\} \), define experiment \( \text{EXP}(b) \) as:

Define \( W_b := \{ \text{event that EXP}(b) = 1 \} \).

\[
\text{Adv}(\mathcal{A}, E) := \left| \Pr[ W_0 ] - \Pr[ W_1 ] \right| \in [0,1]
\]
Alternative Definition of CPA-Security (one-time key)

E is **computational secure** if for all efficient adversary $\mathcal{A}$

$$\text{Adv}(\mathcal{A}, E) \text{ is "negligible".}$$

**Negligible**

- Concrete sense:
  e.g., $< 2^{-40}$

- Asymptotic sense:
  $\text{negl}(n) < \text{any inverse polynomial of } n$, as long as $n$ is sufficiently large.
Defining Perfect Security (one-time key)

E is **perfectly secure** if for all adversary $\mathcal{A}$

$$\text{Adv}(\mathcal{A}, E)$$ is 0.

$\leftrightarrow$ For all explicit $m_0, m_1 \in M$:

$$\{ E(k,m_0) \} = \{ E(k,m_1) \},$$ where $k \leftarrow \mathcal{R}$. 
A Simple Example

- Any deterministic, stateless symmetric encryption scheme is insecure
  - Attacker can easily distinguish encryptions of different plaintexts from encryptions of identical plaintexts
  - This includes ECB mode of common block ciphers!

**Attacker A interacts with Enc(-)**

query Enc(0)

  - Let x=0, y=1 be any two different plaintexts
  - Send x, y to the challenger
  - If C₁=Enc(0) then b=0 else b=1

- The advantage of this attacker A is 1