CS171: Cryptography

Lecture 5

Sanjam Garg

Pseudo OTP

- Pseudo OTP is secure
  - Assuming G is a PRG
  - With respect to our definition
- Gain: Pseudo OTP has a short key
  - n bits instead of $\ell(n)$ bits
- Does pseudo OTP allow encryption of multiple messages?
  - Let’s first define it!

Security for multiple messages: several ways to define!

Mult Security

$\text{PrivK}_{A,\Pi}^{\text{mult}}(n)$

1. A for $i \in \{1 \dots t\}$ outputs $m_{0,i}, m_{1,i} \in \{0,1\}^*, |m_{0,i}| = |m_{1,i}|$.
2. b ← {0,1}, *k* ← Gen(1^n), $c_i \leftarrow Enc_k(m_{b,i})$
3. $c_1 \dots c_t$ is given to A
4. A output b'
5. Output 1 if $b = b$' and 0 otherwise

Encryption scheme $\Pi = (Gen, Enc, Dec)$ is indistinguishable multiple encryptions in the presence of an eavesdropper, or is *mult-secure* if

$\forall$ PPT A it holds that:

$$\Pr\left[\text{PrivK}_{A,\Pi}^{\text{mult}} = 1\right] \leq \frac{1}{2}$$

+ negl(n)

CPA-Security (De facto Minimum)

$\mathrm{PrivK}_{A,\Pi}^{\mathrm{CPA}}(n)$

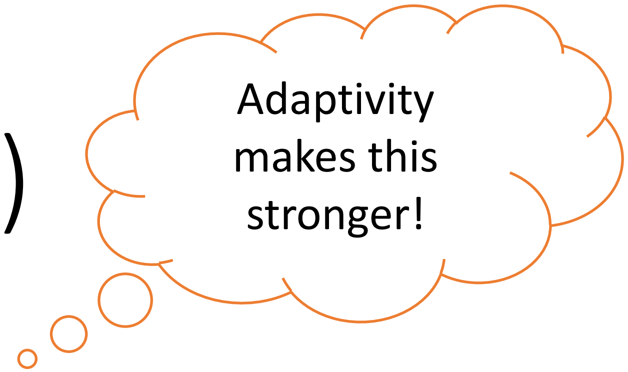
1. Sample $k \leftarrow \mathrm{Gen}(1^n)$, $A^{Enc_k(\cdot)}$ outputs $m_0, m_1 \in \{0,1\}^*, |m_0| = |m_1|$.
2. $b \leftarrow \{0,1\}$, $c \leftarrow Enc_k(m_b)$
3. c is given to $A^{Enc_k(\cdot)}$
4. $A^{Enc_k(\cdot)}$ output b'
5. Output 1 if $b = b'$ and 0 otherwise

Encryption scheme $\Pi = (Gen, Enc, Dec)$ has indistinguishable encryptions under chosen-plaintext attack, or is *CPA-secure* if

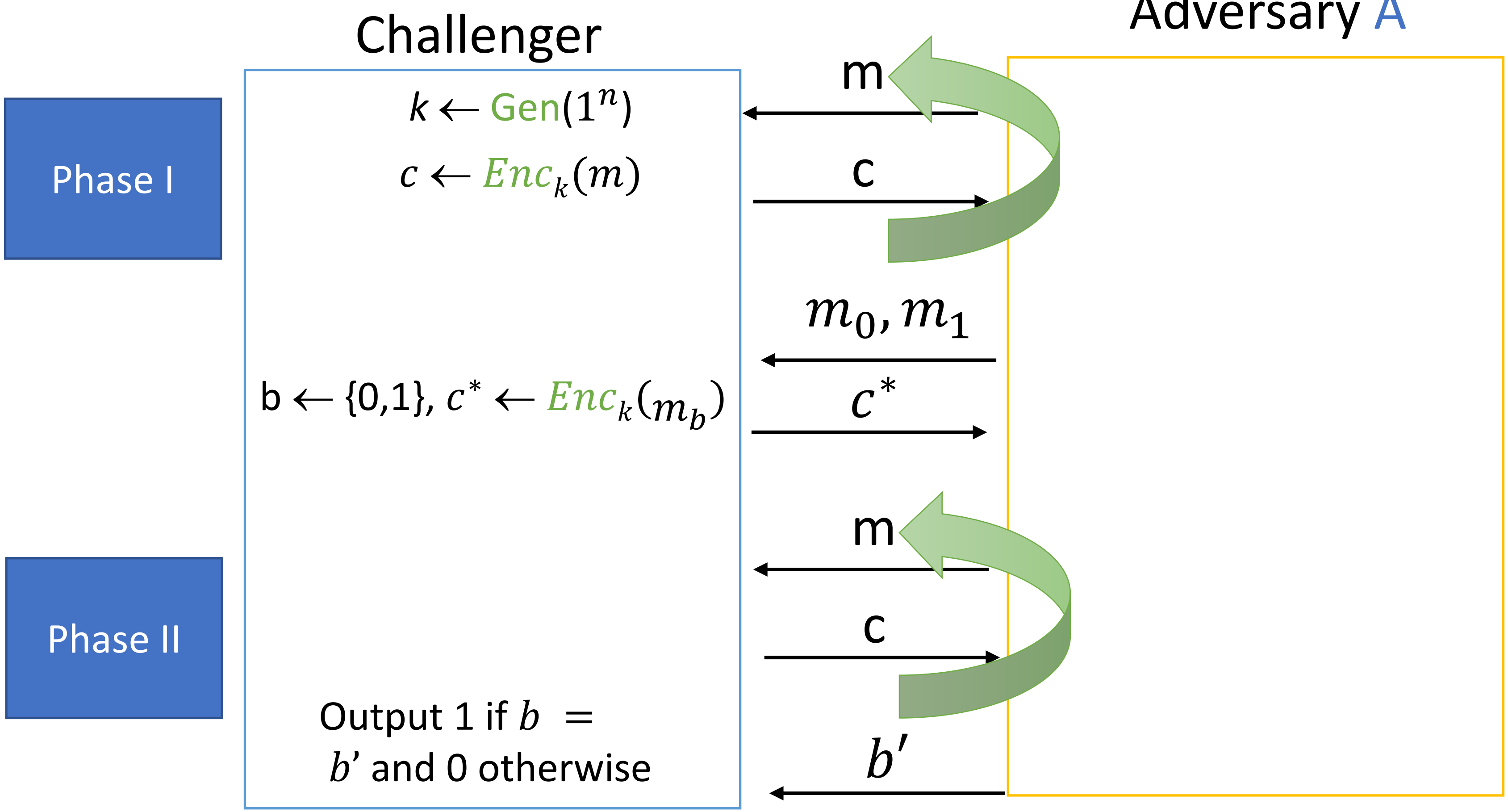
$\forall$ PPT A it holds that:

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

CPA-Security (Pictorially)



$$\mathrm{PrivK}^{\mathrm{CPA}}_{A,\Pi}(n)$$



Is Pseudo OTP CPA-secure?

$\mathrm{PrivK}^{\mathrm{CPA}}_{\mathrm{A},\Pi}(n)$

1. Sample $k \leftarrow$ Gen(1^n), $A^{Enc_k(\cdot)}$ outputs $m_0, m_1 \in \{0,1\}^*, |m_0| = |m_1|$.
2. b $\leftarrow$ {0,1}, $c \leftarrow Enc_k(m_b)$
3. c is given to $A^{Enc_k(\cdot)}$
4. $A^{Enc_k(\cdot)}$ output b'
5. Output 1 if $b = b$' and 0 otherwise

No, here is an attacker!

1. A queries $Enc_k(\cdot)$ on inputs 0^ℓ obtaining c_0.
2. A submits challenge messages 0^ℓ and 1^ℓ
3. Challenger gives c
4. A outputs 0 if $c = c_0$ and 1 otherwise.

Theorem: Any (stateless) encryption scheme with Enc a deterministic function of the key and the message cannot be CPA-secure.

CPA-Security from Multiple Encryptions

- We can define other ``seemingly'' stronger notions of CPA-security. It turns out that these notions are as equivalent as CPA.

- Easy to encrypt long messages:
  $Enc_k(m_1|| \ldots ||m_\ell) = Enc_k(m_1)|| \ldots ||Enc_{k(m_\ell)}$

- No deterministic (stateless) encryption scheme can be CPA secure.

Constructing CPA-Secure Encryption

Pseudorandom Functions (a building block)

First, what is a random function?

- Choose a uniformly random function (from the set of all functions) and then we interact with this fixed function
- Once the functions has been chosen there is no additional randomness involved.

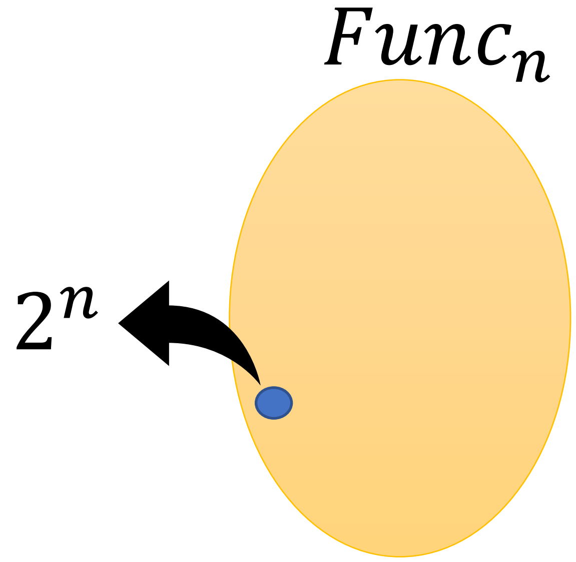
Set of all functions $Func_n$

- $Func_n$ is the set of all functions from $\{0,1\}^n \to \{0,1\}^n$.
- How many functions are there in $Func_n$:
  - How many bits does it take to describe one function?
    - $n \cdot 2^n$
  - $2^{n \cdot 2^n}$
- So, sampling a random function involves sampling one of the functions in $Func_n$ at random and fixing it
- Sometimes useful to sample the function ``on the fly''

Pseudorandom Function (PRF)

- A function that ``looks'' like a uniformly random (i.e., indistinguishable from a random) function.

- Just as for PRGs we will sample our function from a smaller space.



Keyed Functions

- F : $\{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$, where n is the security parameter.
- $F(k, x)$: The first input is the key and the second the input (also denoted by $F_k(x)$)
- Key, input and output lengths could be different, but we will use n for simplicity.
- F_k will be the sampled function which we will claim to be pseudorandom. On input x the output $F_k(x)$) = $F(k, x)$
- Only interested in efficiently computable $F(\cdot,\cdot)$

Pseudorandom Function (PRF)

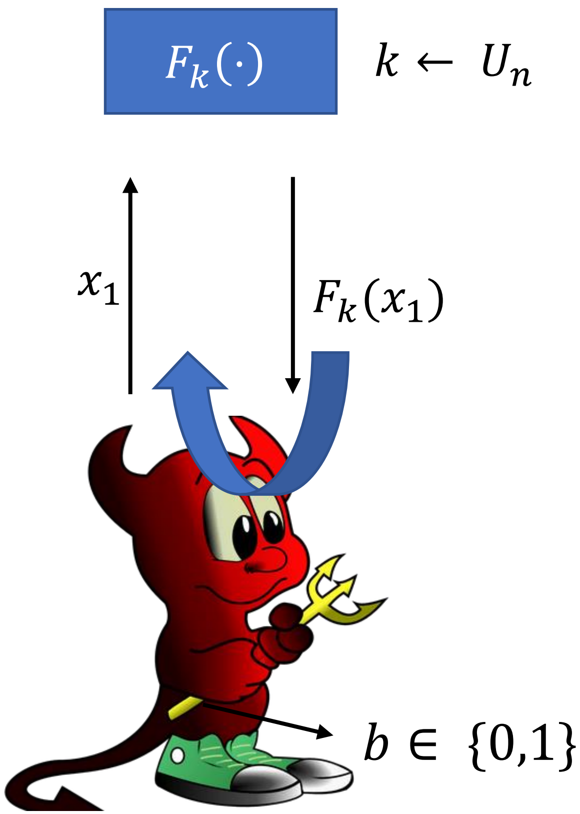
Let $F\colon \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ be an efficient, length-preserving, keyed function. F is a PRF if for all PPT distinguishers D, there is a negligible function $negl(\cdot)$ such that:

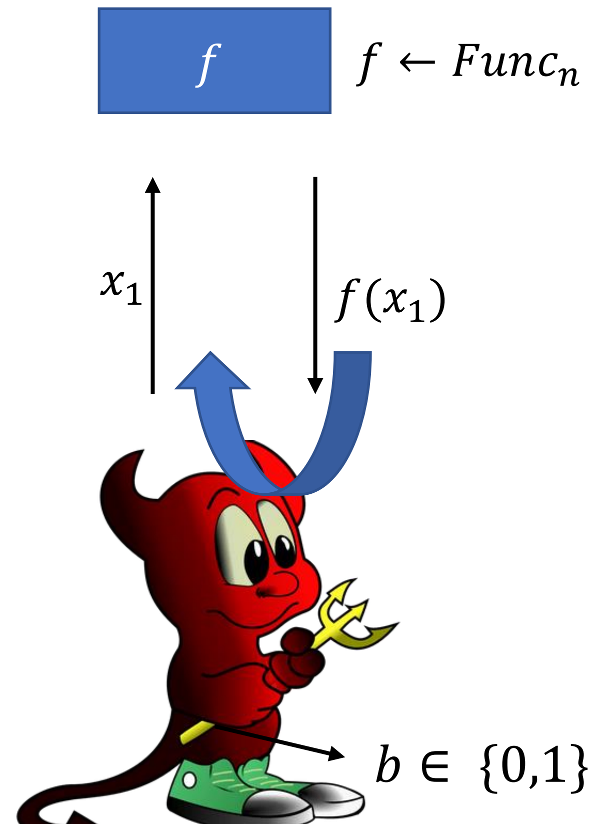
$$\left|\Pr\left[D^{F_k(\cdot)}(1^n) = 1\right] - \Pr\left[D^{f(\cdot)}(1^n) = 1\right]\right| \leq negl(n)$$

where $k \leftarrow U_n$ and $f \leftarrow Func_n$.

Definition by Picture

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





Is this a secure PRF?

- $F(k, x) = k \oplus x$?

- No, because $F(k, x_1) \oplus F(k, x_2) = k \oplus x_1 \oplus k \oplus x_2 = x_1 \oplus x_2$. This would not be the case for a random function.

Do PRFs exist?

- Seemingly stronger primitives that PRGs
- But, we know how we can construct PRFs from PRGs

CPA secure Encryption

Let F be a PRF: $\{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$.

- $Gen(1^n)$: Choose uniform $k \in \{0,1\}^n$ and output it as the key
- $Enc_k(m)$: On input a message $m \in \{0,1\}^n$, sample $r \leftarrow U_n$ output the ciphertext c as
$$c := \langle r, F_k(r) \oplus m \rangle$$
- $\mathrm{Dec}_k(c)$: On input a ciphertext $c = \langle r, s \rangle$ output the message
$$m := F_k(r) \oplus s$$

Encryption scheme is randomized!

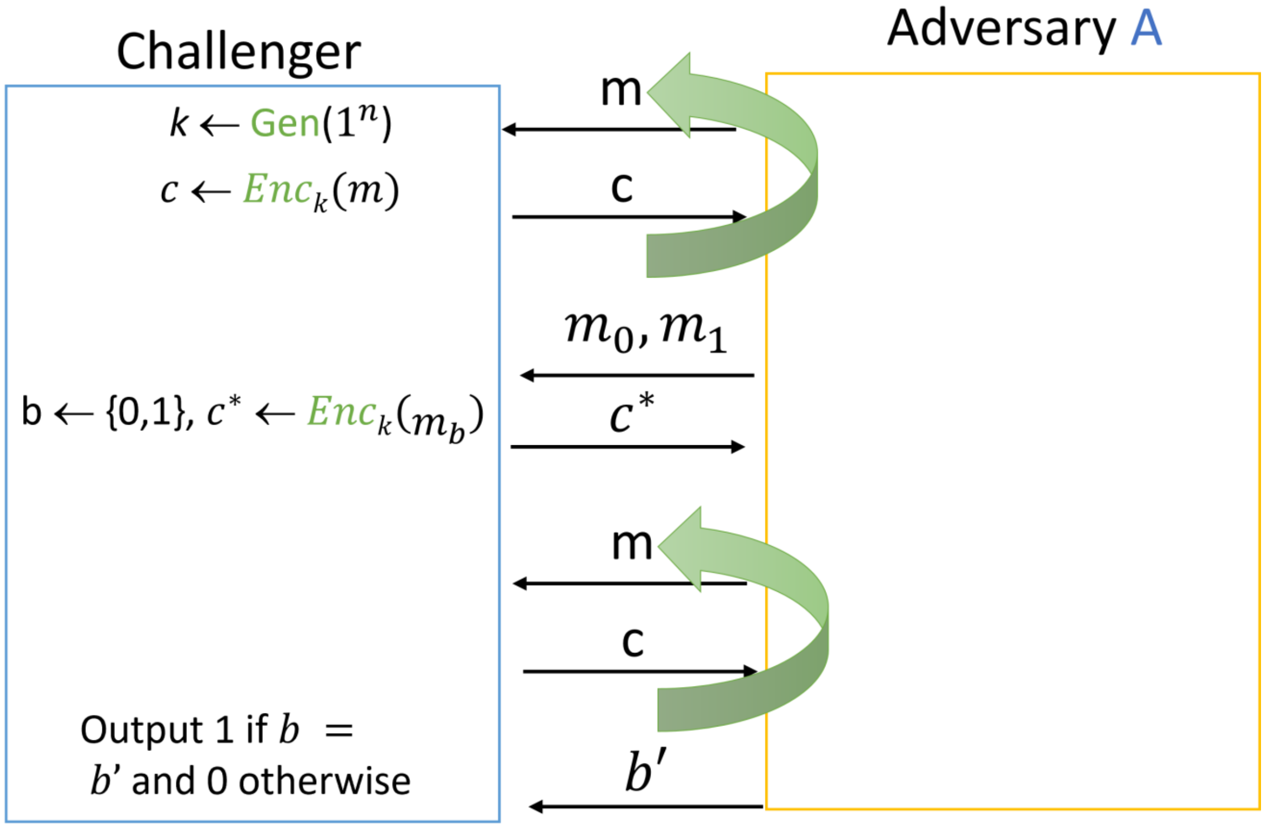
Proof of Security

- Theorem: If F is a PRF, then the construction in the previous slide is a CPA-secure encryption scheme.

- We will prove: Given an adversary A the violates a CPA-security of the encryption we will construct a distinguisher D that distinguishes between a PRF and random function.

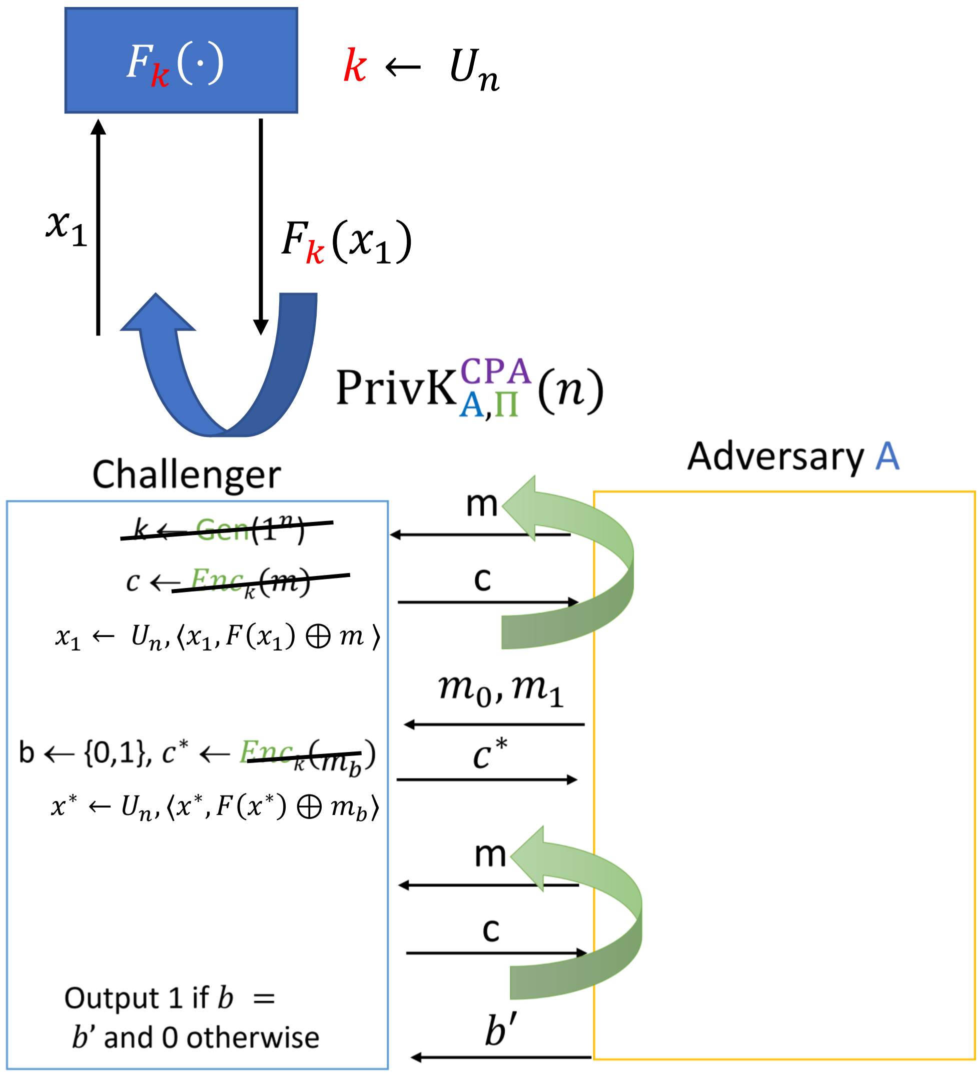
A Breaks CPA-security

$$\mathrm{PrivK}^{\mathrm{CPA}}_{A,\Pi}(n)$$



$$\Pr\left[\mathrm{PrivK}^{\mathrm{CPA}}_{A,\Pi} = 1\right] \geq \frac{1}{2} + \epsilon(n)$$

Step 1: Pull out F_k from Challenger

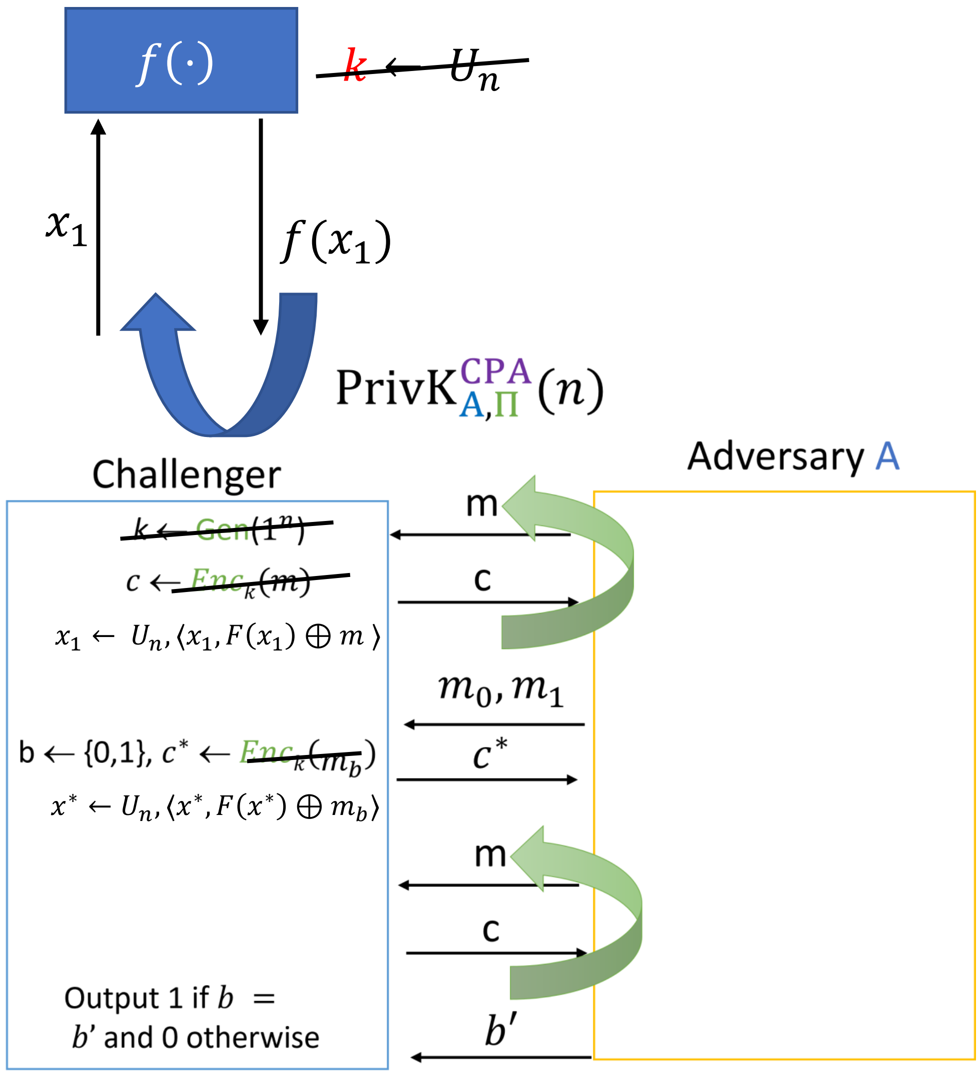


Recall $Enc_k(m)$: On input a message $m \in \{0,1\}^n$, sample $r \leftarrow U_n$ output the ciphertext c as

$$c \coloneqq \langle r, F_k(r) \oplus m \rangle$$

$$\Pr\left[\mathrm{PrivK}^{\mathrm{CPA},1}_{\mathrm{A},\Pi} = 1\right] \geq \frac{1}{2} + \epsilon(n)$$

Step 2: Switch PRF with random f



$$\delta = \left|\Pr\left[\mathrm{PrivK}^{\mathrm{CPA},2}_{\mathrm{A},\Pi} = 1\right] - \Pr\left[\mathrm{PrivK}^{\mathrm{CPA},1}_{\mathrm{A},\Pi} = 1\right]\right|$$

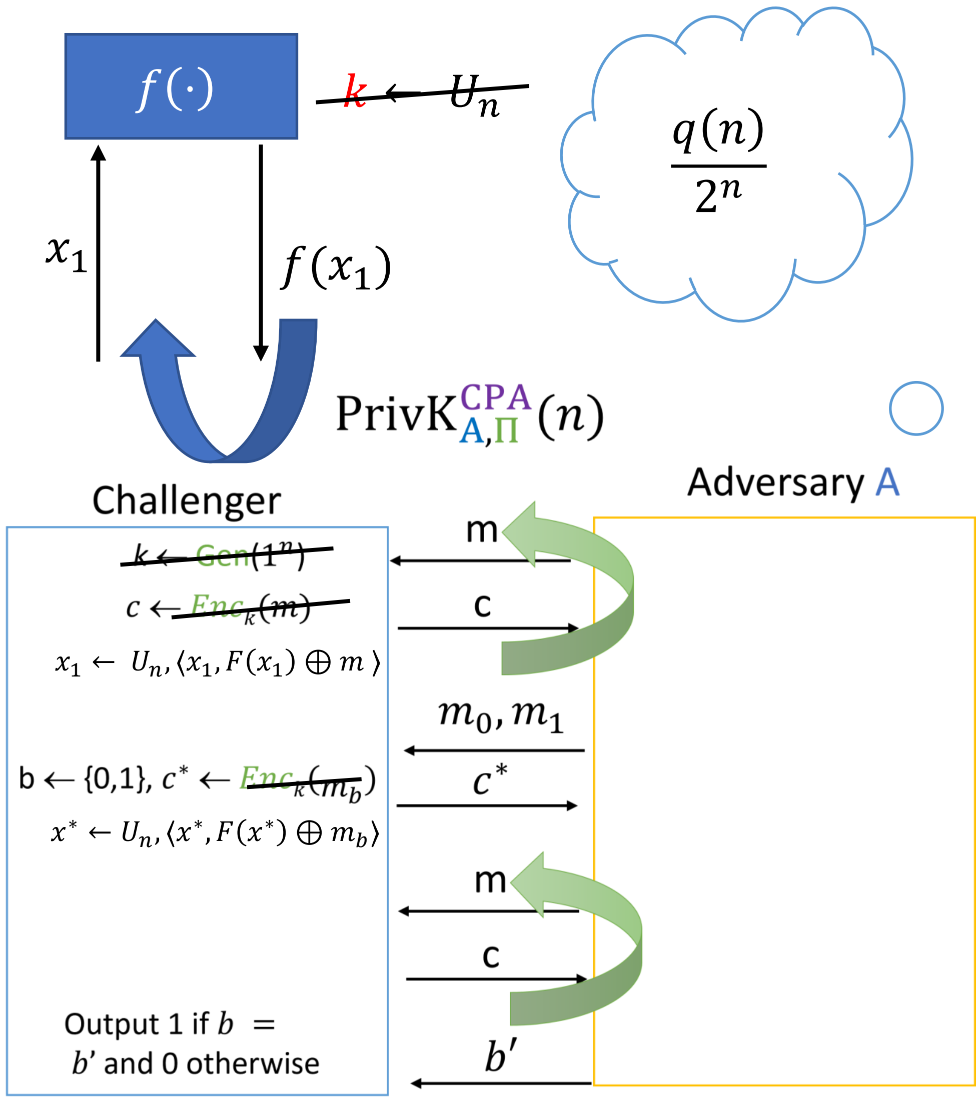
Case I: δ is non-neg(n)

Challenger/Adversary combination distinguishes PRF from random function

$$\left|\Pr\left[CA^{F_k(\cdot)}(1^n) = 1\right] - \Pr\left[CA^{f(\cdot)}(1^n) = 1\right]\right| = \delta$$

A contradiction

Step 2: Switch PRF with random f



$$\delta = \left|\Pr\left[\mathrm{PrivK}^{\mathrm{CPA},2}_{A,\Pi} = 1\right] - \Pr\left[\mathrm{PrivK}^{\mathrm{CPA},1}_{A,\Pi} = 1\right]\right|$$

Case II: δ is neg(n)

$$\Pr\left[\mathrm{PrivK}^{\mathrm{CPA},2}_{A,\Pi} = 1\right] \geq \frac{1}{2} + \epsilon'(n)$$

Claim: If $\forall i$ we have that $x^* \neq x_i$, then $F(x^*)$ is uniform (same as OTP) and $\Pr\left[\mathrm{PrivK}^{\mathrm{CPA},2}_{A,\Pi} = 1\right] = \frac{1}{2}$

A contradiction

PRF based OTP

- Get's CPA security
- Can encrypt message of arbitrary length
  $Enc_k(m_1 || \dots || m_t) = Enc_k(m_1) || \dots Enc_k(m_t)$

- Negative: $Enc_k(m) = \langle r, F_k(r) \oplus m \rangle$
  - Ciphertext size is double the message length

CPA-security is stronger that Mult-security

Why are we looking at this weird scheme ?

- How can we prove this?
  - Construct an encryption scheme Π that is Mult-secure but not CPA-secure.
  - Simplify problem: Assume Φ is Mult-secure and CPA secure then we will weaken Φ to get Π so that it is Mult-secure but not CPA-secure
- Given $\Phi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ we set $\Pi = (\mathrm{Gen}', \mathrm{Enc}', \mathrm{Dec}')$
- $Gen'(1^n)$: Set $k' = (k, m^*)$ where $k, m^* \leftarrow Gen(1^n)$
- $\mathrm{Enc}'_{k'}(m)$: If $m = m^*$ then output m^*. Otherwise, output $Enc_k(m)||\mathrm{m}^*$.
- $\mathrm{Dec}'_{k'}(c)$: Define naturally!

Need to prove that (1) Π is mult-secure but (2) is not CPA-secure!

(1) Π is mult-secure

- The probability A can ask for an encryption of m^* is negligible (or at most $\frac{2t}{2^n}$) as the secret-key has at least n-bits.
- If there are no such ``weird'' queries, then then game is same as the mult-game for Φ.

$\mathrm{PrivK}^{\mathrm{mult}}_{\mathrm{A},\Pi}(n)$

1. A for $i \in \{1 \dots t\}$ outputs $m_{0,i}, m_{1,i} \in \{0,1\}^*, |m_{0,i}| = |m_{1,i}|$.
2. $b \leftarrow \{0,1\}, k \leftarrow$ Gen(1^n), $c_i \leftarrow Enc_k(m_{b,i})$
3. $c_1 \dots c_t$ is given to A
4. A output b'
5. Output 1 if $b = b'$ and 0 otherwise

(2) Π is not CPA-secure

$\mathrm{PrivK}^{\mathrm{CPA}}_{\mathrm{A},\Pi}(n)$

1. Sample $k \leftarrow \mathrm{Gen}(1^n)$, $A^{Enc_k(\cdot)}$ outputs $m_0, m_1 \in \{0,1\}^*, |m_0| = |m_1|$.
2. b $\leftarrow$ {0,1}, $c \leftarrow Enc_k(m_b)$
3. c is given to $A^{Enc_k(\cdot)}$
4. $A^{Enc_k(\cdot)}$ output b'
5. Output 1 if $b = b$' and 0 otherwise

A

1. Query $Enc_k(\cdot)$ on input 0^n and let $c||m^*$ be the received ciphertext
2. Submit $m_0 = m^*$ and $m_1 = 0^n$.
3. Output 0 if $c^* = m^*$ and 1 otherwise.

Thank You!