Cryptography

Lecture 7

Announcements

- HW3 up on course webpage, due Wednesday, 2/22

Agenda

- Last time:
  - SKE secure against eavesdroppers from PRG (K/L 3.3)
- This time:
  - Stream Ciphers
  - CPA Security (K/L 3.4)
  - Pseudorandom Functions (PRF) (K/L 3.5)

Correct usage of PRG's in practice

1. $s \xleftarrow{R} \{0,1\}^n \quad G(s) \in \{0,1\}^{\ell(n)}$

   $s' \xleftarrow{R} \{0,1\}^n \quad G(s') \in \{0,1\}^{\ell(n)}$

Stream Cipher

$\boxed{\text{keep state}}$

Sender

State s_i after sending the i-th message:

128

$$\boxed{s_0 := k}$$

$$s_{i+1} := G(s_i)_2, \ldots, G(s_i)_{n+1}$$ 129

$$pad_{i+1} := G(s_i)_1$$

$$c_{i+1} := m_{i+1} \oplus pad_{i+1} \longrightarrow$$

Receiver

State s_i after receiving the i-th message:

128

$$\boxed{s_0 := k}$$

$$s_{i+1} := G(s_i)_2, \ldots, G(s_i)_{n+1}$$ 129

$$pad_{i+1} := G(s_i)_1$$

$$m_{i+1} := c_{i+1} \oplus pad_{i+1}$$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK_{A,\Pi}^{cpa}(n)$

Adversary $A(1^n)$ Challenger

Chosen Plaintext Attack.

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK_{A,\Pi}^{cpa}(n)$

Adversary $A(1^n)$

Challenger

$k \leftarrow Gen(1^n)$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$

$A^{Enc_k(\cdot)}$

A gets oracle access to the Encryption alg.

Challenger

$k \leftarrow Gen(1^n)$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$

Challenger

$$A^{Enc_k(\cdot)} \xrightarrow{\quad m' \quad} \qquad k \leftarrow Gen(1^n)$$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$

Challenger

$k \leftarrow Gen(1^n)$

$A^{Enc_k(\cdot)}$

$\xrightarrow{m'}$

$\xleftarrow{c'}$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$

Challenger

$k \leftarrow Gen(1^n)$

$A^{Enc_k(\cdot)}$

m'

c'

$\vdots$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$

Challenger

$k \leftarrow Gen(1^n)$

$A^{Enc_k(\cdot)}$

$\xrightarrow{m'}$

$\xleftarrow{c'}$

$\vdots$

$\xrightarrow{m_0, m_1}$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$		Challenger
		$k \leftarrow Gen(1^n)$
$A^{Enc_k(\cdot)}$	$\xrightarrow{\quad m' \quad}$	
	$\xleftarrow{\quad c' \quad}$	
	$\vdots$	
	$\xrightarrow{\quad m_0, m_1 \quad}$	$b \leftarrow \{0,1\}$
	$\xleftarrow{\quad c \quad}$	$c \leftarrow Enc_k(m_b)$

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$		Challenger
$A^{Enc_k(\cdot)}$	$\xrightarrow{m'}$	$k \leftarrow Gen(1^n)$
	$\xleftarrow{c'}$	
	$\vdots$	
	$\xrightarrow{m_0, m_1}$	$b \leftarrow \{0,1\}$
	$\xleftarrow{c}$	$c \leftarrow Enc_k(m_b)$
$A^{Enc_k(\cdot)}$	$\xrightarrow{m'}$	
	$\xleftarrow{c'}$	
	$\vdots$	

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

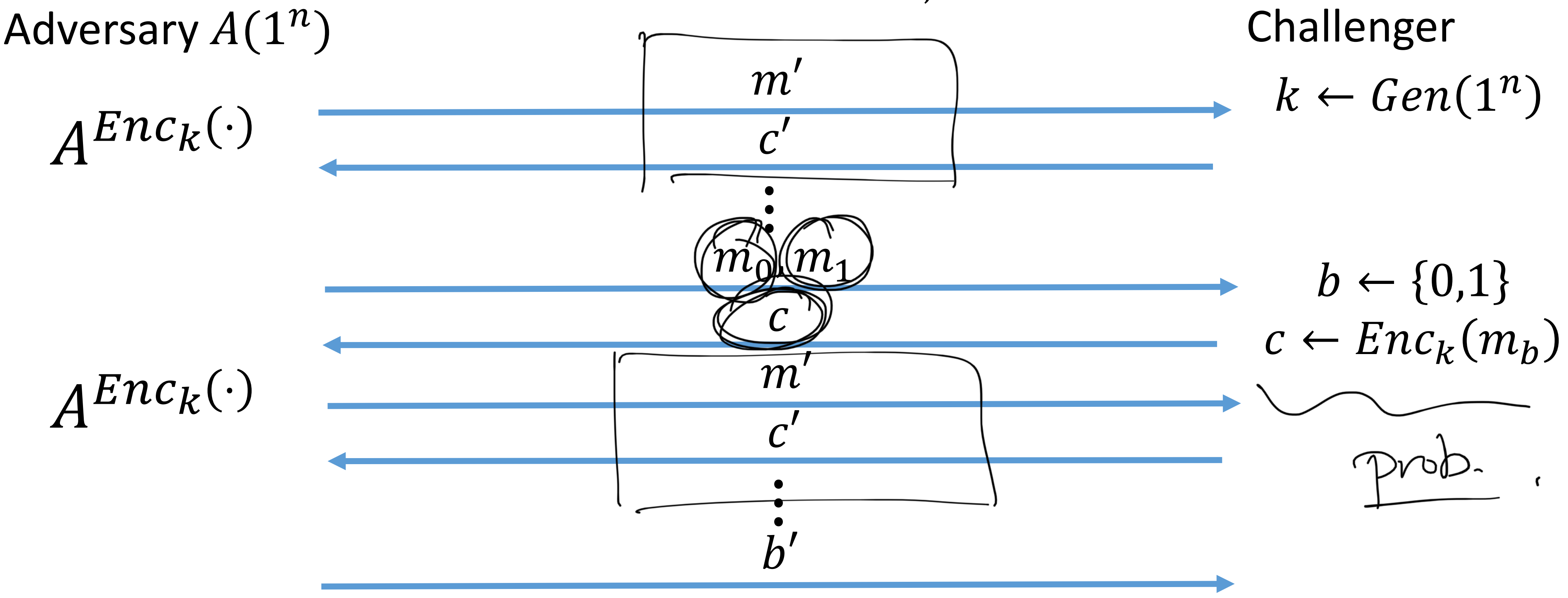
Experiment $PrivK^{cpa}_{A,\Pi}(n)$

Adversary $A(1^n)$		Challenger
$A^{Enc_k(\cdot)}$	m' $\longrightarrow$	$k \leftarrow Gen(1^n)$
	$\longleftarrow$ c'	
	$\vdots$	
	m_0, m_1 $\longrightarrow$	$b \leftarrow \{0,1\}$
	$\longleftarrow$ c	$c \leftarrow Enc_k(m_b)$
$A^{Enc_k(\cdot)}$	m' $\longrightarrow$	
	$\longleftarrow$ c'	
	$\vdots$	
	b' guess for b $\longrightarrow$	

CPA Security

Consider a private-key encryption scheme $\Pi = (Gen, Enc, Dec)$, any adversary A, and any value n for the security parameter.

Experiment $PrivK^{cpa}_{A,\Pi}(n)$



$PrivK^{cpa}_{A,\Pi}(n) = 1$ if $b' = b$ and $PrivK^{cpa}_{A,\Pi}(n) = 0$ if $b' \neq b$.

CPA-Security

The CPA Indistinguishability Experiment $PrivK^{cpa}{}_{A,\Pi}(n)$:

1. A key k is generated by running $Gen(1^n)$.
2. The adversary A is given input 1^n and oracle access to $Enc_k(\cdot)$, and outputs a pair of messages m_0, m_1 of the same length.
3. A random bit $b \leftarrow \{0,1\}$ is chosen, and then a challenge ciphertext $c \leftarrow Enc_k(m_b)$ is computed and given to A.
4. The adversary A continues to have oracle access to $Enc_k(\cdot)$, and outputs a bit b'.
5. The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise.

CPA-Security

Definition: A private-key encryption scheme $\Pi = (Gen, Enc, Dec)$ has indistinguishable encryptions under a chosen-plaintext attack if for all ppt adversaries A there exists a negligible function $negl$ such that

CPA-secure

$$\Pr\left[PrivK^{cpa}{}_{A,\Pi}(n) = 1\right] \leq \frac{1}{2} + negl(n),$$

where the probability is taken over the random coins used by A, as well as the random coins used in the experiment.

CPA-security for multiple encryptions

Theorem: Any private-key encryption scheme that has indistinguishable encryptions under a chosen-plaintext attack also has indistinguishable multiple encryptions under a chosen-plaintext attack.

CPA-secure

secure after seeing a poly. number of encryptions.

CPA-secure Encryption Must Be Probabilisitic

Theorem: If $\Pi = (Gen, Enc, Dec)$ is an encryption scheme in which Enc is a deterministic function of the key and the message, then Π cannot be CPA-secure.

Why not?

Proof Assume Π is deterministic
present an efficient A that wins the CPA game with prob. 1. (contradicts CPA-sec. of Π)

A: pick m_0, m_1. $m_0 \neq m_1$

(1) Make 1 query to CPA oracle

$$\xrightarrow{\quad m_0 \quad}$$

$$\xleftarrow{\quad c_0 \quad}$$

(2)

$$\xrightarrow{\quad (m_0, m_1) \quad}$$

$$\xleftarrow{\quad c \quad}$$

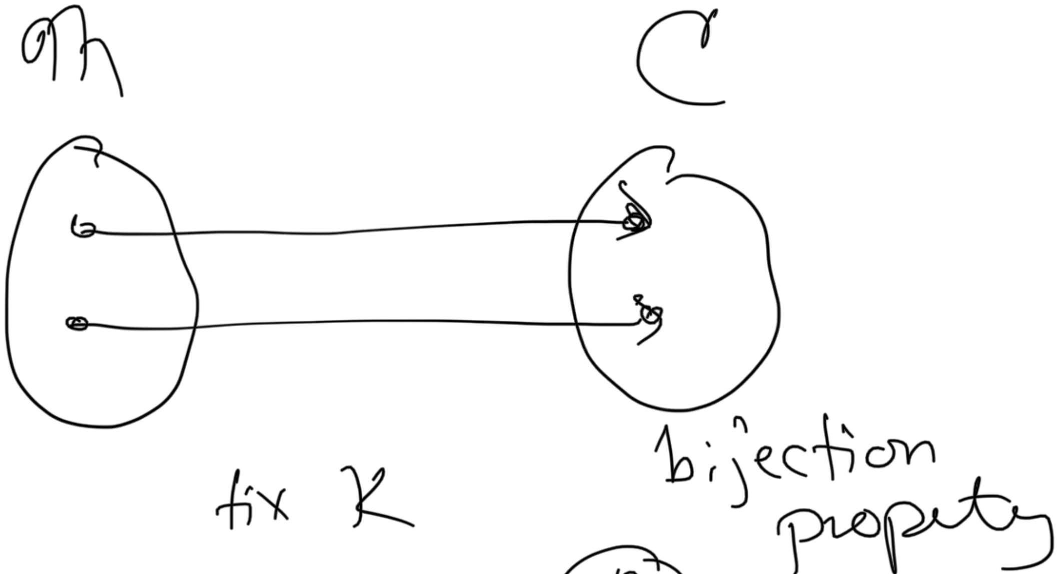
only be the same if $Enc_k(m_0)$ always outputs the same value.

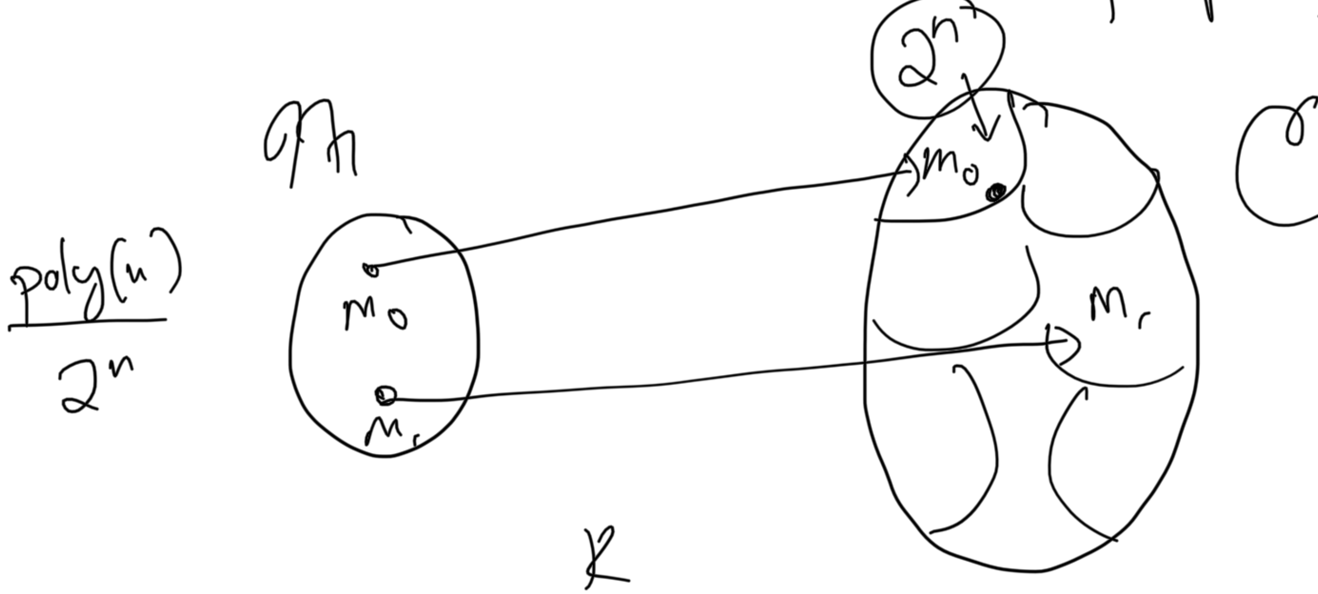
(3) If $c = c_0$, output 0 o/w output 1.

Claim: A wins w/ Pr 1.

$\square$

Constructing CPA-Secure Encryption Scheme





Still possible to perfectly decrypt when Enc is probabilistic.

PRG

Pseudorandom Function

Definition: A keyed function $F: \overbrace{\{0,1\}^*}^{K} \times \boxed{\{0,1\}^*} \rightarrow \{0,1\}^*$ is a two-input function, where the first input is called the key and denoted k.

$\boxed{F_K(\cdot)}$ to denote the restricted func. when first input is fixed to K.

$F_K(1)$ $\quad F_K(2)$ $\quad F_K(3)$

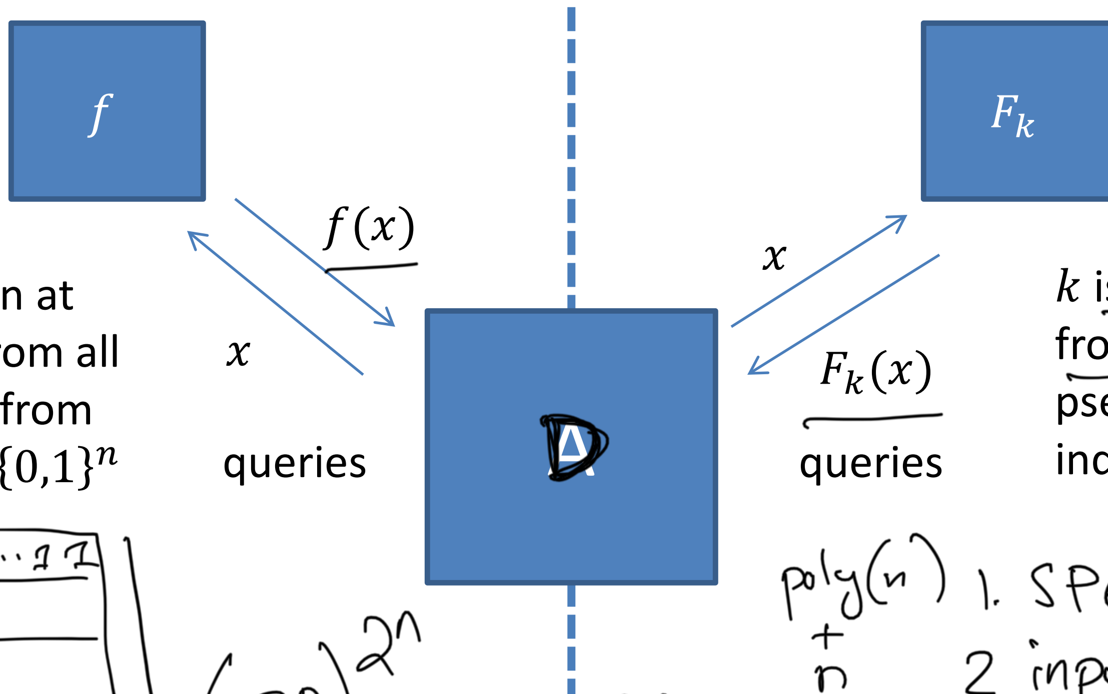
$\downarrow \quad \downarrow \quad \downarrow$

\$ \$ \$

Pseudorandom Function (PRF)

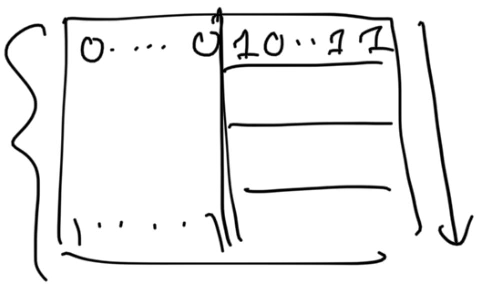
Ideal

Real



f is chosen at random from all functions from $\{0,1\}^n$ to $\{0,1\}^n$

k is chosen at random from $\{0,1\}^n$. F_k is the pseudorandom function indexed by k.



$(2^n)^{2^n} = 2^{n \cdot 2^n}$

poly(n) + n

1. SPEC in terms of 2 inputs: K, X

$F(\cdot,\cdot)$ Public

2. Choose K.

$F_K(\cdot)$

PRF: Any efficient A cannot tell which world it is in.

$$|\Pr[A^f()=1] - \Pr[A^{F_k}()=1]| \le negligible$$

$n 2^n$ bits

Pseudorandom Function

Definition: Let $F: \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ be an efficient, length-preserving, keyed function. We say that F is a pseudorandom function if for all ppt distinguishers D, there exists a negligible function $negl$ such that:

$$\left|\Pr\left[D^{F_k(\cdot)}(1^n) = 1\right] - \Pr\left[D^{f(\cdot)}(1^n) = 1\right]\right| \leq negl(n).$$

where $k \leftarrow \{0,1\}^n$ is chosen uniformly at random and f is chosen uniformly at random from the set of all functions mapping n-bit strings to n-bit strings.