CSE107: Intro to Modern Cryptography

https://cseweb.ucsd.edu/classes/sp22/cse107-a/

Emmanuel Thomé

April 12, 2022

Lecture 5

Symmetric encryption: modes of operation

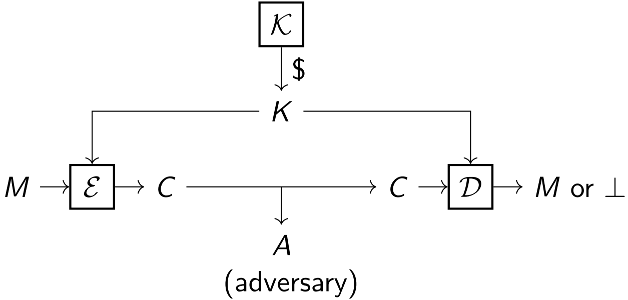
The mode of operation that must never be used: ECB

Randomized encryption, CBC$ and CTR$

Security analysis

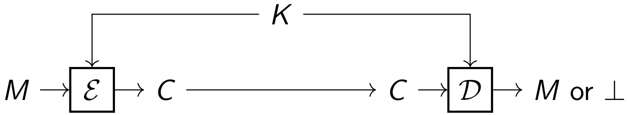
Symmetric Encryption Syntax

A symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ consists of three algorithms:



- $\mathcal{K}$ is the key generation algorithm.
- $\mathcal{E}$ is the encryption algorithm; may be randomized.
- $\mathcal{D}$ is the decryption algorithm; must be deterministic.

Correct decryption requirement



For all K, M we have

$$\mathcal{D}_K(\mathcal{E}_K(M)) = M$$

More formally: For all keys K that may be output by $\mathcal{K}$, and for all M in the *message space*, we have

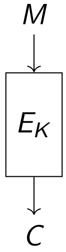
$$Pr[\mathcal{D}_K(\mathcal{E}_K(M)) = M] = 1\ ,$$

where the probability is over the coins of $\mathcal{E}$.

A scheme will usually specify an associated message space.

Modes of operation

Block cipher provides parties sharing K with



which enables them to encrypt a 1-block message.

How do we encrypt a long message using a primitive that only applies to n-bit blocks?
The mechanisms that we use to do that are called modes of operation.

Modes of operation

Modes of operation: our goal

$E : \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^\ell$ a family of functions

Usually a block cipher, in which case $\ell = n$.

From this fixed-length primitive, we want to build a symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ that can encrypt arbitrary length messages.

Behold the difference between E_K and $\mathcal{E}_K$.

Notation: $x[i]$ is the i-th block of a string x, so that $x = x[1] \ldots x[m]$. All blocks are n-bits long, except the last one which may be shorter.

Always:

```
Alg $\mathcal{K}$
$K \xleftarrow{\$} \{0,1\}^k$
return K
```

Plan

The mode of operation that must never be used: ECB

Randomized encryption, CBC$ and CTR$

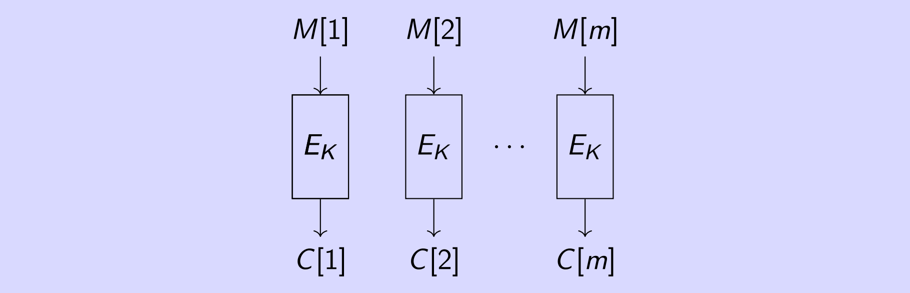
Security analysis

ECB: Electronic Codebook Mode

$\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ where:

```
Alg $\mathcal{E}_K(M)$
for $i = 1, \ldots, m$ do
    $C[i] \leftarrow E_K(M[i])$
return C
```

```
Alg $\mathcal{D}_K(C)$
for $i = 1, \ldots, m$ do
    $M[i] \leftarrow E_K^{-1}(C[i])$
return M
```

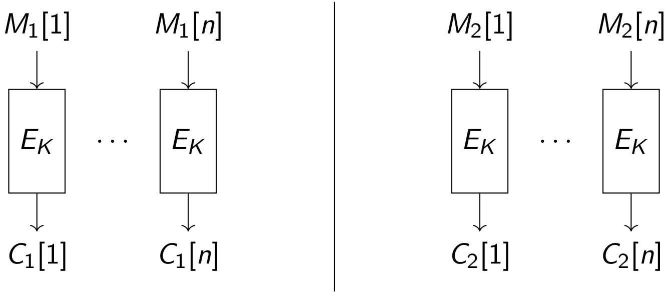


Correct decryption relies on E being a block cipher, so that E_K is invertible.

(In)Security of ECB

Weakness: $M_1 = M_2 \Rightarrow C_1 = C_2$

Why is the above true? Because E_K is deterministic:



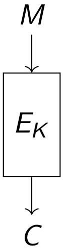
Why does this matter?

(In)Security of ECB

Suppose we know that there are only two possible messages, $Y = 1^n$ and $N = 0^n$, for example representing

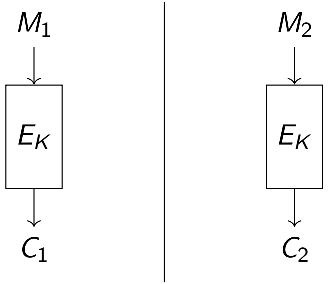
- FIRE or DON'T FIRE a missile
- BUY or SELL a stock
- Vote YES or NO

Then ECB algorithm will be $\mathcal{E}_K(M) = E_K(M)$.



(In)Security of ECB

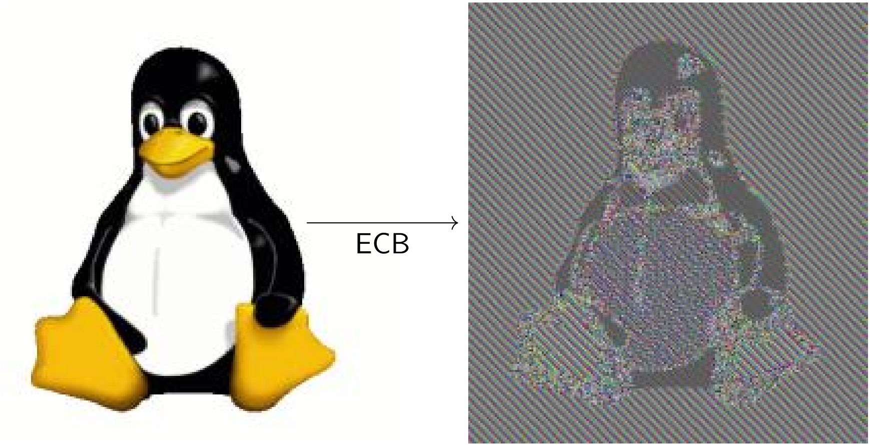
Votes $M_1, M_2 \in \{Y, N\}$ are ECB encrypted and adversary sees ciphertexts $C_1 = E_K(M_1)$ and $C_2 = E_K(M_2)$



Adversary may have cast the first vote and thus knows M_1; say $M_1 = Y$. Then adversary can figure out M_2:

- If $C_2 = C_1$ then M_2 must be Y
- Else M_2 must be N

The ECB Penguin



ECB

Is this avoidable?

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be ANY encryption scheme.

Suppose $M_1, M_2 \in \{Y, N\}$ and

- Sender sends ciphertexts $C_1 \leftarrow \mathcal{E}_K(M_1)$ and $C_2 \leftarrow \mathcal{E}_K(M_2)$
- Adversary A knows that $M_1 = Y$

Adversary says: If $C_2 = C_1$ then M_2 must be Y else it must be N.

Does this attack work?

Is this avoidable?

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be ANY encryption scheme.

Suppose $M_1, M_2 \in \{Y, N\}$ and

- Sender sends ciphertexts $C_1 \leftarrow \mathcal{E}_K(M_1)$ and $C_2 \leftarrow \mathcal{E}_K(M_2)$
- Adversary A knows that $M_1 = Y$

Adversary says: If $C_2 = C_1$ then M_2 must be Y else it must be N.

Does this attack work?

Yes, if $\mathcal{E}$ is deterministic. Deterministic encryption is bad. Really bad.

ECB is always a bad idea

Never, ever use ECB. Never. You will go to hell if you do.

Use of ECB has been the cause of multiple security breaches over the years, in various contexts. ECB should never have been used in the first place.

Plan

The mode of operation that must never be used: ECB

Randomized encryption, CBC$ and CTR$

Security analysis

Randomized encryption

For encryption to be secure it must be randomized

That is, algorithm $\mathcal{E}_K$ flips coins.

If the same message is encrypted twice, we are likely to get back different answers. That is, if $M_1 = M_2$ and we let

$$C_1 \xleftarrow{\$} \mathcal{E}_K(M_1) \text{ and } C_2 \xleftarrow{\$} \mathcal{E}_K(M_2)$$

then

$$Pr[C_1 = C_2]$$

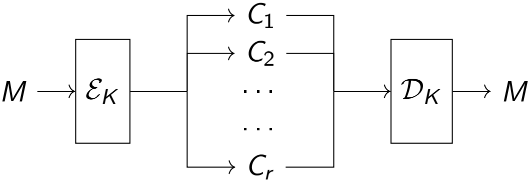
will (should) be small, where the probability is over the coins of $\mathcal{E}$.

Randomized encryption

There are many possible ciphertexts corresponding to each message.

If so, how can we decrypt?

We will see examples soon.



Randomized encryption

A fundamental departure from classical and conventional notions of encryption.

Clasically, encryption (e.g., substitution cipher) is a code, associating to each message a unique ciphertext.

Now, we are saying no such code is secure, and we look to encryption mechanisms which associate to each message a number of different possible ciphertexts.

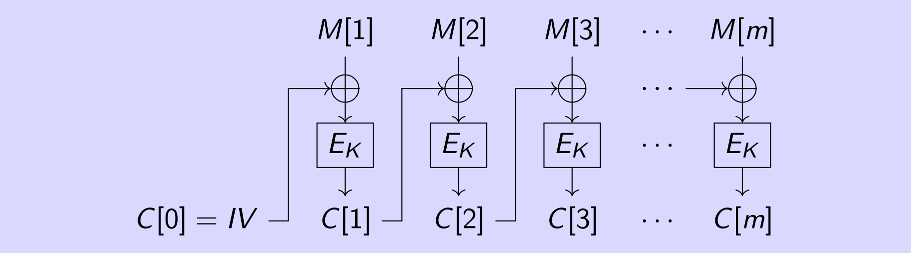
Note: the block cipher primitive E_K is deterministic. That's fine, as long as we don't use it raw.
The mode of operation defining $\mathcal{SE}$ has the opportunity to introduce randomness.

CBC$: Cipher Block Chaining with random IV mode

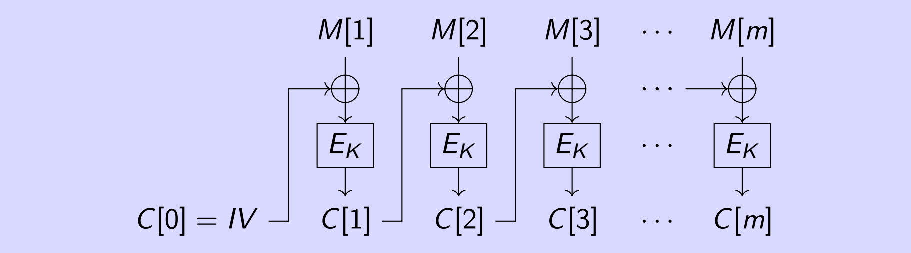
$\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ where:

```
Alg $\mathcal{E}_K(M)$
$C[0] \xleftarrow{\$} \{0,1\}^n$
for $i = 1, \ldots, m$ do
    $C[i] \leftarrow E_K(M[i] \oplus C[i-1])$
return $C$
```

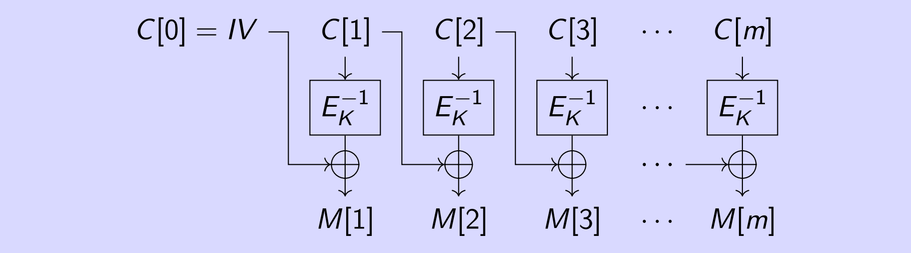
```
Alg $\mathcal{D}_K(C)$
for $i = 1, \ldots, m$ do
    $M[i] \leftarrow E_K^{-1}(C[i]) \oplus C[i-1]$
return $M$
```



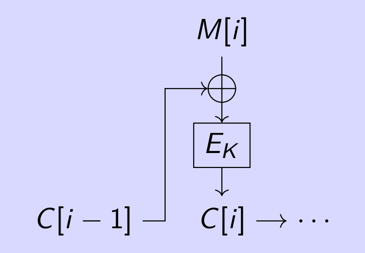
CBC$ decryption

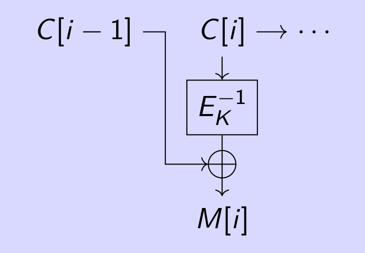


Correct decryption relies on E being a block cipher, so that E_K is invertible.



CBC$ implementation





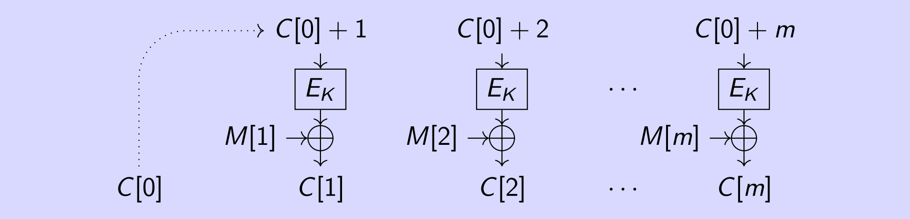
- CBC$ encryption is inherently impossible to parallelize, because of the chaining.
- CBC$ decryption is parallelizable.

CTR$ mode

If $X \in \{0,1\}^n$ and $i \in \mathbb{N}$ then $X + i$ denotes the n-bit string formed by converting X to an integer, adding i modulo 2^n, and converting the result back to an n-bit string. Below the message is a sequence of ℓ-bit blocks:

```
Alg $\mathcal{E}_K(M)$
$C[0] \xleftarrow{\$} \{0,1\}^n$
for $i = 1, \ldots, m$ do
    $P[i] \leftarrow E_K(C[0] + i)$
    $C[i] \leftarrow P[i] \oplus M[i]$
return $C$
```

```
Alg $\mathcal{D}_K(C)$
for $i = 1, \ldots, m$ do
    $P[i] \leftarrow E_K(C[0] + i)$
    $M[i] \leftarrow P[i] \oplus C[i]$
return $M$
```



CTR\$ mode

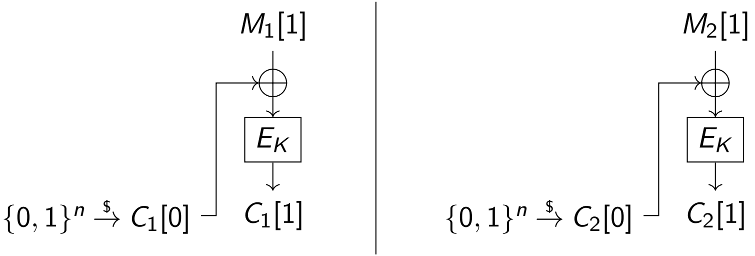
```
Alg $\mathcal{E}_K(M)$
$C[0] \xleftarrow{\$} \{0,1\}^n$
for $i = 1, \ldots, m$ do
    $P[i] \leftarrow E_K(C[0] + i)$
    $C[i] \leftarrow P[i] \oplus M[i]$
return $C$
```

```
Alg $\mathcal{D}_K(C)$
for $i = 1, \ldots, m$ do
    $P[i] \leftarrow E_K(C[0] + i)$
    $M[i] \leftarrow P[i] \oplus C[i]$
return M
```

- $\mathcal{D}$ does not use E_K^{-1}!
  CTR\$ can use a family of functions E that is not required to be a block cipher.
- Encryption and Decryption are parallelizable.

Voting with CBC$

Suppose we encrypt $M_1, M_2 \in \{Y, N\}$ (a 1-block message) with CBC$.


$M_1[1]$
E_K
$\{0,1\}^n \xrightarrow{\$} C_1[0]$
$C_1[1]$
$M_2[1]$
E_K
$\{0,1\}^n \xrightarrow{\$} C_2[0]$
$C_2[1]$

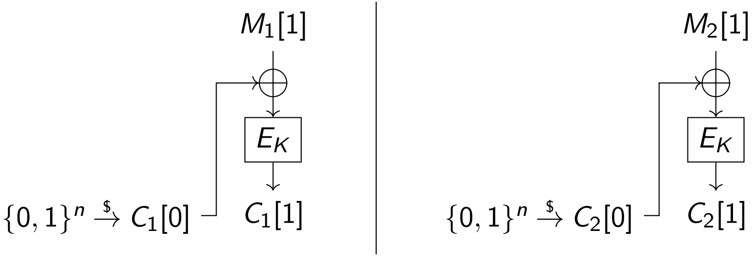
Adversary A sees $C_1 = C_1[0]C_1[1]$ and $C_2 = C_2[0]C_2[1]$.

Suppose A knows that $M_1 = M_1[1] = Y$.

Can A determine whether $M_2 = Y$ or $M_2 = N$?

Voting with CBC$

Suppose we encrypt $M_1, M_2 \in \{Y, N\}$ (a 1-block message) with CBC$.



$M_1[1]$
E_K
$\{0,1\}^n \xrightarrow{\$} C_1[0]$
$C_1[1]$
$M_2[1]$
E_K
$\{0,1\}^n \xrightarrow{\$} C_2[0]$
$C_2[1]$

Adversary A sees $C_1 = C_1[0]C_1[1]$ and $C_2 = C_2[0]C_2[1]$.

Suppose A knows that $M_1 = M_1[1] = Y$.

Can A determine whether $M_2 = Y$ or $M_2 = N$?

NO!

Assessing security

So CBC$ is better than ECB. But is it secure?

CBC$ is widely used so knowing whether it is secure is important

To answer this we first need to decide and formalize what we mean by secure.

Plan

The mode of operation that must never be used: ECB

Randomized encryption, CBC$ and CTR$

Security analysis

Security requirements

Suppose sender computes

$$C_1 \xleftarrow{\$} \mathcal{E}_K(M_1)\,;\ \cdots\ ;\ C_q \xleftarrow{\$} \mathcal{E}_K(M_q)$$

Adversary A has $C_1, \ldots, C_q$

What if A	
Retrieves K	Bad!
Retrieves M_1	Bad!

But also we want to hide all partial information about the data stream, such as

- Does $M_1 = M_2$?
- What is first bit of M_1?
- What is XOR of first bits of M_1, M_2?

Something we won't hide: the length of the message

What we seek

We want a single “master” property MP of an encryption scheme such that

- MP can be easily specified
- We can evaluate whether a scheme meets it
- MP implies ALL the security conditions we want: it guarantees that a ciphertext reveals NO partial information about the plaintext.

Intuition for definition of IND-CPA

The master property MP is called IND-CPA (indistinguishability under chosen plaintext attack).

Consider encrypting one of two possible message streams, either

$$M_0^1, ..., M_0^q$$

or

$$M_1^1, ..., M_1^q \ ,$$

where $|M_0^i| = |M_1^i|$ for all $1 \le i \le q$. Adversary, given ciphertexts $C^1, \ldots,$ C^q and both data streams, has to figure out which of the two streams was encrypted.

We will even let the adversary pick the messages: It picks (M_0^1, M_1^1) and gets back C^1, then picks (M_0^2, M_1^2) and gets back C^2, and so on.

Games for ind-cpa-advantage of an adversary A

Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme

Game $\mathrm{Left}_{\mathcal{SE}}$

```
procedure Initialize
$K \xleftarrow{\$}$ Keys
procedure LR($M_0, M_1$)
return $\mathcal{E}_K(M_0)$
```

Game $\mathrm{Right}_{\mathcal{SE}}$

```
procedure Initialize
$K \xleftarrow{\$}$ Keys
procedure LR($M_0, M_1$)
return $\mathcal{E}_K(M_1)$
```

Associated to $\mathcal{SE}, A$ are the probabilities

$$\Pr\left[\mathrm{Left}_{\mathcal{SE}}^{A} \Rightarrow 1\right] \qquad \Big| \qquad \Pr\left[\mathrm{Right}_{\mathcal{SE}}^{A} \Rightarrow 1\right]$$

that A outputs 1 in each world. The (ind-cpa) advantage of A is

$$\mathbf{Adv}_{\mathcal{SE}}^{\mathrm{ind\text{-}cpa}}(A) = \Pr\left[\mathrm{Right}_{\mathcal{SE}}^{A} \Rightarrow 1\right] - \Pr\left[\mathrm{Left}_{\mathcal{SE}}^{A} \Rightarrow 1\right]$$

Message length restriction

It is required that $|M_0| = |M_1|$ in any query M_0, M_1 that A makes to **LR**. An adversary A violating this condition is considered invalid.

This reflects that encryption is not aiming to hide the length of messages.

The measure of success

$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) \approx 1$ means A is doing well and $\mathcal{SE}$ is not ind-cpa-secure.

$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) \approx 0$ (or ≤ 0) means A is doing poorly and $\mathcal{SE}$ resists the attack A is mounting.

Adversary resources are its running time t and the number q of its oracle queries, the latter representing the number of messages encrypted.

Security: $\mathcal{SE}$ is IND-CPA-secure if $\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A)$ is "small" for ALL A that use "practical" amounts of resources.

Insecurity: $\mathcal{SE}$ is not IND-CPA-secure if we can specify an explicit A that uses "few" resources yet achieves "high" ind-cpa-advantage.

ECB is not IND-CPA-secure

Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Recall that ECB mode defines a symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with

$$\mathcal{E}_K(M) = E_K(M[1])E_K(M[2]) \cdots E_K(M[m])$$

Can we design A so that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) = \Pr\left[\text{Right}^A_{\mathcal{SE}} \Rightarrow 1\right] - \Pr\left[\text{Left}^A_{\mathcal{SE}} \Rightarrow 1\right]$$

is close to 1?

ECB is not IND-CPA-secure

Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Recall that ECB mode defines a symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with

$$\mathcal{E}_K(M) = E_K(M[1])E_K(M[2]) \cdots E_K(M[m])$$

Can we design A so that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) = \Pr\left[\text{Right}^A_{\mathcal{SE}} \Rightarrow 1\right] - \Pr\left[\text{Left}^A_{\mathcal{SE}} \Rightarrow 1\right]$$

is close to 1?

Exploitable weakness of $\mathcal{SE}$: $M_1 = M_2$ implies $\mathcal{E}_K(M_1) = \mathcal{E}_K(M_2)$.

ECB is not IND-CPA-secure

Let $\mathcal{E}_K(M) = E_K(M[1]) \cdots E_K(M[m])$.

```
adversary $A$
$C_1 \leftarrow \mathbf{LR}(0^n, 0^n)$ ; $C_2 \leftarrow \mathbf{LR}(1^n, 0^n)$
if $C_1 = C_2$ then return 1 else return 0
```

ECB is not IND-CPA-secure

Let $\mathcal{E}_K(M) = E_K(M[1]) \cdots E_K(M[m])$.

```
adversary $A$
$C_1 \leftarrow \mathbf{LR}(0^n, 0^n)$ ; $C_2 \leftarrow \mathbf{LR}(1^n, 0^n)$
if $C_1 = C_2$ then return 1 else return 0
```

```
Game $\mathrm{Right}_{\mathcal{SE}}$
procedure Initialize
    $K \xleftarrow{\$} \mathrm{Keys}$
procedure LR($M_0, M_1$)
    return $\mathcal{E}_K(M_1)$
```

Then

$$\Pr\left[\mathrm{Right}_{\mathcal{SE}}^{A} \Rightarrow 1\right] =$$

ECB is not IND-CPA-secure

Let $\mathcal{E}_K(M) = E_K(M[1]) \cdots E_K(M[m])$.

```
adversary A
$C_1 \leftarrow \mathbf{LR}(0^n, 0^n)$ ; $C_2 \leftarrow \mathbf{LR}(1^n, 0^n)$
if $C_1 = C_2$ then return 1 else return 0
```

Game $\mathrm{Right}_{\mathcal{SE}}$

```
procedure Initialize
$K \xleftarrow{\$} \mathrm{Keys}$

procedure LR($M_0, M_1$)
return $\mathcal{E}_K(M_1)$
```

Then

$$\Pr\left[\mathrm{Right}_{\mathcal{SE}}^{A} \Rightarrow 1\right] = 1$$

because $C_1 = E_K(0^n)$ and $C_2 = E_K(0^n)$.

ECB is not IND-CPA-secure

Let $\mathcal{E}_K(M) = E_K(M[1]) \cdots E_K(M[m])$.

```
adversary $A$
$C_1 \leftarrow \textbf{LR}(0^n, 0^n)$ ; $C_2 \leftarrow \textbf{LR}(1^n, 0^n)$
if $C_1 = C_2$ then return 1 else return 0
```

```
Game $\text{Left}_{\mathcal{SE}}$
procedure Initialize
    $K \xleftarrow{\$} \text{Keys}$
procedure LR($M_0, M_1$)
    return $\mathcal{E}_K(M_0)$
```

Then

$$\Pr\left[\text{Left}_{\mathcal{SE}}^{A} \Rightarrow 1\right] =$$

ECB is not IND-CPA-secure

Let $\mathcal{E}_K(M) = E_K(M[1]) \cdots E_K(M[m])$.

```
adversary A
$C_1 \leftarrow \textbf{LR}(0^n, 0^n)$ ; $C_2 \leftarrow \textbf{LR}(1^n, 0^n)$
if $C_1 = C_2$ then return 1 else return 0
```

```
Game $\text{Left}_{\mathcal{SE}}$

procedure Initialize
$K \xleftarrow{\$} \text{Keys}$

procedure $\textbf{LR}(M_0, M_1)$
return $\mathcal{E}_K(M_0)$
```

Then

$$\Pr\left[\text{Left}_{\mathcal{SE}}^{A} \Rightarrow 1\right] = 0$$

because $C_1 = E_K(0^n) \neq E_K(1^n) = C_2$.

ECB is not IND-CPA secure

```
**adversary** $A$
$C_1 \leftarrow \mathbf{LR}(0^n, 0^n)\,;\; C_2 \leftarrow \mathbf{LR}(1^n, 0^n)$
if $C_1 = C_2$ then return 1 else return 0
```

$$\begin{aligned}\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) &= \overbrace{\Pr\left[\text{Right}^{A}_{\mathcal{SE}} = 1\right]}^{1} - \overbrace{\Pr\left[\text{Left}^{A}_{\mathcal{SE}} = 1\right]}^{0} \\ &= 1\end{aligned}$$

And A is very efficient, making only two queries.

Thus ECB is **not** IND-CPA secure.

Why is IND-CPA the "master" property?

We claim that if encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is IND-CPA secure then the ciphertext hides ALL partial information about the plaintext.

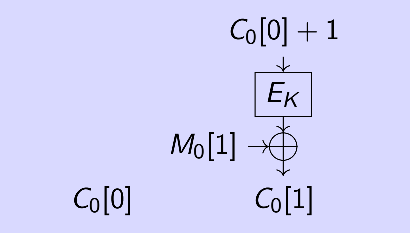
For example, from $C_1 \xleftarrow{\$} \mathcal{E}_K(M_1)$ and $C_2 \xleftarrow{\$} \mathcal{E}_K(M_2)$ the adversary cannot

- get M_1
- get 1st bit of M_1
- get XOR of the 1st bits of M_1, M_2
- etc.

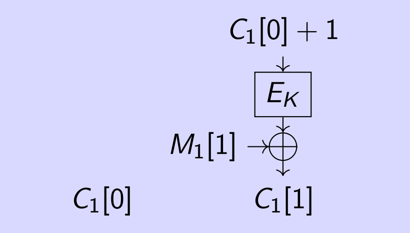
Birthday attack on CTR$

Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^\ell$ be a family of functions and $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ the corresponding CTR$ symmetric encryption scheme.
Suppose 1-block messages M_0, M_1 are encrypted:

$$C_0[0]C_0[1] \xleftarrow{\$} \mathcal{E}(K, M_0)$$



$$C_1[0]C_1[1] \xleftarrow{\$} \mathcal{E}(K, M_1)$$



Let us say we are **lucky** If $C_0[0] = C_1[0]$. If so:

$$C_0[1] = C_1[1] \text{ if and only if } M_0 = M_1$$

So if we are lucky we can detect message equality and violate IND-CPA.

Birthday attack on CTR$

Let $1 \leq q < 2^n$ be a parameter and let $\langle i \rangle$ be integer i encoded as an ℓ-bit string.

```
**adversary** $A$
for $i = 1, ..., q$ do
    $C^i[0]C^i[1] \xleftarrow{\$} \mathbf{LR}(\langle i \rangle, \langle 0 \rangle)$
$S \leftarrow \{(j, t)\colon C^j[0] = C^t[0]$ and $j < t\}$
If $S \neq \emptyset$, then
    $(j, t) \xleftarrow{\$} S$
    If $C^j[1] = C^t[1]$ then return 1
return 0
```

Birthday attack on CTR\$: Right game analysis

```
adversary A
for $i = 1, ..., q$ do
    $C^i[0]C^i[1] \xleftarrow{\$} \mathbf{LR}(\langle i \rangle, \langle 0 \rangle)$
$S \leftarrow \{(j, t) : C^j[0] = C^t[0]$ and $j < t\}$
If $S \neq \emptyset$, then
    $(j, t) \xleftarrow{\$} S$
    If $C^j[1] = C^t[1]$ then return 1
return 0
```

Game $\mathrm{Right}_{\mathcal{SE}}$

```
procedure Initialize
$K \xleftarrow{\$}$ Keys

procedure LR$(M_0, M_1)$
$C[0] \xleftarrow{\$} \{0,1\}^n$
$C[1] \leftarrow E(K, C[0] + 1) \oplus M_1$
return $C[0]C[1]$
```

If $C^j[0] = C^t[0]$ (lucky) then

$$C^j[1] = \langle 0 \rangle \oplus E_K(C^j[0] + 1) = \langle 0 \rangle \oplus E_K(C^t[0] + 1) = C^t[1]$$

so

$$\Pr\left[\mathrm{Right}_{\mathcal{SE}}^{A} \Rightarrow 1\right] = \Pr\left[S \neq \emptyset\right] = C(2^n, q)$$

Birthday attack on CTR\$: Left game analysis

```
adversary $A$
for $i = 1, \ldots, q$ do
    $C^i[0]C^i[1] \xleftarrow{\$} \mathbf{LR}(\langle i \rangle, \langle 0 \rangle)$
$S \leftarrow \{(j, t) \colon C^j[0] = C^t[0]$ and $j < t\}$
If $S \neq \emptyset$, then
    $(j, t) \xleftarrow{\$} S$
    If $C^j[1] = C^t[1]$ then return 1
return 0
```

Game $\mathrm{Left}_{\mathcal{SE}}$

```
procedure Initialize
$K \xleftarrow{\$}$ Keys

procedure $\mathbf{LR}(M_0, M_1)$
$C[0] \xleftarrow{\$} \{0,1\}^n$
$C[1] \leftarrow E(K, C[0] + 1) \oplus M_0$
return $C[0]C[1]$
```

If $C^j[0] = C^t[0]$ (lucky) then

$$C^j[1] = \langle j \rangle \oplus E_K(C^j[0] + 1) \neq \langle t \rangle \oplus E_K(C^t[0] + 1) = C^t[1]$$

so

$$\Pr\left[\mathrm{Left}^A_{\mathcal{SE}} \Rightarrow 1\right] = 0.$$

Birthday attack on CTR$

$$
\begin{aligned}
\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) &= \Pr\left[\text{Right}^{A}_{\mathcal{SE}} \Rightarrow 1\right] - \Pr\left[\text{Left}^{A}_{\mathcal{SE}} \Rightarrow 1\right] \\
&= C(2^n, q) - 0 \;\geq\; 0.3 \cdot \frac{q(q-1)}{2^n}
\end{aligned}
$$

Conclusion: CTR$ can be broken (in the IND-CPA sense) in about $2^{n/2}$ queries, where n is the block length of the underlying block cipher, regardless of the cryptanalytic strength of the block cipher.

Security of CTR$

So far: A q-query adversary can break CTR$ with advantage $\approx \frac{q^2}{2^{n+1}}$

Question: Is there any better attack?

Security of CTR$

So far: A q-query adversary can break CTR$ with advantage $\approx \frac{q^2}{2^{n+1}}$

Question: Is there any better attack?

Answer: NO!

We can prove that the best q-query attack short of breaking the block cipher has advantage at most

$$\frac{2(q-1)\sigma}{2^n}$$

where σ is the total number of blocks across all messages encrypted.

Example: If q 1-block messages are encrypted then $\sigma = q$ so the adversary advantage is not more than $2q^2/2^n$.

For $E =$ AES this means up to about 2^{64} blocks may be securely encrypted, which is good.

Security of CTR$

Theorem: security of CTR$ [BDJR97]

Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^\ell$ be a family of functions and $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ the corresponding CTR$ symmetric encryption scheme. Let A be an ind-cpa adversary against $\mathcal{SE}$ that has running time t and makes at most q **LR** queries, the messages across them totaling at most σ blocks. Then there is a prf-adversary B against E such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) \le 2 \cdot \mathbf{Adv}^{\text{prf}}_{E}(B) + \frac{2(q-1)\sigma}{2^n}$$

Furthermore, B makes at most σ oracle queries and has running time $t + \Theta(\sigma \cdot (n + \ell))$.

Intuition

We won't prove this, but let's give some intuition.

We assume for simplicity that both messages in each **LR** query of A are m blocks long. Thus $\sigma = mq$.

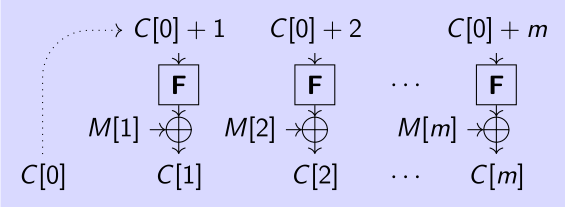
Note a block is ℓ bits, so each message in a query is $m\ell$ bits.

We let $C_i = C_i[0]C_i[1]\ldots C_i[m]$ denote the response of the **LR** oracle to A's i-th query.

Intuition for IND-CPA security of CTR$

Consider the CTR$ scheme with E_K replaced by a random function $\mathbf{F}$ with range $\{0,1\}^\ell$.

```
Alg $\mathcal{E}_{\mathbf{F}}(M)$
$C[0] \xleftarrow{\$} \{0,1\}^n$
for $i = 1, \ldots, m$ do
    $P[i] \leftarrow \mathbf{F}(C[0]+i)$
    $C[i] \leftarrow P[i] \oplus M[i]$
return $C$
```



Analyzing this is a thought experiment, but we can ask whether it is IND-CPA secure.

If so, the assumption that E is a PRF says CTR$ with E is IND-CPA secure.

CTR$ with a random function

Let W be the event that the points

$$C_1[0]+1,\ldots,C_1[0]+m,\ldots,C_q[0]+1,\ldots,C_q[0]+m\ ,$$

on which **F** is evaluated across the q encryptions, are all distinct.

Case 1: W happens. Then the encryption is a one-time-pad: ciphertexts are random, independent strings, regardless of which message is encrypted. So A has zero advantage.

Case 2: W doesn't happen. Then A may have high advantage but it does not matter because $\Pr[W]$ doesn't happen is small. (It is the small additive term in the theorem.)

Security of CBC$

Theorem: security of CBC$ [BDJR97]

Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ the corresponding CBC$ symmetric encryption scheme. Let A be an ind-cpa adversary against $\mathcal{SE}$ that has running time t and makes at most q **LR** queries, the messages across them totaling at most σ blocks. Then there is a prf-adversary B against E such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{SE}}(A) \leq 2 \cdot \mathbf{Adv}^{\text{prf}}_{E}(B) + \frac{\sigma^2}{2^n}$$

Furthermore, B makes at most σ oracle queries and has running time $t + \Theta(\sigma \cdot n)$.

CBC must be used with extreme caution

CBC mode is IND-CPA secure, but vulnerable both in theory and practice to chosen ciphertext attacks, which we will cover in future lectures.

Probably best to avoid using it because of the difficulty of implementing it securely.