CS 442
Introduction to Cryptography

Lecture 25: Review Class

Instructor: Aarushi Goel

Spring 2026

Announcements

* HW5 is due on May 3.
* Final exam on April 12 (12-3 PM).
* Syllabus: Everything from Lecture 1 to Lecture 22..
* You can bring two handwritten cheat sheets.

Message Authentication Codes (MACs)

Definition: A one-time message athentication code scheme with message space $\mathcal{M}$, key space $\mathcal{K}$ and signature/tag space $\mathcal{S}$, comprises of the following algorithms:

* $\text{Keygen} \rightarrow K$: This algorithm samples a key $K \xleftarrow{\$} \mathcal{K}$.
* $\text{Sign}(m, K) \rightarrow \sigma$: On input a message $m \in \mathcal{M}$ & key $K \in \mathcal{K}$, it outputs a signature $\sigma \in \mathcal{S}$.
* $\text{Verify}(m, \sigma, K) \rightarrow b$: On input a message $m \in \mathcal{M}$, key $K \in \mathcal{K}$ and signature $\sigma \in \mathcal{S}$, it outputs a bit $b \in \{0,1\}$ (where 1 means yes, 0 means no).

These algorithms must satisfy the following:

→ Correctness: $\forall K \in \mathcal{K}$, $m \in \mathcal{M}$, it holds that $\Pr[\text{Verify}(m, \text{Sign}(m, K), K) = 1] = 1$

→ Unforgeability: For all PPT adversaries, there exists a negligible function $\nu(.)$, such that the following holds in the game below:

$$\Pr[\text{Adv wins}] \leq \nu(|K|)$$

Many-time message Authentication game:

Adversary

Challenger

KeyGen → K

polynomial number of queries

$m_i \in M$

σ_i

Sign $(m_i, K) \rightarrow \sigma_i$

(m, σ)

Adv wins if:

① $\forall i \quad m \neq m_i$

② Verify $(m, \sigma, K) = 1$

MACs

Let $MAC_1 = (Keygen_1, Sign_1, Verify_1)$ and $MAC_2 = (Keygen_2, Sign_2, Verify_2)$ be two MAC schemes. Only one of these is many-time unforgeable, but we do not know which one. Which of the following are unforgeable MACs:

1. Keygen: $K_1 \leftarrow Keygen_1$, $K_2 \leftarrow Keygen_2$. Output $K = (K_1, K_2)$.
   $Sign((K_1, K_2), m)$: $\sigma = (Sign_1(K_1, m), Sign_2(K_2, m))$.

This is an unforgeable MAC.

* Let us assume for the sake of the contradiction that a PPT adversary can forge a MAC $\sigma = (\sigma_1, \sigma_2)$ on a message m, for which the adversary has never seen another MAC. If σ is valid, then both σ_1 & σ_2 must be valid MACs with respect to the two underlying MAC schemes.

However, since we know that at least one of the underlying schemes is unforgeable, no PPT adversary should be able to forge a valid signature on the MAC scheme.

Hence our assumption is wrong, and no PPT adversary can forge a valid signature $\sigma = (\sigma_1, \sigma_2)$ in this new MAC scheme.

2. Keygen : $K_1 \leftarrow \text{Keygen}_1$, $K_2 \leftarrow \text{Keygen}_2$. Output $K = (K_1, K_2)$.

$\text{Sign}((K_1, K_2), m_1 \| m_2)$: $\sigma = \text{Sign}_1(K_1, m_1) \oplus \text{Sign}_2(K_2, m_2)$

This is not an unforgeable MAC.

Consider the following adversary: (Let's assume MAC_1 is unforgeable).

Adversary

$m = m_1 \| m_2$ →

← σ

Challenger

$(m' = m_1 \| m_3, \sigma')$ →

Since MAC_2 is not unforgeable, forge σ_2 on m_2 & σ_2' on $m_3 \neq m_2$

Compute $\sigma' = \sigma \oplus \sigma_2 \oplus \sigma_2'$

this is a valid signature on $m' = m_1 \| m_3$

The case where MAC_2 is unforgeable, but MAC_1 is not will also be similar.

Collision-Resistant Hash Functions

Definition: A family of functions $H = \{h_K : D_K \to R_K\}_{K \in \mathcal{K}}$ is a collision-resistant hash function family (CRHF) if:

* Easy to Sample: There exists a PPT Gen, such that $K \leftarrow Gen(n)$, $K \in \mathcal{K}$.
* Compression: $|R_K| < |D_K|$.
* Easy to Evaluate: There exists a polynomial-time algorithm Eval, such that, given $x \in D_K$, $K \in \mathcal{K}$, $Eval(K, x) = h_K(x)$.
* Collision Resistance: For all non-uniform PPT adversaries A, there exists a negligible function $\mu(\cdot)$, such that

$$Pr\left[\begin{array}{l|l} x \neq x' \text{ and } & K \xleftarrow{\$} Gen(n), \\ h_K(x) = h_K(x') & (x, x') \leftarrow A(K, n) \end{array}\right] \leq \mu(n).$$

Merkle - Damgård Transform

* Let $h_K : \{0,1\}^{2n} \to \{0,1\}^n$ be a CRHF
* We can design a CRHF $H_K : \{0,1\}^* \to \{0,1\}^n$ as follows:

1. Let $x \in \{0,1\}^L$ be an L-bit input that we want to hash using H.
2. Parse $x = x_1, \ldots, x_B$, where $B = \frac{L}{n}$ and each $x_i \in \{0,1\}^n$.
   If L is not divisible by n, we pad x with appropriately many 0s.
3. Set $IV = 0^n$ initialization vector.

Claim: If h_K is a CRHF, then so is H_K.

MAC and Hash (from Homework 4)

(10 points) Let $\left\{h^k : \{0,1\}^{2n} \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$ be a family of collision-resistant hash functions and $\left\{H^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$ be a family of collision-resistant hash functions obtained via a Merkle-Damgård transformation on $\left\{h^k : \{0,1\}^{2n} \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$. Let $k \xleftarrow{\$} \mathcal{K}$, consider the following message authentication code scheme for messages $m \in \{0,1\}^*$.

- $\mathsf{KeyGen} \to K$: Sample a key $K \xleftarrow{\$} \{0,1\}^n$.
- $\mathsf{Sign}(K, m) \to \sigma$: Compute and output $\sigma = H^k(m \| K)$.
- $\mathsf{Verify}(K, m, \sigma) \to b$: Check if $H^k(m \| K)$ equals σ. Output $b = 1$ if equal, and $b = 0$ otherwise.

Determine whether this is a many-time unforgeable message authentication code (MAC) scheme). If you believe it is, provide an explanation. If you believe it it not, show a concrete attack.

This is a many-time unforgeable MAC

Let's assume we only use this scheme for messages $m \in \{0,1\}^{2n}$.

for any message $m = m_1 \| m_2$

$$\sigma = h_K(h_K(h_K(h_K(IV \| m_1) \| m_2) \| K) \| L)$$

* To win the unforgeability game, after querying signatures on messages of its choice, the adversary must forge a signature σ' on a new message, m'.

Case1: Let σ' be such that it is also a valid signature on some other previously querried message m.

Let $m = m_1 \| m_2$ $m' = m_3 \| m_4$. This would imply $h_K(h_K(IV \| m_1) \| m_2)$ $= h_K(h_K(IV \| m_3) \| m_4)$. But a PPT adversary should not be able find such m_3, m_4 because of collision resistance.

Case 2: σ' is not equal to any other previously queried σ.

Observe that this is only possible if the adversary can guess K. However, since K is sampled uniformly at random, the probability that he can guess K is $\frac{1}{2^n}$, which is extremely small for sufficiently large n.

One-Message Secure Public-Key Encryption

Definition: A one-message secure PKE scheme with message space $\mathcal{M}$, key space $\mathcal{K}$, ciphertext space $\mathcal{C}$, comprises of the following algorithms:

* $\text{Gen} \rightarrow pk, sk$: This algorithm samples a pair $(pk, sk) \in \mathcal{K}$.
* $\text{Enc}(pk, m) \rightarrow c$: On input a public key pk & a message $m \in \mathcal{M}$, it outputs ciphertext $c \in \mathcal{C}$.
* $\text{Dec}(sk, c) \rightarrow m$: On input a secret key sk & a ciphertext $c \in \mathcal{C}$, it outputs message $m \in \mathcal{M}$.

These algorithms must satisfy the following:

→ **Correctness:** For $(pk, sk) \leftarrow \text{Gen}$, $\forall m \in \mathcal{M}$, the following holds:

$$Pr[\text{Dec}(sk, \text{Enc}(pk, m)) = m] = 1$$

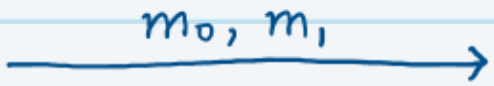
→ **Security:** For all PPT adversaries A, $\exists$ a negligible function $v(.)$, such that:

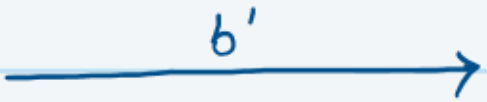
$$Pr\left[\begin{array}{l|l} (pk, sk) \leftarrow \text{Gen}(1^n), & \\ (m_0, m_1) \leftarrow A(pk), & A(pk, \text{Enc}(sk, m_b)) = b \\ b \xleftarrow{\$} \{0,1\} & \end{array}\right] \leq \frac{1}{2} + v(n).$$

Another Interpretation of One-message Security

Adversary

Challenger





$$Pr[b = b'] \leq \frac{1}{2} + \nu(n).$$

OR equivalently: $\{pk, Enc(pk, m_0)\} \approx_c \{pk, Enc(pk, m_1)\}$

$\forall m_0, m_1 \in \mathcal{M}.$

Public-Key Encryption

Order-preserving encryption: The attempt is to look at the possibility of a PKE where the ciphertexts follow the same lexicographic order as the messages. This property would be useful for computing on encrypted databases. Let (Gen, Enc, Dec) be a public-key encryption for message space $\mathcal{M}$ such that: $\forall m_1, m_2 \in \mathcal{M}$, if $m_1 \leq m_2$, then $Enc(pk, m_1) \leq Enc(pk, m_2)$, where $(pk, sk) \leftarrow Gen$. Can such an encryption be IND-CPA secure?

→ No!

consider the following adversary:

Adversary

Challenger

Let $m_0 < m_1$

Compute $c_0 = Enc(pk, m_0)$

$c_1 = Enc(pk, m_1)$

pk ←

m_0, m_1 →

c ←

$(pk, sk) \leftarrow Gen$

$b \xleftarrow{\$} \{0,1\}$

$c = Enc(pk, m_b)$

if $c_0 \leq c$ and $c < c_1$, output $b = 0$

else output $b = 1$.

Ciphertext Only Attack (COA)

Definition: A COA-secure encryption scheme with message space $\mathcal{M}$, key space $\mathcal{K}$, ciphertext space $\mathcal{C}$, comprises of the following algorithms:

* $\text{KeyGen} \longrightarrow K$: This algorithm samples a key $K \in \mathcal{K}$.
* $\text{Enc}(K, m) \rightarrow c$: On input a key $K \in \mathcal{K}$ and a message $m \in \mathcal{M}$, it outputs ciphertext $c \in \mathcal{C}$.
* $\text{Dec}(K, c) \rightarrow m$: On input a key $K \in \mathcal{K}$ and a ciphertext $c \in \mathcal{C}$, it outputs message $m \in \mathcal{M}$.

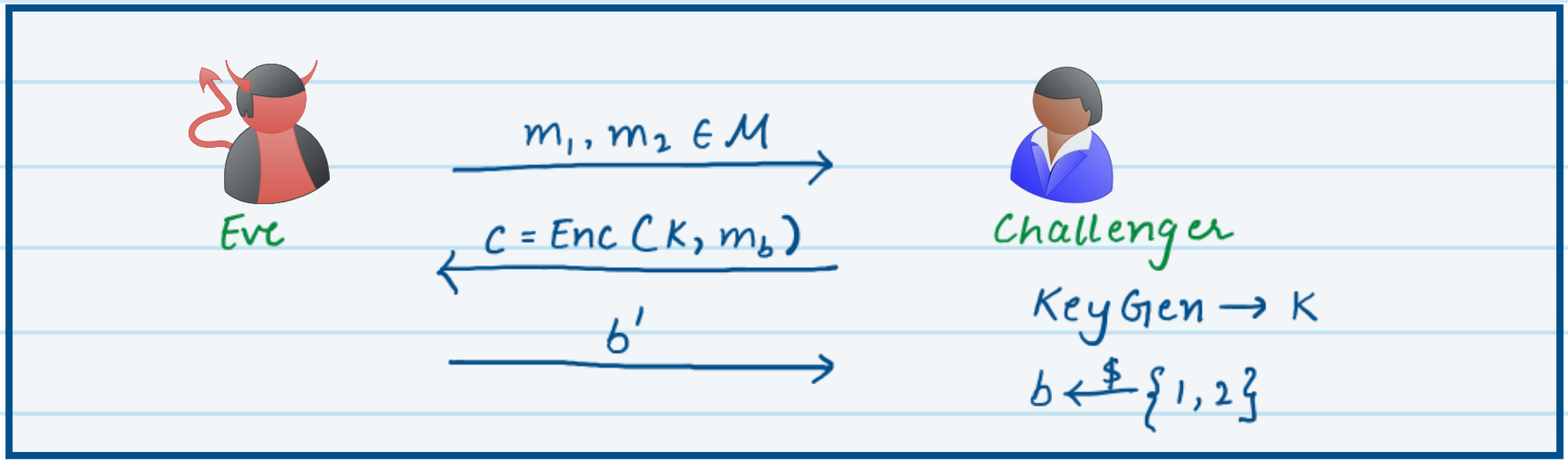
These algorithms must satisfy the following:

→ Correctness: $\forall K \in \mathcal{K}$, $\forall m \in \mathcal{M}$, it holds that:

$$Pr[\text{Dec}(K, \text{Enc}(K, m)) = m] = 1$$

→ COA-Security: For every PPT Eve, $\exists$ a negligible function $\nu(.)$, such that the following holds in the game below:

$$Pr[b = b'] \leq \frac{1}{2} + \nu(|K|)$$


Eve
$m_1, m_2 \in \mathcal{M}$
$c = Enc(K, m_b)$
b'
Challenger
KeyGen → K
$b \xleftarrow{\$} \{1, 2\}$

CPA Secure Encryption (Also called multi-message secure encryption)

Definition: A CPA-secure encryption scheme with message space $\mathcal{M}$, key space $\mathcal{K}$, ciphertext space $\mathcal{C}$, comprises of the following algorithms:

* $\text{KeyGen} \longrightarrow K$: This algorithm samples a key $K \in \mathcal{K}$.
* $\text{Enc}(K, m) \rightarrow c$: On input a key $K \in \mathcal{K}$ and a message $m \in \mathcal{M}$, it outputs ciphertext $c \in \mathcal{C}$.
* $\text{Dec}(K, c) \rightarrow m$: On input a key $K \in \mathcal{K}$ and a ciphertext $c \in \mathcal{C}$, it outputs message $m \in \mathcal{M}$.

These algorithms must satisfy the following:

→ Correctness: $\forall K \in \mathcal{K}$, $\forall m \in \mathcal{M}$, it holds that:

$$Pr[\text{Dec}(K, \text{Enc}(K, m)) = m] = 1$$

→ CPA-Security: For all PPT adversaries, there exists a negligible function $\nu(\cdot)$ such that the following holds in the game below:

$$Pr[b = b'] \leq \frac{1}{2} + \nu(|K|)$$

Adversary

Challenger

KeyGen → K

query polynomial no. of times { m → ← c } Enc(K, .)

$m_0, m_1 \in \mathcal{M}$ →

← $c_b = Enc(K, m_b)$

$b \xleftarrow{\$} \{0,1\}$

can include previously queried messages

query polynomial no. of times { m → ← c } Enc(K, .)

b' →

Modes of Operation (from Homework 3)

Let $\{f_k\}_k$ be a family of *pseudorandom permutations (PRPs)*, where $f_k : \{0,1\}^n \to \{0,1\}^n$ and $k \in \{0,1\}^n$. Consider the following mode of operation (a variant of counter (CTR) mode from Lecture 14) for encrypting arbitrarily long messages $m \in \{0,1\}^{nt}$, where t is arbitrary:

- $\mathsf{KeyGen} \to k$: Sample a key $k \xleftarrow{\$} \{0,1\}^n$.
- $\mathsf{Enc}(k, m) \to c$: Parse $m = m_1 \| \ldots \| m_t$, where each $m_i \in \{0,1\}^n$. Sample $\mathsf{ctr} \xleftarrow{\$} \{0,1\}^n$. For each $i \in [t]$, compute $c_i = f_k(\mathsf{ctr} + i + m_i)$. Output $c = (\mathsf{ctr}, c_1, \ldots, c_t)$.

1. **(5 points)** Describe the corresponding decryption algorithm.
2. **(10 points) Is this scheme CPA-secure?** If you believe it is CPA-secure, provide an intuitive explanation (a formal proof is not required). If you believe it is not secure, describe a concrete attack that violates CPA-security.
3. **(10 points)** Recall from Lectures 13 that a *ciphertext only attack (COA)-secure* encryption scheme is one where each key is used to encrypt only a single message. **Is this scheme COA secure?** If you believe it is secure, provide a formal proof. Otherwise, describe a concrete attack that violated COA security.

1 $Dec(K, c)$: Parse $c = (ctr, c_1, \ldots, c_t)$

$\forall\, i \in [n]$, compute $m_i = f_K^{-1}(c_i) - ctr - i$

Output $m = m_1 \| \ldots \| m_n$.

2. This scheme is not CPA-secure. Consider the following adversary.

Adversary

Challenger

$K \leftarrow Gen$

Let $m_0 = 0\ldots01 \| 0\ldots10 \| \ldots$

$m_1 = 0^n \| 1^n \| \ldots$

m_0, m_1 →

← c

$b \xleftarrow{\$} \{0,1\}$

$c = Enc(pk, m_b)$

parse $c = (ctr, c_1 \ldots, c_n)$

if $c_1 = c_2$ output $b = 0$

else output $b = 1$.

Observe that if c is an encryption of $m_0 = 0\ldots01 \| 0\ldots010$

$$\text{Enc}(pk, c): \quad c_1 = f_k(0^n \oplus 0\ldots01 \oplus 0\ldots01) = f_k(0^n \oplus 0\ldots010 \oplus 0\ldots010)$$
$$= c_2$$

3. This scheme is not COA-secure.

Why? The same attack as above still works.

Key-Exchange Protocol

randomized

Definition: A two-party key-exchange protocol π is a probabilistic interactive algorithm between two parties Alice and Bob, where transcript represents all the messages exchanged between Alice and Bob during the protocol. At the end, Alice locally computes $K_A \in \{0,1\}^n$, and Bob locally computes $K_B \in \{0,1\}^n$.

* **Correctness:** $\forall n \in \mathbb{N}$, $Pr[K_A = K_B] = 1$
* **Security:** For all PPT adversaries Eve, there exists a negligible function $\nu(.)$, such that, for $r \xleftarrow{\$} \{0,1\}^n$,

$$Pr[Eve(transcript, K_A) = 1] - Pr[Eve(transcript, r) = 1] \leq \nu(n).$$

In other words.

$$\{(transcript, K_A)\} \approx_c \{(transcript, r)\}$$

computationally indistinguishable.

Key Exchange

Consider a cyclic group $(G, \cdot, g, q)$ where the DDH assumption does not hold

$\Rightarrow$ Sample $x, y, z \xleftarrow{\$} \mathbb{Z}_q$, compute $h_1 = g^x$, $h_2 = g^y$. Given (G, q, g, h_1, h_2) it is easy to distinguish between g^{xy} and g^z.

However, the following assumption called the computational Diffie-Hellman (CDH) assumption still holds.

> * CDH Assumption: Sample $x, y, z \xleftarrow{\$} \mathbb{Z}_q$, compute $h_1 = g^x$, $h_2 = g^y$.
>
> Given (G, q, g, h_1, h_2), it is hard to compute g^{xy}.

Does the Diffie-Hellman key exchange protocol still remain secure in this group?

Diffie-Hellman Key-Exchange Protocol

Alice

Bob

$(\mathbb{G}, g, q)$

$x \xleftarrow{\$} \mathbb{Z}_q$

$X = g^x$

X

Y

$y \xleftarrow{\$} \mathbb{Z}_q$

$Y = g^y$

$K = Y^x$

$K = X^y$