Cryptography
CS 555

Topic 11: Authenticated Encryption + CCA-Security

Recap

- Message Authentication Codes
- Secrecy vs Confidentiality

Today's Goals:

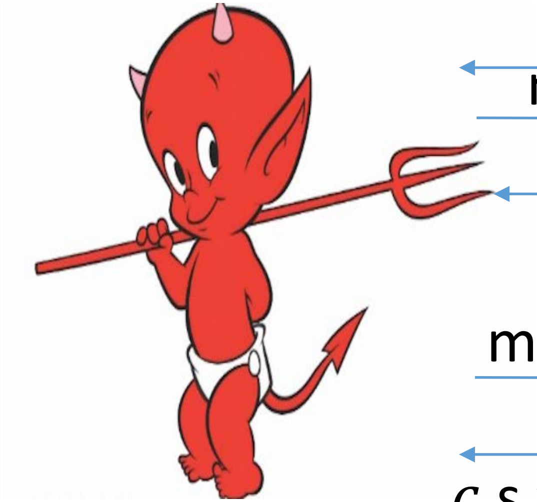
- Authenticated Encryption
- Build Authenticated Encryption Scheme with CCA-Security

Authenticated Encryption

Encryption: Hides a message from the attacker

Message Authentication Codes: Prevents attacker from tampering with message

Unforgeable Encryption Experiment ($\mathrm{Encforge}_{A,\Pi}(n)$)



m_1 →

← $c_1 = \mathrm{Enc}_K(m_1)$

m_2 →

← $c_2 = \mathrm{Enc}_K(m_2)$

⋮

m_q →

← $c_q = \mathrm{Enc}_K(m_q)$

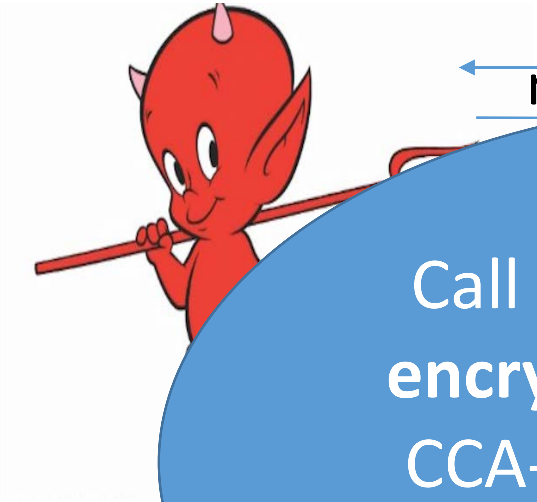
c s.t $c \notin \{c_1, \ldots, c_q\}$ →

← $\mathrm{Encforge}_{A,\Pi}(n) = 1$ if $\mathrm{Deck}(c) \neq \perp$

K = Gen(.)

$$\forall PPT\ A\ \exists \mu\ (\text{negligible})\ \text{s.t}$$
$$\Pr\left[\mathrm{Encforge}_{A,\Pi}(n) = 1\right] \leq \mu(n)$$

Unforgeable Encryption Experiment ($\text{Encforge}_{A,\Pi}(n)$)



m_1

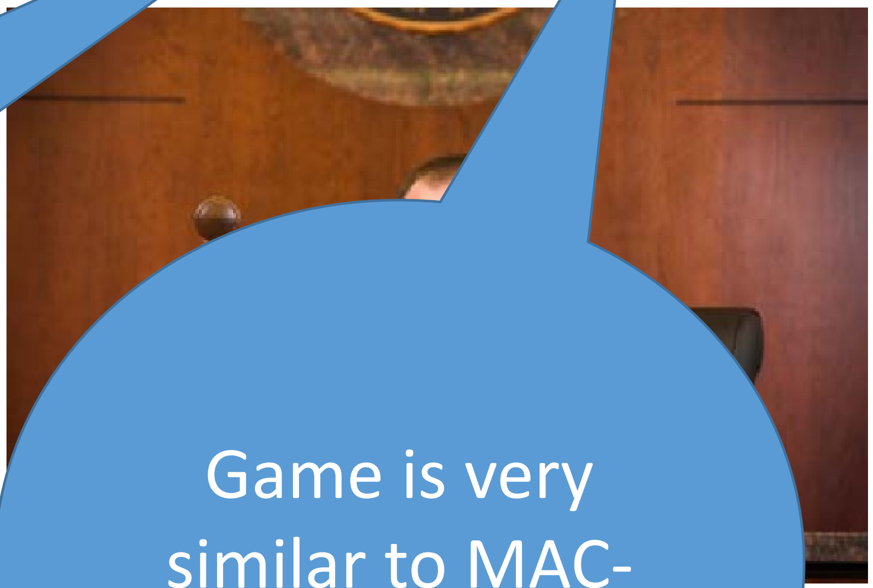
c_1 = Enc$_K$(m

m_2

$_2$)

(m$_q$)

$\neq \perp$



Call Π an **authenticated encryption scheme** if it is CCA-secure and any PPT attacker wins Encforge with negligible probability

Game is very similar to MAC-Forge game

(negligible) s.t

$$\Pr\left[\text{Encforge}_{A,\Pi}(n) = 1\right] \le \mu(n)$$

Building Authenticated Encryption

Attempt 1: Let $\mathrm{Enc}'_K(m)$ be a CPA-Secure encryption scheme and let $\mathrm{Mac}'_K(m)$ be a secure MAC

$$Enc_K(m) = \langle \mathrm{Enc}'_K(m), \mathrm{Mac}'_K(m) \rangle$$

Any problems?

$$\mathrm{Enc}'_K(m) = \langle r, F_k(r) \oplus m \rangle$$
$$\mathrm{Mac}'_K(m) = F_k(m)$$

Building Authenticated Encryption

Attempt 1:

$$Enc_K(m) = \langle r, F_k(r) \oplus m, F_k(m) \rangle$$

CPA-Attack:

- Intercept ciphertext c

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

- Ask to encrypt r

$$c_r = Enc_K(r) = \langle r', F_k(r') \oplus r, F_k(r) \rangle$$

$$m = F_k(r) \oplus (F_k(r) \oplus m)$$

Building Authenticated Encryption

Attempt 1: Let $\mathrm{Enc}'_K(m)$ be a CPA-Secure encryption scheme and let $\mathrm{Mac}'_K(m)$ be a secure MAC

$$Enc_K(m) = \langle \mathrm{Enc}'_K(m), \mathrm{Mac}'_K(m) \rangle$$

Attack exploited fact that same secret key used for MAC’/Enc’

Independent Key Principle

“different instances of cryptographic primitives should always use independent keys”

Building Authenticated Encryption

Attempt 2: (Encrypt-and-Authenticate) Let $\mathrm{Enc}'_{K_E}(m)$ be a CPA-Secure encryption scheme and let $\mathrm{Mac}'_{K_M}(m)$ be a secure MAC. Let $K = (K_E, K_M)$ then

$$Enc_K(m) = \left\langle \mathrm{Enc}'_{K_E}(m), \mathrm{Mac}'_{K_M}(m) \right\rangle$$

Any problems?

$$\mathrm{Enc}'_{K_E}(m) = \left\langle r, F_{K_E}(r) \oplus m \right\rangle$$
$$\mathrm{Mac}'_{K_M}(m) = F_{K_M}(m)$$

Building Authenticated Encryption

Attempt 2:

$$Enc_K(m) = \langle r, F_{K_E}(r) \oplus m, F_{K_M}(m) \rangle$$

CPA-Attack:

- Select m_0,m_1
- Obtain ciphertext c

$$c = \langle r, F_{K_E}(r) \oplus mb, F_{K_M}(m_b) \rangle$$

- Ask to encrypt m_0

$$c_r = \langle r', F_{K_E}(r') \oplus m_0, F_{K_M}(m_0) \rangle$$

$$F_{K_M}(m_0) =? F_{K_M}(m_b)$$

Building Authenticated Encryption

Attempt 2:

$$Enc_K(m) = \langle r, F_{K_E}(r) \oplus m, F_{K_M}(m) \rangle$$

CPA-Attack:

- Select m_0,m_1
- Obtain ciphertext c

$$c = \langle r, F_{K_E}(r) \oplus mb, F_{K_M}$$

- Ask to encrypt m_0

$$c_r = \langle r', F_{K_E}(r') \oplus m_0, F_{K_M}(m_0$$

$$F_{K_M}(m_0) =?\, F_{K_M}(m_b)$$

Encrypt and Authenticate Paradigm does not work in general

Building Authenticated Encryption

Attempt 3: (Authenticate-then-encrypt) Let $\mathrm{Enc}'_{K_E}(m)$ be a CPA-Secure encryption scheme and let $\mathrm{Mac}'_{K_M}(m)$ be a secure MAC. Let $K = (K_E, K_M)$ then

$$Enc_K(m) = \left\langle \mathrm{Enc}'_{K_E}(m \parallel t), \right\rangle \text{ where } \mathrm{t} = \mathrm{Mac}'_{K_M}(m)$$

Doesn't necessarily work: See textbook

Building Authenticated Encryption

Attempt 4: (Encrypt-then-authenticate) Let $\mathrm{Enc}'_{K_E}(m)$ be a CPA-Secure encryption scheme and let $\mathrm{Mac}'_{K_M}(m)$ be a secure MAC. Let $K = (K_E, K_M)$ then

$$Enc_K(m) = \langle \mathrm{c}, \mathrm{Mac}'_{K_M}(\mathrm{c}) \rangle \text{ where } \mathrm{c} = \mathrm{Enc}'_{K_E}(m)$$

Secure?

Building Authenticated Encryption

Theorem: (Encrypt-then-authenticate) Let $\mathrm{Enc}'_{K_E}(m)$ be a CPA-Secure encryption scheme and let $\mathrm{Mac}'_{K_M}(m)$ be a secure MAC. Then the following construction is an authenticated encryption scheme.

$$Enc_K(m) = \left\langle \mathrm{c}, \mathrm{Mac}'_{K_M}(\mathrm{c}) \right\rangle \text{ where } \mathrm{c} = \mathrm{Enc}'_{K_E}(m)$$

Proof?

Two Tasks:

$\mathrm{Encforge}_{A,\Pi}$

CCA-Security

Building Authenticated Encryption

Theorem: (Encrypt-then-authenticate) Let $\mathrm{Enc}'_{K_E}(m)$ be a CPA-Secure encryption scheme and let $\mathrm{Mac}'_{K_M}(m)$ be a secure MAC. Then the following construction is an authenticated encryption scheme.

$$Enc_K(m) = \left\langle \mathrm{c}, \mathrm{Mac}'_{K_M}(\mathrm{c}) \right\rangle \text{ where } \mathrm{c} = \mathrm{Enc}'_{K_E}(m)$$

Proof Intuition: Suppose that we have already shown that any PPT attacker wins $\mathrm{Encforge}_{A,\Pi}$ with negligible probability.

Why does CCA-Security now follow from CPA-Security?

CCA-Attacker has decryption oracle, but cannot exploit it! Why?

Always sees ⊥ "invalid ciphertext" when he query with unseen ciphertext

Proof Sketch

1. Let ValidDecQuery be event that attacker submits new/valid ciphertext to decryption oracle

2. Show Pr[ValidDecQuery] is negl(n) for any PPT attacker
   - Hint: Follows from strong security of MAC since
   $$Enc_K(m) = \langle \mathrm{c}, \mathrm{Mac}'_{K_M}(\mathrm{c}) \rangle$$
   - This also implies unforgeability.

3. Show that attacker who does not issue valid decryption query wins CCA-security game with probability ½ + negl(n)
   - Hint: otherwise we can use A to break CPA-security
   - Hint 2: simulate decryption oracle by always returning ⊥ when given new ciphertext

Secure Communication Session

- Solution? Alice transmits $c_1 = Enc_K(m_1)$ to Bob, who decrypts and sends Alice $c_2 = Enc_K(m_2)$ etc…
- Authenticated Encryption scheme is
  - Stateless
  - For fixed length-messages
- We still need to worry about
  - Re-ordering attacks
    - Alice sends 2n-bit message to Bob as $c_1 = Enc_K(m_1)$, $c_2 = Enc_K(m_2)$
  - Replay Attacks
    - Attacker who intercepts message $c_1 = Enc_K(m_1)$ can replay this message later in the conversation
  - Reflection Attack
    - Attacker intercepts message $c_1 = Enc_K(m_1)$ sent from Alice to Bob and replays to c_1 Alice only

Secure Communication Session

- Defense
  - Counters ($CTR_{A,B}$,$CTR_{B,A}$)
    - Number of messages sent from Alice to Bob ($CTR_{A,B}$) --- initially 0
    - Number of messages sent from Bob to Alice ($CTR_{B,A}$) --- initially 0
    - Protects against Re-ordering and Replay attacks
  - Directionality Bit
    - $b_{A,B} = 0$ and $b_{B,A} = 1$ (e.g., since A < B)

- Alice: To send m to Bob, set $c=Enc_K(b_{A,B} \parallel CTR_{A,B} \parallel m)$, send c and increment $CTR_{A,B}$
- Bob: Decrypts c, (if $\perp$ then reject), obtain $b \parallel CTR \parallel m$
  - If $CTR \neq CTR_{A,B}$ or $b \neq b_{A,B}$ then reject
  - Otherwise, output m and increment $CTR_{A,B}$

Authenticated Security vs CCA-Security

- Authenticated Encryption → CCA-Security (by definition)

- CCA-Security does not necessarily imply Authenticate Encryption
  - But most natural CCA-Secure constructions are also Authenticated Encryption Schemes
  - Some constructions are CCA-Secure, but do not provide Authenticated Encryptions, but they are less efficient.
- Conceptual Distinction
  - CCA-Security the goal is secrecy (hide message from active adversary)
  - Authenticated Encryption: the goal is integrity + secrecy

Next Class

- Read Katz and Lindell 5.1-5.2
- Cryptographic Hash Functions
- Homework 2 Assigned