Lecture: Encryption Intro
Plan

- Recap: PRF
- CPA Security [Weak encryption]
- CPA-secure encryption from PRF: Counter mode
- Pseudorandom permutation

Logistics

* Pset 1 out tomorrow.
* ONLY collab w/ pset grp
* We will assign pset groups tonight.
Recap: Pseudorandomness

OWF: Easy to compute, hard to invert
PRG: Stretch short random seed into long pseudorandom string
PRF: A keyed fn $\mathcal{S}$ st. $\mathcal{S}(k, \cdot)$ "looks like" random fn when $k \in \mathcal{K}$

A PRF is an $\mathcal{S}$. fn

$\mathcal{S} : \mathcal{K} \times \{0, 1\}^n \rightarrow \{0, 1\}^n$

s.t. $\forall \mathsf{adv} \mathcal{A}$, $\exists \negl$ fn st.

$$|\Pr[\mathcal{A}(\cdot) = 1 : k \in \mathcal{K}] - \Pr[\mathcal{A}(\cdot) = 1 : R \in \mathcal{R}]| \leq \negl(n).$$

Q: Is PRF still pseudorandom if Adv gets 1 bit of key?
Counter Mode

PRF $\Rightarrow$ One-time (comp. sec.) enc w/ short key

PRF $f: \mathbb{Z}^n \rightarrow \{0,1\}^n$

$Enc(k, m) :=$ $m = (k, 0) \oplus f(k, 1) \oplus f(k, 2) \oplus f(k, 3) \cdots$

$Dec(k, c) := c \oplus [f(k, 0) \| \cdots ]$

Idea:
If adv can distinguish $Enc(k, m_0)$ from $Enc(k, m_1)$
Can break PRF
Weaknesses of the one-time pad
* Long key $\leftarrow$ PRF
* One ct per key $\leftarrow$ Today
* Adversary can tamper w/ ct $\leftarrow$ Next time

Goal: Enc scheme secure if adversary can see many msg encrypted with same key.

$\Rightarrow$ NO Integrity protection (next time)

Applications:
* File encryption
* Some Internet protocols

Our security design is going to consider strong adversary:
* Gets one of many msg of its choice [why?]
* Just has to dist enc of m, m', (chosen)

"IND-CPA security"

Historical example:
Give msg to embassy, ask to relay to home govt

$\Rightarrow$ Enc of chosen msg!
**CPA Security**

For an enc scheme \((\text{Enc}, \text{Dec})\) over \((K, M, C)\), define game:

\[
\begin{array}{c}
\text{Chal} \\
K \leftarrow \mathcal{K} \\
\end{array} \quad \begin{array}{c}
\text{Adv} \\
\end{array} \\
\begin{array}{c}
\quad m; \\
\quad c_i \leftarrow \text{Enc}(k, m_i) \\
\quad (m_0, m_1) \leftarrow \mathsf{Rand} \\
\quad C^* \leftarrow \text{Enc}(k, m_b) \\
\end{array}
\]

Let \(W_b\) = output of game \(b\).

We say \((\text{Enc}, \text{Dec})\) is CPA-secure if

\[\forall \text{ eff adv } \mathcal{A} : \exists \text{ negl fn st.} \]

\[|\Pr[W_0] - \Pr[W_1]| \leq \text{negl}.\]

If we want to be fully precise, parameterize everything by security parameter \(\lambda\) or \(\lambda^*\).

Weak: What if adv can't see decryption of chosen ct?

Tamper msg?
CPA-Secure Enc must be randomized?

Intuition:
* Think about SSH—encryption of 8-bit chars.
  
  \[
  \text{pass} \quad \text{pass} \quad \text{pass} \quad 0 \quad 0 \quad 1
  \]

\[\Rightarrow\text{Need to defeat freq attack}\]

Concretely, show attack in CPA game.
* Even WEAK encryption requires randomness?

\[\Rightarrow\text{Obvious? Or very non-obvious?}\]

(GM'84)
CPA-Secure Enc from PRF.

* Let \((\text{Enc}, \text{Dec})\) over \(\mathcal{K}, \mathcal{M}, \mathcal{E}\) be a one-time (perfectly) secure enc scheme.

* Let \(f: \mathcal{K} \times \{0,1\}^n \rightarrow \mathcal{K}\) be a PRF

Then

\[
\text{Enc}'(k', m) := \\
\begin{align*}
&x \leftarrow \{0,1\}^n \\
k' \leftarrow f(k, x) \\
&\text{output } (x, \text{Enc}(k, m))
\end{align*}
\]

\[
\text{Dec}'(k', (x, c)) := \\
\begin{align*}
k' &\leftarrow f(k, x) \\
&\text{output } \text{Dec}(k', c)
\end{align*}
\]

Show instantiation w/ one-time pad.

\(\Rightarrow\) Still malleable!

Adv breaking \(\text{Enc}'\) breaks either \(\text{Enc}\) or PRF \(S\).

(See Boneh-Shoup Thm 5.2)

**Thm:** For all CPA adv \(A\) making \(Q\) CPA queries,

\[
\text{CPAadv}[A, \text{Enc}'] \leq \frac{Q^2}{2^n} + 2 \cdot \text{PerAdv}[B, S]
\]
**PRP (“Block cipher”)**

* Used to be dominant, ... less so now

\[
P: \mathcal{X} \times \{0,1\}^n \to \{0,1\}^n
\]

\[
P^{-1}: \mathcal{X} \times \{0,1\}^n \to \{0,1\}^n
\]

**Correctness:***  \( \forall k \in \mathcal{X} \ \forall x \in \{0,1\}^n \)

\[
P^{-1}(k, P(k,x)) = x
\]

**Pseudorandomness**

Same as PRF except that adv gets oracle access to \( P(k, \cdot) \), \( P^{-1}(k, \cdot) \). Can't dist from \( T \{ \cdot \}, T^{-1} \{ \cdot \} \) for \( k \in \mathcal{X}, T \triangleleft \text{Perms}[\{0,1\}^n] \)

* People thought you needed to “encrypt” & “decrypt” less PRF-based constructions simpler, faster (many core)

Still, important NIST-standardized ciphers are PRBs:

- DES (1975) \( \quad |k| = 2^{56} \quad n = 64 \)
- 3DES \( \quad |k| = 2^{168} \quad n = 64 \)
- AES (1998) \( \quad |k| \in \{2^{128}, 2^{96}, 2^{64}\} \quad n = 128 \)

\text{N.B. DES key size is far too small.}  

in U.S. \text{SECRET}: AES-128/192/256 ? Algs are \text{public};

Top SECRET: AES-192/256 \text{TOP SECRET: AES-192/256}
Things to know about PRPs

- **NEVER** use directly to encrypt ("ECB mode")

- CPUs have HW support for AES (GBs per second)
  
  (AES-NI)

- Can use as a PRF, as long as you don't use too much

```
"PRF Switching Lemma" (See Boreh-Shoup)

Let \( P : \mathbb{Z}_2 \times \mathcal{S}_\mathbb{Z} \rightarrow \{0,1\} \) be a PRF.

Then for any PRP adv \( \mathcal{A}_{\text{PRP}} \), \( \exists \) PRF adv \( \mathcal{A}_{\text{PRF}} \) s.t.

\[
|\mathcal{A}_{\text{PRP}} - \mathcal{A}_{\text{PRF}}| \leq \frac{q^2}{2^n+1}.
\]

Intuition:

* Collisions in outputs is only diff b/w PRF & PRP
* Until \( q \leq 2^{n/2} \), will not expect to see collisions
  by Birthday paradox
* After that, can distinguish

→ Very common to use AES in counter mode ("AES-CBC")

What about 3DES???