Today: Symmetric Encryption

Authentication

Recall: Block ciphers

Encrypts blocks of fixed length

"Ideal cipher": Random permutation

Eg. of block ciphers: DES & AES

Note: Even an "ideal cipher" does not offer "perfect security"

Eg. the adv can see if the same msg is encrypted twice

Main Drawback: Encrypts msgs of fixed length

Symmetric Encryption:

- Allows to encrypt msgs of arbitrary length
ECB Mode

Insecure attempt: \( M = (M_0, M_1, \ldots, M_n) \)  
\( C_i = E_k(M_i) \) output \((C_0, C_1, \ldots, C_n)\)

Counter mode (CTR) : Generates pseudorandom bits from the key, and encrypts msg by xoring with pseudorandom bits.

**Initial vector**

\[
\begin{align*}
IV & \downarrow \\
\rightarrow & \\
E_k & \\
\rightarrow & \\
\rightarrow & \\
\rightarrow & \\
\rightarrow & + \\
M_0 &
\end{align*}
\]

\[
\begin{align*}
IV + 1 & \downarrow \\
\rightarrow & \\
E_k & \\
\rightarrow & \\
\rightarrow & \\
\rightarrow & + \\
M_1 &
\end{align*}
\]

\[
\begin{align*}
IV + N & \downarrow \\
\rightarrow & \\
E_k & \\
\rightarrow & \\
\rightarrow & + \\
M_n &
\end{align*}
\]  
\( \text{(a la one time pad)} \)

\( \uparrow \text{known as a stream cipher} \)

\[
\begin{align*}
&& \\
+ & \\
M_0 & M_1 & M_n
\end{align*}
\]

\[
\begin{align*}
C_0 & C_1 & C_n
\end{align*}
\]

Output \( IV, (C_0, C_1, \ldots, C_n) \)

Should never use same IV twice!

(for example, can choose IV at random)

Cipher Block Chaining Mode (CBC)

\[
\begin{align*}
IV & \downarrow + \\
\rightarrow & \\
E_k & \\
\rightarrow & \\
\rightarrow & \\
\rightarrow & \\
\rightarrow & \\
M_0 & M_1 & M_2
\end{align*}
\]

\[
\begin{align*}
IV, (C_0, C_1, \ldots)
\end{align*}
\]}, output
In CBC mode, if msg is not of length which is a multiple of block length, need to pad. (e.g., add $10\ldots0$ to each $msg$

Are these modes of operations secure?

We consider two security notions:

Security against Chosen Plaintext Attacks (CPA)

Security against Chosen Ciphertext Attacks (CCA).

Claim: If block cipher is indistinguishable from ideal cipher then these encryption schemes are CPA secure (if IV is random).

(They are not CCA secure).

Definition: An encryption scheme is CCA-secure if

an "efficient" (prob. poly time) adversary can win in the following game w.p. $\approx \frac{1}{2}$ ($\frac{1}{2} + \text{negl}$).

Let $Enc_k$ denote the encryption alg. with key $K$

Let $Dec_k$ denote the decryption alg.

[Note: $Enc_k$ is the alg. of the stream cipher, not the block cipher]
Game

Phase I
("Find")

- Adv is given black-box access to Enc_k, Dec_k
- Adv outputs two msgs M_0, M_1 of same length (and state information s).

Phase II
("Guess")

- Adv is given C ← Enc_k(M_b) for randomly chosen b ← {0,1}, and is given black-box access to Enc_k & Dec_k (except on C), and is given state s.
- Adv outputs bit b', and wins iff b' = b.

CPA-Game: Same except Adv is never given oracle to Dec_k (only to Enc_k)

|b - b'| is called the advantage of the Adv.

The encryption scheme is CCA (or CPA) secure if an efficient Adv, its advantage is negligible.

"Ps" that CTR is CPA-secure if E_k is ideal cipher:

Adv can query Enc_k w. many msgs and will learn

E_k(IV + j) i = 1, ... , 2^q
j = 0, 1, ... n
# of queries # of blocks
As long as the challenge msg $M_b$ is encrypted using fresh $\{IV^* + j\}$ that will never be reused, and hence $x_0, ..., x_n$ are ind. from random, and hence serve as a "good" one-time pad.

* A CPA-secure encryption must be randomized or stateful.

CBC is CPA secure if IV is chosen randomly. If IV is not random this encryption can be insecure even if the underlying block cipher is secure (ideally).

Ex: Suppose IV is unique but is used sequentially, starting $w$. IV = 1, 2, ...

arbitrary distinct

Then choose $M_0, M_1$ for challenge cipher texts (of length 1K1),

upon getting $(IV, C)$:

\[ E_k(M_0 \oplus IV) \]

Query Enc w. $M'$ st. $M' \oplus (IV + 1) = M_0 \oplus IV$, and receive $(IV + 1, C')$. If $C' = C$ then guess $b = 0$.

Otherwise, guess $b = 1$. 
Thm: CBC & CTR are not CCA secure.

Pf: Adv picks $M_0 = 0^N$ & $M_1 = 1^N$

Given $c \leftarrow \text{Enc}_k(M_b)$, let $C' = 1^{st}$ half of the bits of $C$ (w. same IV).

Since $C' \neq C$, adv is allowed to query Dec$_k$ w. $C'$, which gives $1^{st}$ half bits of $M_b$, revealing $b$.

How do we design CCA-secure schemes?

1. Construct a scheme that is only CPA secure
2. Add authentication.

Message Authentication Code (MAC)

Provides integrity (authenticity), not confidentiality.
Alice \[ M, \text{MAC}_k(M) \rightarrow \text{Bob} \]

Bob recomputes \( \text{MAC}_k(M) \), and verifies that it agrees with what he received. If not reject the msg.

- Allows Bob to verify that \( M \) originated from Alice, and arrived unmodified.
- Alice & Bob need to share a secret key.
- Orthogonal to confidentiality, typically we do both (encrypt & append MAC on the ciphertext for integrity).

Security for MAC:

Goal: Security against adaptive chosen msg attack:

Adversary \( \text{Adv} \) is given pairs \( (M_i, \text{MAC}_k(M_i)) \) to msg \( M_i \) of his choice, and cannot generate any new \( M^* \) w. a valid \( \text{MAC}_k(M^*) \).

* Similar to signatures, but in the symmetric key setting.
Note: If MAC has $t$ bits, then Adv can guess w.p. $2^{-t}$. Therefore $t$ needs to be large enough.

\[ \text{Thm:} \quad \text{CPA-secure encryption} + \text{secure MAC} \Rightarrow \text{CCA-secure encryption scheme} \]

Intuitively, adding a MAC to the ciphertexts makes the decryption oracle useless to the adversary.

**How to construct a MAC:**

1. From hash functions (HMAC)
2. From block ciphers (CBC-MAC or CMAC)

**MAC from block ciphers**

\[ \text{1st attempt:} \quad \text{CBC-MAC}_k(M): \text{Encrypt } M \text{ w. CBC mode} \]

\[ \text{IV} = 0 \rightarrow M_1 \rightarrow \oplus \rightarrow C_1 \]

\[ \text{M_2} \rightarrow \oplus \rightarrow C_2 \]

\[ \text{M_3} \rightarrow \oplus \rightarrow C_3 \]

...
Insecure!

Given single block msg $M_1$ & tag $T_1 = E_k(M_1)$
and single block msg $M_2$ & tag $T_2 = E_k(M_2)$

$T_2$ is tag of $M_1 \parallel M_2 \oplus T_1$

**The Fix:** Process last block differently:
- All blocks use key $k_1$ and last block uses key $k_2$.

$$\text{CMAC}$$

**Thm:** CMAC is a secure MAC, if $E_k$ is an ideal cipher.

* Why does changing the key used in the last block fix security?*

* Why is it important to use fixed IV?*
Descr. [crypto 2000]:

Succinct & efficient CCA secure enc. scheme

(UFE : Unbalanced Feistel Encryption)

\[ M = (m_1, \ldots, m_n) \] sequence of blocks (length b)
\[ K = (K_1, K_2, K_3) \] three independent keys for the block cipher.

\[ \text{Enc}_K(M) : \]

1. Compute \((r, c_1, \ldots, c_n)\) using CTR mode with secret key \(K_1\):
   \[ r \leftarrow \{0, 1\}^b \]
   \[ x_i = E_{K_1}(r \oplus i), \quad i \in [n] \]
   \[ c_i = m_i \oplus x_i \]

2. Compute CMAC of \((c_1, \ldots, c_n)\) w.r.t. secret keys \(K_2, K_3\):

   \[ z_0 = 0^b \]
   \[ z_i = E_{K_2}(c_i \oplus z_{i-1}), \quad i \in [n-1] \]
   \[ z_n = E_{K_3}(c_n \oplus z_{n-1}) \leftarrow \text{last block uses } K_3 \]

3. Let \(\bar{c} = r \oplus z_n\)
   Output \((c_1, \ldots, c_n, \bar{c})\)
Encryption can be done in a single pass over the data ("online" property).

Decryption requires two passes:
- First to compute $Z_n$ (CMAC of $C=(C_1,\ldots,C_n)$).
- Compute $\Gamma = \Gamma \oplus Z_n$
- Decrypt $(\Gamma, C_1,\ldots,C_n)$ to get $M$.

* Provides CCA security

Does not provide authenticity

* Length of ciphertext $|C,\Gamma| = |M| + b$

single block