Observe that a block cipher can be used to construct a PRG:
\[ F : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n \]
be a block cipher

Define \( G : \{0,1\}^n \rightarrow \{0,1\}^n \) as
\[ G(k) = F(k,1) \| F(k,2) \| \cdots \| F(k,l) \]

\[ \uparrow \]

String concatenation write input as an \( n \)-bit string (just require that \( n > \log q \))

Theorem. If \( F \) is a secure PRF, then \( G \) is a secure PRG.

Proof. As usual, we show the contrapositive: if \( G \) is not a secure PRG, then \( F \) is not a secure PRF.

Suppose we have efficient adversary \( A \) for \( G \). We use \( A \) to build adversary for \( F \):

\[ b \in \{0,1\}^n \]

Algo based on \( A \):

Adversary for breaking \( F \)

Algo \( F^{\text{ch}} \) challenger for \( F \)

1. If \( l = \text{poly} \), then \( B \) is efficient
2. If \( b = 0 \) \( : B \) sends \( G(k) \) to \( A \)
3. If \( b = 1 \) \( : B \) sends uniformly random string \( f \) to \( A \)

Adversary \( A \) does:

1. \( F(k) \rightarrow A \)
2. \( f \rightarrow A \)
3. \( A \) outputs \( b \)

By security of \( F \), we will get:

\[ \Pr[b = 1] \approx \frac{1}{2} \]

We said \( \text{PRP} \) above (just revised this).

But... we used a block cipher \( \text{PRP} \) in our construction above. Does the proof still go through?

Not quite...

- For a random function, \( f(z) = f(y) \) with probability \( \frac{1}{2^n} \)
- For a random permutation, \( f(z) = f(y) \) with probability 0

Adversary won't notice unless it sees a "collision" [i.e., two values \( x, y \) where \( f(x) = f(y) \)]

PRF Switching Lemma. Let \( F : K \times X \rightarrow X \) be a secure PRP. Then, for any \( Q \)-query adversary \( A \):

\[ \text{PRFAdv}[A,F] = \text{PRFAdv}[A,F] \leq \frac{Q^2}{2^{1/2}} \]

Proof Idea. Adversary essentially cannot tell the difference unless it sees a collision. If there is no collision, then it is just seeing random values. How many queries before there is a collision? Birthday paradox: \( Q \approx \sqrt{2^{32}} \)

Take-away: If \( 1 \times 11 \) is large (e.g., exponential), then we can use a PRP as a PRF.

- 3DES: \( n = 64 \) so \( 1 \times 11 = 2^{64} \) [if adversary makes \( < 2^{32} \) queries, then can use it as a PRF]
- AES: \( n = 128 \) so \( 1 \times 11 = 2^{128} \) [if adversary makes \( < 2^{64} \) queries, then can use it as a PRF]
Thus far: PRP/PRF in "counter mode" gives us a stream cipher (one-time encryption scheme)

How do we reuse it? Choose a random starting point (called an initialization vector)

"randomized counter mode"

<table>
<thead>
<tr>
<th>m_1</th>
<th>m_2</th>
<th>m_3</th>
<th>m_4</th>
</tr>
</thead>
</table>

divide message into blocks (based on block size of PRF)

<table>
<thead>
<tr>
<th>IV</th>
<th>F(k,IV)</th>
<th>F(k,IV)</th>
<th>F(k,IV)</th>
<th>F(k,IV)</th>
</tr>
</thead>
</table>

observe: ciphertext is longer than the message (required for CPA security)

Theorem: Let $F: \mathbb{K} \times \mathbb{K} \rightarrow \mathbb{Y}$ be a secure PRF and let $\mathcal{T}_{\text{ICM}}$ denote the randomized counter mode encryption scheme from above for $l$-block messages ($M = \chi^l$). Then, for all efficient CPA adversaries $A$, there exists an efficient PRF adversary $B$ such that

$$\text{CPAAdv}[A, \mathcal{T}_{\text{ICM}}] \leq \sqrt{\frac{4Q^2l}{1X^1}} + 3 \cdot \text{PRFAdv}[B, F]$$

$Q$: number of encryption queries

$l$: number of blocks in message

Intuition:
1. If there are no collisions (i.e., PRF never evaluated on the same block), then it is as if everything is encrypted under a fresh one-time pad.
2. Collision event: $(x, x + 1, \ldots, x + (l-1))$ overlaps with $(x', x' + 1, \ldots, x' + (l-1))$ when $x, x' \in \chi^l$

$$\Pr[\text{collision}] \leq \frac{2l Q^2}{1X^1}$$

There are $\leq Q^2$ possible pairs $(x, x')$, so by a union bound,

$$\Pr[\text{collision}] \leq \frac{2l Q^2}{1X^1}$$

3. Remaining factor of $2$ in advantage due to intermediate distribution (hybrid argument):

Encrypt $m$ with PRF $\rightarrow$ PRFAdv$[B, F] + \frac{2l Q^2}{1X^1}$

Encrypt $m_0$ with fresh one-time pad $\rightarrow 0$

Encrypt $m_1$ with fresh one-time pad $\rightarrow \text{PRFAdv}[B, F] + \frac{2l Q^2}{1X^1}$

Encrypt $m_0$ with PRF $\rightarrow \text{PRFAdv}[B, F] + \frac{2l Q^2}{1X^1}$

Interpretation: If $|X^1| = 2^{128}$ (e.g., AES), and messages are 1 MB long ($2^{16}$ blocks) and we want the distinguishing advantage to be below $2^{-32}$, then we can use the same key to encrypt

$$Q \leq \frac{\sqrt{1X^1} \cdot 2^{-32}}{4l} = \frac{2^{16}}{2^{18}} = \sqrt{2^{18}} = 2^{9} \approx 1 \text{ trillion messages!}$$
Counter mode is parallelizable, simple-to-implement, just requires PRF — preferred mode of using block ciphers

Other block cipher modes of operation:

- Cipherblock chaining (CBC): common mode in the past (e.g., TLS 1.0, still widely used today)

```
+---------+---------+---------+  +---------+---------+---------+
|         |         |         |  |         |         |         |
| Cipher  | Encryption | Decryption |  | Cipher  | Encryption | Decryption |
| Text    |           |            |  | Text    |           |            |
| IV      | m₁       | c₁        |  | IV      | c₁       | m₁       |
|         | F(k₁)    | F(k₁)     |  |         | F⁻¹(k₁)  | F⁻¹(k₁)  |
|         | c₂        | c₂        |  |         | m₂       | m₂       |
|         | F(k₂)    | F(k₂)     |  |         | F⁻¹(k₂)  | F⁻¹(k₂)  |
|         | c₃        | c₃        |  |         | m₃       | m₃       |
```

- Cipher text is indistinguishable from random string if PRP is evaluated on distinct inputs
- Collision probability ≤ \( \frac{2^m}{1 \cdot l} \)
  - this is larger than collision prob for randomized counter mode by a factor of \( \frac{1}{2} \) [overlap of \( Q \) random intervals vs. \( Q \) random points]

Interpretation: CBC mode provides weaker security compared to counter mode: 

\[
\frac{2^{64} \cdot 2^{12}}{\frac{1}{2} \cdot 2^{12}} = 2^{62} = 2^{31.5} \approx 1 \text{ billion messages}
\]

\( \Rightarrow 2^{75} \approx 180 \times \text{smaller than using counter mode} \)

Theorem: Let \( F : \mathbb{K} \times \mathbb{K} \rightarrow \mathbb{Y} \) be a secure PRP and let \( \text{Tc}_{\text{CBC}} \) denote the CBC encryption scheme for \( l \)-block messages \( (M = X \times \mathbb{K}) \). Then, for all efficient CPA adversaries \( A \), there exists an efficient PRF adversary \( B \) such that

\[
\text{CPAAdv}[A, \text{Tc}_{\text{CBC}}] \leq \frac{2^{2l}}{1 \cdot l} + 2 \cdot \text{PRFAdv}[B, F]
\]

\( Q \): number of encryption queries

\( l \): number of blocks in message
Padding in CBC mode: each ciphertext block is computed by feeding a message block into the PRP

\[ \Rightarrow \text{message must be an even multiple of the block size} \]

\[ \Rightarrow \text{when used in practice, need to pad messages} \]

Can we pad with zeroes? Cannot decrypt! What if original message ended with a bunch of zeroes?

Requirement: padding must be invertible.

CBC padding in TLS 1.0: if \( k \) bytes of padding is needed, then append \( k \) bytes to the end, with each byte set to \( k-1 \)

(for AES-CBC) if \( 0 \) bytes of padding is needed, then append a block of 16 bytes, with each byte equal to \( 15 \)

\[ \Rightarrow \text{dummy block needed to ensure pad is invertible} \]

\[ \Rightarrow \text{called PKCS#5/PKCS#7 (public-key cryptography standards)} \]

Need to pad in CBC encryption can be exploited in “padding oracle” attacks

Padding in CBC can be avoided using idea called “cipher text stealing” (as long as messages are more than 1 block)

Comparing CTR mode to CBC mode:

CTR mode

1. no padding needed (shorter ciphertexts)
2. parallelizable
3. only requires PRF (no need to invert)
4. tighter security
5. IVs have to be non-repeating (and spaced far apart)

CBC mode

1. padding needed
2. sequential
3. requires PRF
4. less tight security (re-key more often)
5. requires unpredictable IVs

\[ \text{easy to implement: IV = non-null counter} \]

\[ \text{only needs to be non-repeating (can be predictable)} \]

\[ \text{bottom-line: use randomized or nonce-based counter mode whenever possible: simpler, easier, and better than CBC!} \]

A tempting and bad way to use a block cipher: ECB mode (electronic codebook)

\[
\begin{align*}
\text{Encryption:} & \quad \text{Apply block cipher to each block of the message} \\
\text{Decryption:} & \quad \text{Invert each block of the ciphertext}
\end{align*}
\]

\[
\begin{align*}
m_1 & \quad \begin{array}{ccc} m_2 & m_3 \end{array} \\
F(k_1) & \quad \begin{array}{ccc} F(k_2) & F(k_3) \end{array} \\
C_1 & \quad \begin{array}{ccc} C_2 & C_3 \end{array}
\end{align*}
\]

\[ \text{Scheme is deterministic! Cannot be CPA secure!} \]

\[ \text{Not even semantically secure!} \]

\[ (m_0, m_0) \text{ vs. } (m_0, m_1) \text{ where } m, f(m) \]

\[ \text{cipher text blocks output are the same} \]

\[ \text{cipher text blocks output are different} \]

\[ \text{NEVER USE ECB MODE FOR ENCRYPTION!} \]