Crypto Class Notes, Feb 3

Mark Lindberg

π: CPA Secure: $\forall$ PPT A, "Advantage" of adversary is negligible.
More formally, $Pr(Priv\ K^{CPA}_{A,\pi} = 1) \leq \frac{1}{2} + NEGL(n)$.
Where the Priv K is the adversary guessing b correctly.
Today, EAV-secure $\leftrightarrow$ PRG. Pseudo-Random Generator.
CPA-secure $\leftrightarrow$ PRF. Psedo-Random Function.
If one-way functions exist, PRFs exist, and CPA-secure encryptions do exist.
What is a PRF? It is a "keyed" function.
Generally written as $F(\underbrace{k}_{\text{Key}}, \underbrace{x}_{\text{Input}})$, also written as $F_k(x)$.

For security parameter m, the key, input, and output lengths are specified by 3 polynomials, ℓ_{key}, ℓ_{in}, and ℓ_{out} respectively. The key is $\ell_{key}(m)$ bits long, the input is $\ell_{in}(m)$ bits long, and the output is $\ell_{out}(m)$ bits long.

In the text, "length-preserving" means all results from the polynomials are equal to m. The text focuses on this special case. We will make this assumption for this lecture as well. In the last question on the homework, we look at PRFs, and the definition will need to be generalized. (Given in HW.)

DEFINITION: PRF (Pseudo-Random Function?): Let F be a length-preserving, keyed, efficient (polynomial time computable) function. F is a PRF if for all PPT (Probabilistic Polynomial Time) distinguishers D, $|\underbrace{Pr(D^{F_k}(1^n) - 1)}_{\text{Random choice of } k} - \underbrace{Pr(D^f(1^m))}_{\substack{\text{Random chance of } f \in Func_m \\ \text{randomness in } D.}}| \leq NEGL(n)$ (We're considering F_k and f as oracles that return the particular n-bit output string for that input string. In constant time.) where $Func_m$ is the set of all functions from m by strings to m-bit strings.

We can think of representing an $f \in Func_m$ as a table. $m \cdot 2^m$ bits. Then $|Func_m| = 2^{m \cdot 2^m}$.

Now we will present a scheme that is CPA secure.

CPA-secure ENC-scheme $\pi = (Gen, Enc, Dec)$ (construction 3.30?).

Assume F is a PRF. (Caution for last HW problem: We're assuming length-preserving here, but not true for last problem.) Then $Gen(1^n)$ produces a random n-bit key k. Then $Enc_k(m)$ returns a pair $(\underbrace{?}_{n \text{ bits.}}, \underbrace{F_k(?) \oplus m}_{n \text{ bits.}})$. where ? is a random n-bit string.

$Dec_k(c)$, where k is n bits, and c is $2n$ bits. c is of the form $(?, s)$, each of which are n bits. $m = ?? \oplus F_k(?)$.

Theorem 3.31: π is CPA-secure.
Define $\widetilde{\pi} = (\widetilde{Gen}, \widetilde{Enc}, \widetilde{Dec})$.
$\widetilde{Gen}(1^n)$ returns a random f in $Func_n$. (f is our key, but also is a function.)
$\widetilde{Enc}_f(m)$ returns $(r, f(r) \oplus m)$
$\widetilde{Dec}_f(c)$, $c = r, s$, $m = f(r) \oplus s$.
Two parts to the proof:
Part 1: $|Pr(Priv\ K^{CPA}_{A,\pi} = 1) - Pr(Priv\ K^{CPA}_{A,\widetilde{\pi}} = 1)| \leq NEGL(m)$.
Construct a PPT distinguisher D for F.

- Simulate A on input 1^m.
- When A calls oracle on a message m
  - D generates random n-bit string r.
  - D calls its own oracle on r, getting y.
  - D returns to A $(r, y \oplus m)$.
- A determines its messages m_0 and m_1.
- Determine b.
- Use oracle call to get c.
- D outputs 1 iff A is correct.

Negligible implies negligible!
Need $\forall$ PPT A, $Pr(Priv\ K^{CPA}_{A,\pi} = 1) \leq \frac{1}{2} + NEGL(n)$.
By part 1, it is sufficient to prove that $Pr(Priv\ K^{CPA}_{A,\widetilde{\pi}} = 1) \leq \frac{1}{2} + NEGL(n)$.
Then negl + negl = negl, so yay! :)
Part 2: Intuition: Run the experiment, replace π with $\widetilde{\pi}$. It's in the text.