CS171: Cryptography

Lecture 16 – Review Lecture

Sanjam Garg

MACs - Formally

- $(Gen, Mac, Vrfy)$
- $Gen(1^n)$: Outputs a key k.
- $Mac_k(m)$: Outputs a tag t.
- $Vrfy_k(m, t)$: Outputs 0/1.
- Correctness: $\forall\ n, k \leftarrow\ Gen(1^n), \forall m \in \{0,1\}^*$, we have that $Vrfy_k(m, Mac_k(m)) = 1$.
- Default Construction of $Vrfy$ (for deterministic Mac): $Vrfy_k(m, t)$ outputs 1 if and only $Mac_k(m) = t$.

Unforgeability/Security of MAC

$\mathrm{MacForge}_{A,\Pi}(1^n)$

1. Sample $k \leftarrow \mathrm{Gen}(1^n)$.
2. Let (m^*, t^*) be the output of $A^{Mac_k(\cdot)}$. Let M be the list of queries A makes.
3. Output 1 if $Vrfy_k(m^*, t^*) = 1 \wedge m^* \notin M$ and 0 otherwise.

$\Pi = (Gen, Mac, Vrfy)$ is existentially unforgeable under adaptive chosen attack, or is *eu-cma-secure* if

$\forall$ PPT A it holds that:

$\Pr[\mathrm{MacForge}_{A,\Pi} = 1] \leq$ negl(n)

Practice Problem 1 – MAC Combiner

- Combine two cryptosystems
- Give MAC Schemes $\Pi_1 = (Gen_1, Mac_1, Vrfy_1)$and $\Pi_2 = (Gen_2, Mac_2, Vrfy_2)$, construct a MAC Scheme $\Pi = (Gen, Mac, Vrfy)$ that is secure as long as at least one of Π_1 and Π_2 is secure.

Construction

- $Gen(1^n)$: Outputs key $k = (k_1, k_2)$ where $\mathrm{k}_1 \leftarrow Gen_1(1^n)$ and $\mathrm{k}_2 \leftarrow Gen_2(1^n)$.
- $Mac_k(m)$: Outputs a tag $t = (t_1, t_2)$ where where $\mathrm{t}_1 \leftarrow {Mac_1}_{k_1}(m)$ and $\mathrm{t}_2 \leftarrow {Mac_2}_{k_2}(m)$.
- $Vrfy_k(m, t)$: Output ${Vrfy_1}_{k_1}(m, t_1) \wedge {Vrfy_2}_{k_2}(m, t_2)$

Proof of Security

- If an attacker A breaks Π then there exists two attackers A_1, A_2 such that A_1 breaks Π_1 and A_2 breaks Π_2.

Unforgeable Encryption

$\mathrm{EncForge}_{A,\Pi}(1^n)$

1. Sample $k \leftarrow$ Gen(1^n).
2. Let c^* be the output of $A^{Enc_k(\cdot)}(1^n)$. Let Q be the list of messages A gets ciphertexts for from the oracle.
3. Output 1 if $Dec_k(c^*) \notin \{\perp\} \cup Q$ and 0 otherwise.

$\Pi = (Gen, Enc, Dec)$ is unforgeable if

$\forall$ PPT A it holds that:

$\Pr[\mathrm{EncForge}_{A,\Pi} = 1] \leq$ negl(n)

Authenticated Encryption

- A private-key encryption scheme is an authenticated encryption scheme if it is CCA-secure and unforgeable.

Hard to come up with legitimate looking ciphertexts of new messages!

The power of decryption queries doesn't help!

Practice Problem 2 – Unforgeable Encryption Combiner

- Give Unforgeable encryption schemes $\Pi_1 = (Gen_1, Enc_1, Dec_1)$and $\Pi_2 = (Gen_2, Enc_2, Dec_2)$, is $\Pi = (Gen, Enc, Dec)$ below an unforgeable encryption as long as at least one of Π_1 and Π_2 is secure.
- $Gen(1^n)$: Outputs key $k = (k_1, k_2)$ where $\mathrm{k}_1 \leftarrow Gen_1(1^n)$ and $\mathrm{k}_2 \leftarrow Gen_2(1^n)$.
- $Enc_k(m)$: Outputs a tag $c = (c_1, c_2)$ where where $\mathrm{c}_1 \leftarrow Enc_{1_{k_1}}(r)$ and $\mathrm{c}_2 \leftarrow Enc_{2_{k_2}}(m \oplus r)$ where $r \leftarrow \{0,1\}^{|m|}$.
- $Dec_k(c)$: ??

Practice Problem 2 – Is this CPA secure?

- Yes!
- Proof: DIY

Practice Problem 2 – Unforgeable Encryption Combiner

- No!
- Adversary A given $c = (c_1, c_2)$ and $c' = (c_1', c_2')$ outputs a new ciphertext

$$c^* = (c_1, c_2')$$

Hash Function Definition

- Hash function $H: \{0,1\}^* \rightarrow \{0,1\}^\ell$
  - A collision is distinct x and x' such that $H(x) = H(x')$
- A hash function (with output length ℓ) is a pair of PPT algorithms (Gen, H) satisfying the following:
  - $Gen(1^n)$: Outputs s.
  - H: On input a key s and a string $x \in \{0,1\}^*$ output a string $H^s(x) \in \{0,1\}^{\ell(n)}$.

    s is public
- If H^s is defined only for inputs $\{0,1\}^{\ell'(n)}$ where $\ell'(n) > \ell(n)$, then (Gen, H) is a fixed-length hash function for inputs of length ℓ'.

Hash Function Security

$HashColl_{A,\Pi}(n)$

1. Sample $s \leftarrow \text{Gen}(1^n)$.
2. Let x, x' be the output of $A(1^n, s)$.
3. Output 1 if $x \neq x'$ and $H^s(x) = H^s(x')$ and 0 otherwise.

$\Pi = (Gen, H)$ is collision resistant if

$\forall$ PPT A it holds that:

$\Pr[HashColl_{A,\Pi}(n) = 1] \leq$ negl(n)

No secrets!

Practice Problem 3 – Hash Function Combiner

- Given $\Pi_1 = (Gen_1, H_1)$and $\Pi_2 = (Gen_2, H_2)$, is $\Pi = (Gen, H)$ a CRHF as long as at least one of Π_1 and Π_2 is a secure CRHF.
- $Gen(1^n)$: Outputs key $s = (s_1, s_2)$ where $\mathrm{s}_1 \leftarrow Gen_1(1^n)$ and $\mathrm{s}_2 \leftarrow Gen_2(1^n)$.
- $H_s(m)$: Outputs $h = (h_1, h_2)$ where where $\mathrm{h}_1 \leftarrow {H_1}_{s_1}(m)$ and $\mathrm{h}_2 \leftarrow {H_2}_{s_2}(m)$.

Practice Problem 3 – Hash Function Combiner

- If an attacker A breaks Π then there exists two attackers A_1, A_2 such that A_1 breaks Π_1 and A_2 breaks Π_2.
- Adversary A gives $H(m) = H\ (m')$ outputs
- Observe $H(m)\ =\ H_1(m), H_2(m)$
- Thus, $H_1(m), H_2(m) = H_1(m'), H_2(m')$
- (m, m') is a collision for both Π_1 and Π_2

Merkle Hash Construction

- Construct MH: $\{0,1\}^{2^\ell n} \to \{0,1\}^n$ from a hash function H: $\{0,1\}^{2n} \to \{0,1\}^n$

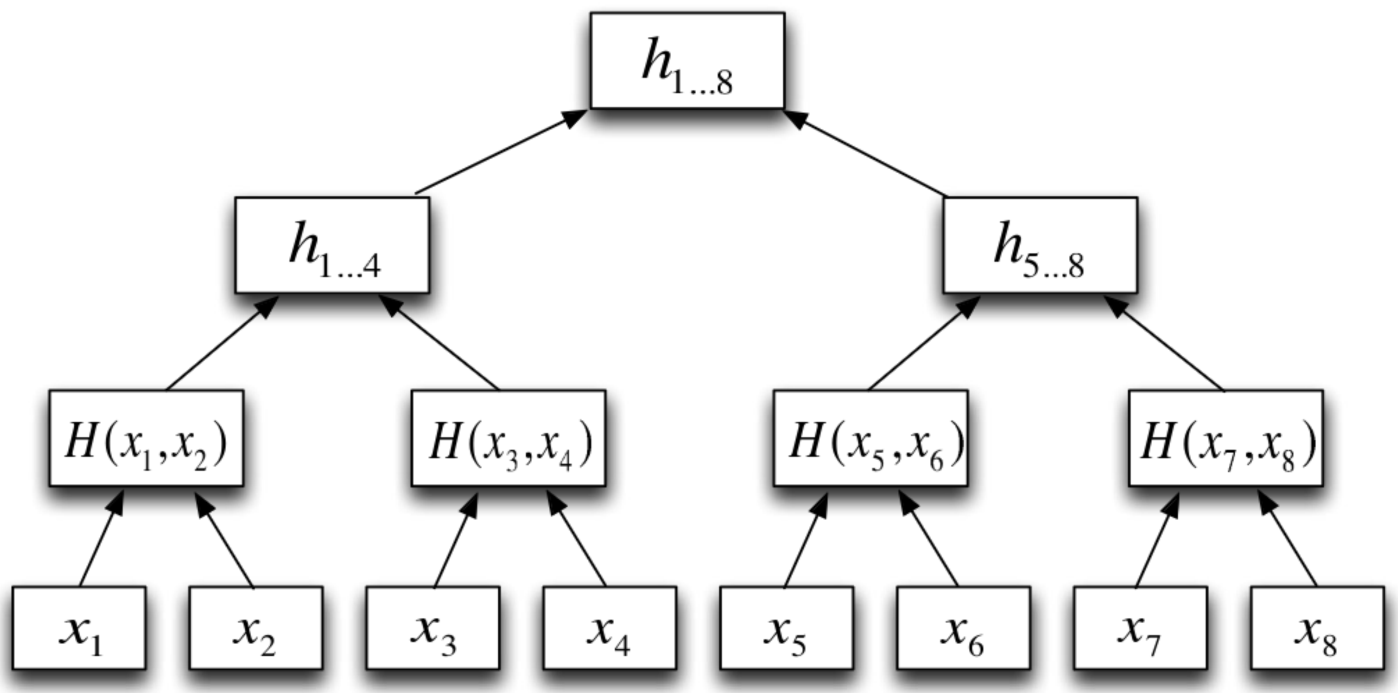


FIGURE 5.5: A Merkle tree.

Proof

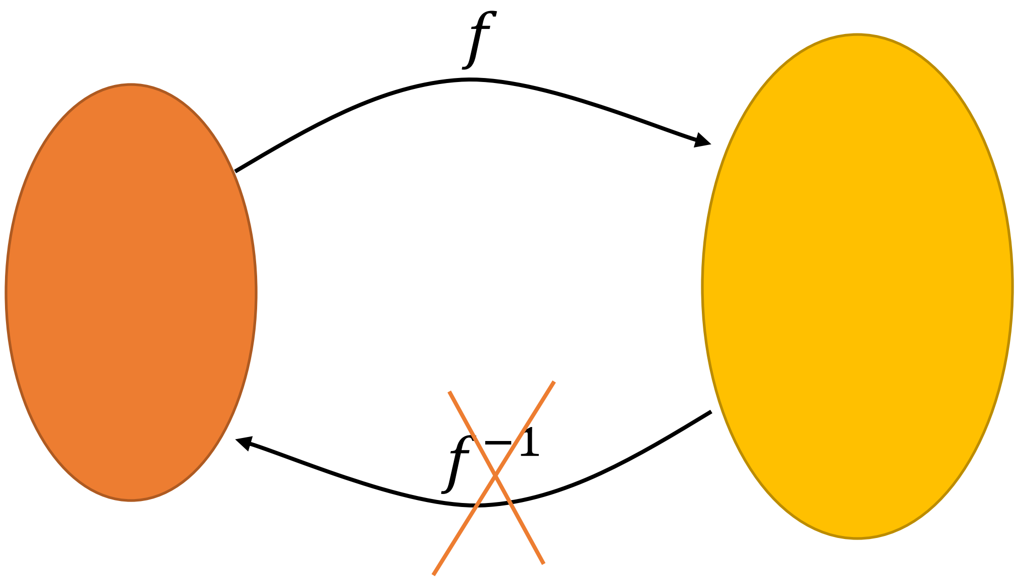
- If MH is not CRHF then H is not a CRHF.
- Given a collision $MH(M) \ = \ MH(M')$ such that $M \neq M'$
- We can find a collision for h from the two trees.

Merkle Hash Construction

- Alice/Prover and Bob/Verifier have access to Merkle Hash h
- Alice wants to prove to Bob that the i-th input value for hashing to $h = MH(\dots, m_i, ..)$ is m_i
- Alice can send $m_1, \dots m_\ell$ to Bob and it can verify that the hash was computed correctly and recover m_i
- Can Alice send something smaller?

Define: One-Way Functions

- A function f: $\{0,1\}^* \rightarrow \{0,1\}^*$ that is easy to compute but hard to invert



One-Way Functions: Formally

- A function f: $\{0,1\}^* \rightarrow \{0,1\}^*$ is a one-way function if:
- (easy to compute) There exists a polynomial-time algorithm M_f computing f; i.e., for all x, $M_f(\mathrm{x}) = f(x)$.
- (hard to invert) For all PPT A, there is a negligible function $negl$ such that

$$\Pr_{x \leftarrow \{0,1\}^n}\left[A\left(1^n, f(x)\right) \in f^{-1}\left(f(x)\right)\right] \leq negl(n)$$

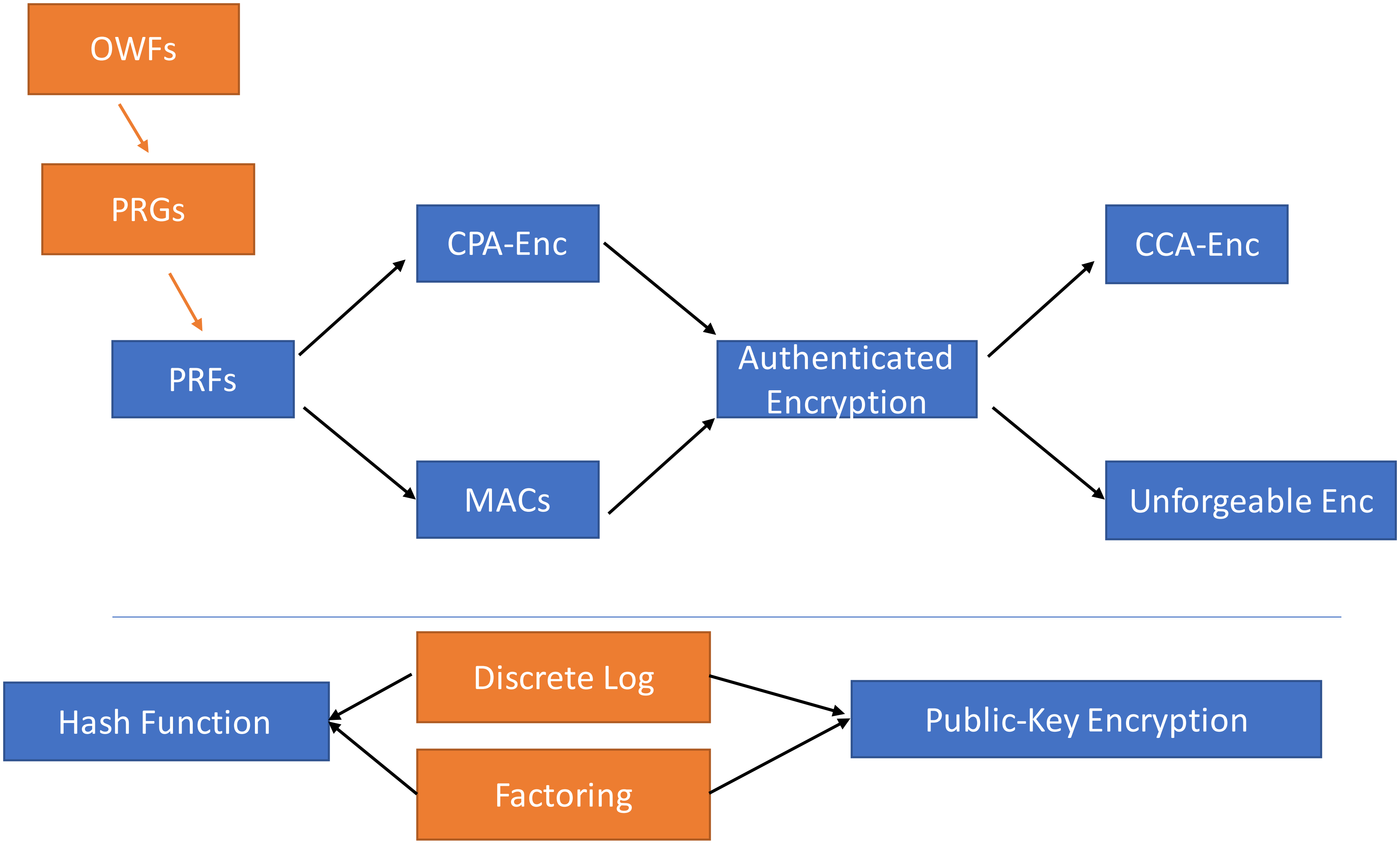
Practice Problem 4 – OWF Combiner

- If $f||g$ is a OWF as long as at least one of f and g is a OWF

$$f||g(\mathrm{x}, \mathrm{y}) = \mathrm{f}(\mathrm{x})||\mathrm{g}(\mathrm{y})$$

- Proof:

Implication Graph



Public-Key Encryption

- A public-key encryption scheme is a triple of PPT algorithms (Gen, Enc, Dec) such that:

1. $Gen(1^n) \rightarrow (pk, sk)$
2. $Enc(pk, m) \rightarrow c$
3. $Dec(sk, c) \rightarrow m/\perp$

- Correctness: For all (pk, sk) output by $Gen(1^n)$, we have that ∀ (legal) $m, Dec\left(sk, Enc(pk, m)\right) = m$
- Security: EAV-security, CPA-security?

EAV Security

$\mathrm{PubK}_{A,\Pi}^{\mathrm{eav}}(n)$

1. $(pk, sk) \leftarrow G(1^n)$ and give pk to A.
2. A outputs $m_0, m_1 \in \{0,1\}^*, |m_0| = |m_1|$.
3. b $\leftarrow$ {0,1}, $c \leftarrow Enc(pk, m_b)$
4. c is given to A and it outputs b'
5. Output 1 if $b = b'$ and 0 otherwise

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

$\forall$ PPT A it holds that:

$$\Pr\left[\mathrm{PubK}_{A,\Pi}^{\mathrm{eav}} = 1\right] \leq \frac{1}{2}$$

+ negl(n)

CCA Security

Much harder in the PKE setting.

$\mathrm{PubK}^{\mathrm{CCA}}_{\mathrm{A},\Pi}(n)$

1. $(pk, sk) \leftarrow G(1^n)$ and give pk to A.
2. $A^{Dec(sk,\cdot)}$ outputs $m_0, m_1 \in \{0,1\}^*, |m_0| = |m_1|$.
3. b $\leftarrow$ {0,1}, $c \leftarrow Enc(pk, m_b)$
4. c is given to $A^{Dec(sk,\cdot)}$ and it outputs b' (query c not allowed)
5. Output 1 if $b = b'$ and 0 otherwise

Encryption scheme $\Pi = (Gen, Enc, Dec)$ is indistinguishable in the presence of a CCA attacker, or is *CCA-secure* if

$\forall$ PPT A it holds that:

$$\Pr\left[\mathrm{PubK}^{\mathrm{cca}}_{\mathrm{A},\Pi} = 1\right] \leq \frac{1}{2}$$

+ negl(n)

ElGamal Encryption

Correctness?

1. $Gen(1^n) \to (pk, sk)$
    1. Run $\mathcal{G}(1^n)$ to obtain (G, g, q).
    2. Sample $x \leftarrow Z_q$ and set $h = g^x$
    3. Set $pk = (G, g, q, h)$ and $sk = x$.
2. $Enc(pk, m \in G) \to c = (c_1, c_2)$
    1. Parse $pk = (G, g, q, h)$
    2. Sample $r \leftarrow Z_q$ and set $c_1 = g^r$ and $c_2 = m \cdot h^r$
3. $Dec(sk, c) \to m/\perp$
    1. Parse $c = (c_1, c_2)$
    2. Output $\frac{c_2}{c_1^r}$

Security based on DDH!

Thank You!