Lecture 9: Signature Schemes

Notes by Yael Kalai

MIT - 6.5620
Lecture 9 (October 1, 2025)

> ***Warning:*** This document is a rough draft, so it may contain bugs. Please feel free to email me with corrections.

Recap

So far we saw how Alice and Bob can communicate securely, guaranteeing both secrecy and authenticity, and achieving the gold standard of CCA security. But all this required a strong assumption: That Alice and Bob share a secret key! What if they live in different countries and cannot share a secret key?

At first, it might seem like a shared secret key is necessary. After all, if Alice sends a message to Bob, and they don't have a secret, then what distinguishes the adversary from Bob? Nevertheless, we will see we can get secrecy and authenticity (and CCA security) without sharing a secret key!

Today

- Define the notion of a signature scheme, which is the public-key analogue of a MAC.
- Construct a one-time secure signature scheme (Lamport's one-time signature scheme).
- Introduce the Hash-then-Sign paradigm.

Definition of a signature scheme

Definition 1. A signature scheme is associated with a message space $\{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ and with three PPT algorithms $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$, with the following sytanx:

- Gen: Takes as input the security parameter 1^λ in unary and outputs a pair $(\mathsf{vk}, \mathsf{sk})$ of a public verification key and a secret signing key.
- Sign: Takes as input a secret signing key sk and a message $m \in \mathcal{M}_\lambda$ and outputs a signature σ.

- Ver: Takes as input a verification key vk, a message m and a signature σ and outputs 0/1, indicating accept or reject.

A signature scheme is required to satisfy the following completeness guarantee: For every $\lambda \in \mathbb{N}$ and every $m \in \mathcal{M}_\lambda$,

$$\Pr[\mathsf{Ver}(\mathsf{vk}, m, \mathsf{Sign}(\mathsf{sk}, m)) = 1] = 1$$

where the probability is over $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$ and over the randomness of Sign (if it is randomized).[1]

[1] Ver is always deterministic.

Definition 2. A signature scheme (Gen, Sign, Ver) with message space $\{\mathcal{M}_\lambda\}_{\lambda\in\mathbb{N}}$ is said to be existentially unforgeable against adaptive chosen message attacks if for every poly-size $\mathcal{A}$ there exists a negligible function μ such that for every $\lambda \in \mathbb{N}$, $\mathcal{A}$ wins in the following game with probability at most $\mu(\lambda)$:

1. The challenger samples $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$ and sends vk to $\mathcal{A}$.
2. $\mathcal{A}$ can choose a message $m_i \in \mathcal{M}_\lambda$ and obtain $\sigma_i \leftarrow \mathsf{Sign}(\mathsf{sk}, m_i)$.

   This step can be repeated polynomially many times.
3. $\mathcal{A}$ outputs (m^*, σ^*).

$\mathcal{A}$ wins if $m^* \notin \{m_i\}$ and $\mathsf{Ver}(\mathsf{vk}, m^*, \sigma^*) = 1$.

Remark. A more concise way to state this security definition is to say that for every poly-size $\mathcal{A}$ there exists a negligible function μ such that for every $\lambda \in \mathbb{N}$,

$$\Pr[\mathcal{A}^{\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{vk}) = (m^*, \sigma^*) \text{ s.t. } \mathsf{Ver}(\mathsf{vk}, m^*, \sigma^*) = 1 \;\wedge\; m^* \notin Q] \leq \mu(\lambda)$$

where Q denotes the set of all oracle calls that $\mathcal{A}$ makes to the oracle, and the probability is over $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$ and over the randomness of Sign (if it is randomized).

Remark. Note that this definition is very similar to the security definition we saw for MACs except that here the adversary is given the public verification key vk.

Lamport's One-Time Signatures

One of the magical things about signatures is that, even though they are public-key objects they can be constructed from the minimal assumption of one-way functions!

To see this, we will start by constructing a much simple object: a one-time signature scheme. This is a signature scheme with a much weaker security requirement. It is the same security requirement as above, except that the adversary is allowed to make only a single oracle call to the signing oracle. This seems like too weak of a security guarantee, since why would the adversary see only a single signature? Indeed, this will only serve as a stepping stone to our final construction.

Lamport's one-time signature scheme. Lamport constructed a very simple one-time secure signature scheme [1] from any one-way function

$$f : \{0,1\}^* \to \{0,1\}^*.$$

Let $\mathcal{M}_\lambda = \{0,1\}^n$ be the message space.

- $\mathsf{Gen}(1^\lambda)$ does the following:
  1. Sample at random $x_{1,0}, x_{1,1}, \ldots, x_{n,0}, x_{n,1} \leftarrow \{0,1\}^\lambda$.
  2. For every $i \in [n]$ and $b \in \{0,1\}$ compute $y_{i,b} = f(x_{i,b})$.
  3. Output $\mathsf{vk} = \{y_{i,b}\}_{i\in[n],b\in\{0,1\}}$ and $\mathsf{sk} = \{x_{i,b}\}_{i\in[n],b\in\{0,1\}}$.
- $\mathsf{Sign}(\mathsf{sk}, m)$ does the following:
  1. Parse $m = (m_1, \ldots, m_n)$
  2. Output $\sigma = (x_{1,m_1}, \ldots, x_{n,m_n})$.
- $\mathsf{Ver}(\mathsf{vk}, m, \sigma)$ does the following:
  1. Parse $\sigma = (x'_1, \ldots, x'_n)$.
  2. Output 1 if and only if for every $i \in [n]$ it holds that $y_{i,m_i} = f(x'_i)$.

Theorem 3. *This is a one-time secure signature scheme.*

Proof. It is easy to see that it satisfies the completeness guarantee. Hence we will focus on proving soundness. Suppose for contradiction that there exists a poly-size $\mathcal{A}$ and a non-negligible ϵ such that for every $\lambda \in \mathbb{N}$

$$\Pr[\mathcal{A}^{\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{vk}) = (m^*, \sigma^*) \text{ s.t. } \mathsf{Ver}(\mathsf{vk}, m^*, \sigma^*) = 1 \;\; \wedge \;\; m^* \neq m] \geq \epsilon(\lambda),$$

where m is the (single) query that $\mathcal{A}$ makes to its oracle, and where the probability is over $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.

We will construct a poly-size $\mathcal{B}$ that inverts the one-way function f with non-negligible probability. $\mathcal{B}$ on input $y = f(x)$, for $x \in \{0,1\}^\lambda$, does the following:

1. Sample at random $i^* \leftarrow [n]$ and $b^* \leftarrow \{0,1\}$.
2. For every $(i,b) \in ([n] \times \{0,1\}) \setminus (i^*, b^*)$ sample at random $x_{i,b} \leftarrow \{0,1\}^\lambda$ and let $y_{i,b} = f(x_{i,b})$.
3. Set $y_{i^*,b^*} = y$.

4. Let $\mathsf{vk} = \{y_{i,b}\}_{i\in[n],b\in\{0,1\}}$.
5. Emulate $\mathcal{A}^{\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{vk})$ by simulating its oracle as follows:
    - If $\mathcal{A}$ sends a message $m = (m_1, \ldots, m_n)$ such that $m_{i^*} = b^*$ then output $\perp$.
    - Otherwise, output $(x_{1,m_1}, \ldots, x_{n,m_n})$ which $\mathcal{B}$ knows since it does not include x_{i^*,b^*}, which is the preimage of the external y that $\mathcal{B}$ takes as input.
6. Let (m^*, σ^*) be the output of $\mathcal{A}$.
7. Output $\sigma^*_{i^*}$.

We next argue that

$$\Pr[f(\sigma^*_{i^*}) = y] \geq \frac{\epsilon(\lambda)}{2n}$$

To this end, we first note that

$$\Pr[m^*_{i^*} \neq m_{i^*} \quad \wedge \quad m^*_{i^*} = b^*] \geq \frac{1}{2n}$$

By our assumption

$$\Pr[f(\sigma^*_{i^*}) = y_{i^*,m^*_{i^*}}] \geq \epsilon(\lambda)$$

Therefore

$$\Pr[m^*_{i^*} \neq m_{i^*} \quad \wedge \quad m^*_{i^*} = b^* \quad \wedge \quad f(\sigma^*_{i^*}) = y_{i^*,b^*}] =$$
$$\Pr[m^*_{i^*} \neq m_{i^*} \quad \wedge \quad m^*_{i^*} = b^*] \cdot \Pr[f(\sigma^*_{i^*}) = y_{i^*,b^*} \mid m^*_{i^*} \neq m_{i^*} \quad \wedge \quad m^*_{i^*} = b^*] \geq$$
$$\frac{1}{2n} \cdot \epsilon(\lambda),$$

where the latter follows from the fact that (i^*, b^*) were sampled uniformly at random, and the distribution of vk is independent of (i^*, b^*), which in turn follows from the fact that $y = f(x)$ for a uniformly chosen $x \leftarrow \{0,1\}^\lambda$.

□

We note that the above scheme is not only one-time secure, but also the secret key is longer than the message to be signed. In what follows we show how to convert this scheme into a one-time secure one where the secret key is shorter than the message length. While this may seem like a minor issue, it will be an important stepping stone into constructing the final (many message secure) scheme.

Hash-then-Sign paradigm

One way to deal with long messages is to use a *collision resistant hash functions*.

Definition 4. A hash family is a family of functions $H = \{H_{\mathsf{hk}}\}$ associated with a PPT key generation algorithm Gen_H, such that the following two conditions hold:

- **Shrinking** For every $\lambda \in \mathbb{N}$ and every hk in the support of $\mathsf{Gen}_H(1^\lambda)$,
$$H_{\mathsf{hk}} : \{0,1\}^* \to \{0,1\}^\lambda.$$
- **Efficiency** There exists a poly-time algorithm that given $\mathsf{hk} \in \{0,1\}^\lambda$ and $x \in \{0,1\}^*$ outputs $H_{\mathsf{hk}}(x)$.

Definition 5. A hash family (H, Gen_H) is said to be *collision resistant* for every poly-size $\mathcal{A}$ there exists a negligible function μ such that

$$\Pr[\mathcal{A}(\mathsf{hk}) = (x, x') \text{ s.t. } x \neq x' \text{ and } H_{\mathsf{hk}}(x) = H_{\mathsf{hk}}(x')] \leq \mu(\lambda)$$

Remark. Note that in the above definition, the hash key hk is public, and we assume that it is known to the adversary. This is in contrast to a PRF where the key must remain secret to ensure any kind of security guarantees.

We next show how to use a collision resistant hash family to increase the message space to be $\mathcal{M}_\lambda = \{0,1\}^n$ for any $n = \text{poly}(\lambda)$. Specifically, given any signature scheme $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ with message space $\mathcal{M} = \{0,1\}^\lambda$, and given any collision resistant hash family (H, Gen_H), consider the following signature scheme, denoted by $(\mathsf{Gen}', \mathsf{Sign}', \mathsf{Ver}')$ with message space $\mathcal{M} = \{0,1\}^n$:

- Gen'**:** On input 1^λ, do the following:
  1. Sample $(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$.
  2. Sample $\mathsf{hk} \leftarrow \mathsf{Gen}_H(1^\lambda)$.
  3. Let $\mathsf{vk}' = (\mathsf{vk}, \mathsf{hk})$ and let $\mathsf{sk}' = (\mathsf{sk}, \mathsf{hk})$.
  4. Output $(\mathsf{vk}', \mathsf{sk}')$.
- Sign'**:** On input (sk', m) do the following:
  1. Parse $\mathsf{sk}' = (\mathsf{sk}, \mathsf{hk})$.
  2. Output $\mathsf{Sign}(\mathsf{sk}, H_{\mathsf{hk}}(m))$.
- Ver'**:** On input $(\mathsf{vk}', m, \sigma)$ do the following:
  1. Parse $\mathsf{vk}' = (\mathsf{vk}, \mathsf{hk})$.
  2. Output $\mathsf{Ver}(\mathsf{vk}, H_{\mathsf{hk}}(m), \sigma)$.

Theorem 6. *If* $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ *is a signature scheme with message space* $\{0,1\}^\lambda$ *that is existentially unforgeable against one-time (resp., many-time) adaptive chosen message attacks then* $(\mathsf{Gen}', \mathsf{Sign}', \mathsf{Ver}')$ *is a signature scheme with message space* $\{0,1\}^n$ *that is existentially unforgeable against one-time (resp., many-time) adaptive chosen message attacks.*

Proof. Suppose for the sake of contradiction that there exists a poly-size adversary $\mathcal{A}$ and a non-negligible function ϵ such that for every $\lambda \in \mathbb{N}$,

$$\Pr[\mathcal{A}^{\mathsf{Sign}'(\mathsf{sk},\cdot)}(\mathsf{vk}') = (m^*, \sigma^*) \text{ s.t. } \mathsf{Ver}'(\mathsf{vk}', m^*, \sigma^*) = 1 \quad \wedge \quad m^* \notin Q] \geq \epsilon(\lambda)$$

where Q is the query set that $\mathcal{A}$ sends to its oracle.[2] Denote by $Q = \{m_i\}_{i=1}^{\ell}$. We distinguish between two cases:

- **Case 1:** There exists a non-negligible function δ such that for every $\lambda \in \mathbb{N}$,

  $$\Pr[\exists i \in [\ell] \text{ s.t. } H(m^*) = H(m_i)] \geq \delta(\lambda).$$

  In this case we can use $\mathcal{A}$ to break the collision resistant property of (H, Gen_H).

- **Case 2:** There exists a negligible function μ such that for every $\lambda \in \mathbb{N}$,

  $$\Pr[\exists i \in [\ell] \text{ s.t. } H(m^*) = H(m_i)] \leq \mu(\lambda).$$

  In this case we can use $\mathcal{A}$ to break the security of the underlying signature scheme $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$.

□

[2] $|Q| = 1$ in the case of one-time security and $|Q| = \text{poly}(\lambda)$ in the case of many-time security.

References

[1] Leslie Lamport. Constructing digital signatures from a one-way function. Technical Report CSL-98, SRI International, Computer Science Laboratory, Menlo Park, CA, October 1979. Technical Report.