Symmetric Encryption

Viet Tung Hoang

The slides are loosely based on those of Prof. Mihir Bellare, UC San Diego.
Agenda

1. Modes of Encryption: ECB, CBC, CTR

2. Formalizing Security
Encryption Syntax

Key Gen

$K \xrightarrow{\$} K$

Encrypt

$M \xrightarrow{\mathcal{E}} C$

Encryption can be probabilistic

Decrypt

$C \xrightarrow{\mathcal{D}} \begin{cases} M \\ \bot \end{cases}$
(Bad) Encryption Using Blockcipher: ECB

\[ E : \{0, 1\}^k \times \{0, 1\}^n \rightarrow \{0, 1\}^n \]

Can encrypt any message whose length is a multiple of \( n \)
ECB Is **Insecure**

Message  

ECB ciphertext  

Properly encrypted ciphertext
Why Is ECB So Bad?

If $M_i = M_j$ then $C_i = C_j$
ECB Horror Stories

Half the apps in Android used ECB to encrypt data

An Empirical Study of Cryptographic Misuse in Android Applications

Zoom used ECB to encrypt video conferencing

Adobe used ECB to encrypt passwords

Zoom concedes custom encryption is substandard as Citizen Lab pokes holes in it
Randomized Encryption: CBC

randomly chosen

$C_0 \rightarrow M_1 \rightarrow E_K \rightarrow C_1 \rightarrow E_K \rightarrow C_2 \rightarrow E_K \rightarrow C_3 \rightarrow E_K \rightarrow C_4 \rightarrow M_4$

sequential
Decryption of CBC
Dealing with Fragmentary Data

**Naive solution:** Pad with $10^*$

**Example:** Suppose that the block length is 16 bytes.

![Diagram showing padding for 31 bytes and 32 bytes](image)

Padding is required, otherwise can’t decrypt

**Problem:** Waste bandwidth, and for full-length msg, waste a blockcipher call
Ciphertext Stealing in CBC

Exercise: How to use ciphertext stealing if msg is shorter than 1 block?
Randomized Encryption: CTR

fully parallelizable

randomly chosen

$C_0 + 1 \xrightarrow{E_K} C_1 \xrightarrow{E_K} C_2 \xrightarrow{E_K} C_3$

$C_0 + 2$

$C_0 + 3$
Dealing with Fragmentary Data

randomly chosen

$C_0 + 1$  
$E_K$  
$M_1$  
$C_1$

$C_0 + 2$  
$E_K$  
$M_2$  
$C_2$

truncate

$C_0 + 3$  
$E_K$  
$M_3$  
$C_3$
1. Modes of Encryption: ECB, CBC, CTR

2. Formalizing Security
Formalizing Security

**Intuition:** A good encryption scheme should hide all *partial information* about the plaintexts

But we won’t hide the length of the plaintexts

**Informal Definition:** Adversary can’t even distinguish the encryption of its *own chosen messages*

“A good disguise should not allow a mother to distinguish her own children” [GM82]
Formalizing Security: Left-or-Right

Left $\mathcal{E}$

procedure Enc($M_0, M_1$)
Return $\mathcal{E}_K(M_0)$

Right $\mathcal{E}$

procedure Enc($M_0, M_1$)
Return $\mathcal{E}_K(M_1)$

In each query, the two messages must have the same length

$\text{Adv}^{\text{lr}}_\mathcal{E}(A) = \Pr[\text{Right}_A^\mathcal{E} \Rightarrow 1] - \Pr[\text{Left}_A^\mathcal{E} \Rightarrow 1]$
Formalizing Security: Real-or-Random

\[ \text{Real}_\mathcal{E} \]

Procedure \( \text{Enc}(M) \)
Return \( \mathcal{E}_K(M) \)

\[ \text{Rand}_\mathcal{E} \]

Procedure \( \text{Enc}(M) \)
\( C \leftrightarrow \mathcal{E}_K(M'); \ C' \leftrightarrow \{0, 1\}^{|C|}; \text{ Return } C' \)

\[ \text{Adv}_{\mathcal{E}}^{\text{rr}}(A) = \Pr[\text{Real}_\mathcal{E}^A \Rightarrow 1] - \Pr[\text{Rand}_\mathcal{E}^A \Rightarrow 1] \]
Exercise: Break LR Security of ECB
Exercise: Breaking RR Security

**Question:** Break the real-or-random security of this scheme using a single query of a 2-block message.