Lecture 18: Message Authentication Codes & Digital Signatures

MACs and Signatures

- Both are used to assert that a message has indeed been generated by a party
- MAC is the private-key version and Signatures are public-key version
- Note: Message hiding is not part of the (intuitive) security requirements

Message Authentication Codes

- Defined by $(\mathsf{Gen}, \mathsf{Tag}, \mathsf{Ver})$ algorithms
- The signer and the verifier meet to generate a secret key $\mathsf{sk} \sim \mathsf{Gen}(1^n)$
- The signer sends a message $m \in \{0,1\}^n$ along with a tag $\tau \sim \mathsf{Tag}_{\mathsf{sk}}(m)$ (Note that the tag generation algorithm can be randomized)
- The verifier, upon receiving $(\widetilde{m}, \widetilde{\tau})$ verifies using $\mathsf{Ver}_{\mathsf{sk}}(\widetilde{m}, \widetilde{\tau}) \in \{0,1\}$

Correctness and Security

- Correctness: For any message $m \in \{0,1\}^n$,

$$\Pr[\mathsf{sk} \sim \mathsf{Gen}(1^n) \colon \mathsf{Ver}_{\mathsf{sk}}(m, \mathsf{Tag}_{\mathsf{sk}}(m)) = 1] = 1$$

- Security: For any adversary $\mathcal{A}$ the following holds

$$\Pr\left[\mathsf{sk} \sim \mathsf{Gen}(1^n) \colon \begin{array}{c} \mathcal{A}^{\mathsf{Tag}_{\mathsf{sk}}(\cdot)} = (m', \tau') \ \wedge \\ m' \notin Q \ \wedge \ \mathsf{Ver}_{\mathsf{sk}}(m', \tau') = 1 \end{array}\right] \leqslant \mathsf{negl}(n),$$

  where Q is the set of all queries made to the tagging oracle $\mathsf{Tag}_{\mathsf{sk}}(\cdot)$ by the adversary $\mathcal{A}$

- Note: If the security is restricted to $|Q| = k$, then it is called a k-time secure MAC

1-time MAC

- Computational assumptions are not necessary for MACs
- Consider the following construction of 1-time MAC
  - $\mathrm{Gen}(1^n)$ samples $r^{(b,i)} \xleftarrow{\$} \{0,1\}^k$, where $b \in \{0,1\}$ and $i \in [n]$, and outputs:

$$\mathrm{sk} = \begin{pmatrix} r^{(0,1)} & r^{(0,2)} & \dots & r^{(0,i)} & \dots & r^{(0,n)} \\ r^{(1,1)} & r^{(1,2)} & \dots & r^{(1,i)} & \dots & r^{(1,n)} \end{pmatrix}$$

  - $\mathrm{Tag}_{\mathrm{sk}}(m)$ outputs

$$\tau = r^{(m_1,1)} r^{(m_2,2)} \dots r^{(m_n,n)}$$

  - $\mathrm{Ver}_{\mathrm{sk}}(\widetilde{m}, \widetilde{\tau})$ outputs 1 if and only if all the following tests pass:

$$r^{(\widetilde{m}_i,i)} = \widetilde{\tau}_i$$

Proof of Security

- Suppose an adversary $\mathcal{A}$ queries the tagging oracle at $m^{(1)}$ and gets outputs $\tau^{(1)}$
- Then it outputs (m', τ'), where $m' \neq m^{(1)}$
- If $m' \neq m^{(1)}$ then there exists i such that $m'_i \neq m^{(1)}_i$
- So, the probability that (m', τ') is a valid signature is at most the probability of guessing $r^{(m'_i, i)}$, which is at most 2^{-k}
- So, for $k = \omega(\log n)$ this is a secure 1-time MAC scheme

(poly-time) MAC using One-way Functions

- We will construct it using Pseudo-Random Functions (PRFs) and we already know that PRFs can be constructed from OWFs
- Suppose $\{f_1, \dots, f_{k(n)}\}$ is a PRF family
- Consider the following scheme:
  - $\mathsf{Gen}(1^n)$ samples $\mathsf{sk} \xleftarrow{\$} \{1, \dots, K(n)\}$
  - $\mathsf{Tag}_{\mathsf{sk}}(m) = f_{\mathsf{sk}}(m)$
  - $\mathsf{Ver}_{\mathsf{sk}}(\widetilde{m}, \widetilde{\tau}) = 1$ if and only if $f_{\mathsf{sk}}(\widetilde{m}) = \widetilde{\tau}$
- Use the intuition to prove its security:
  - PRF family is computationally indistinguishable from the family of random functions
  - Given evaluation of a random function at some points Q, it is impossible to predict the output of the function at $m' \neq Q$

Digital Signatures

- Defined by (Gen, Sign, Ver) algorithms
- Correctness:

$$\Pr[(\mathsf{sk}, \mathsf{pk}) \sim \mathsf{Gen}(1^n) \colon \mathsf{Ver}_{\mathsf{pk}}(m, \mathsf{Sign}_{\mathsf{sk}}(m)) = 1] = 1$$

- Security:

$$\Pr\left[(\mathsf{sk}, \mathsf{pk}) \sim \mathsf{Gen}(1^n) \colon \begin{array}{c} \mathcal{A}^{\mathsf{Sign}_{\mathsf{sk}}(\cdot)}(\mathsf{pk}) = (m', \sigma') \ \wedge \\ m' \notin Q \ \wedge \ \mathsf{Ver}_{\mathsf{pk}}(m', \sigma') = 1 \end{array}\right] \leqslant \mathsf{negl}(n)$$

1-time Digital Signatures using OWF: Lamport Scheme

- Let $f\colon \{0,1\}^n \to \{0,1\}^n$ be a OWF
- $\mathsf{Gen}(1^n)$ samples $r^{(b,i)} \stackrel{\$}{\leftarrow} \{0,1\}^n$, for $b \in \{0,1\}$ and $i \in \{1,\dots,n\}$ and computes $y^{(b,i)} = f(r^{(b,i)})$. Output

$$\mathsf{sk} = \begin{pmatrix} r^{(0,1)} \dots r^{(0,i)} \dots r^{(0,n)} \\ r^{(1,1)} \dots r^{(1,i)} \dots r^{(1,n)} \end{pmatrix}$$

$$\mathsf{pk} = \begin{pmatrix} y^{(0,1)} \dots y^{(0,i)} \dots y^{(0,n)} \\ y^{(1,1)} \dots y^{(1,i)} \dots y^{(1,n)} \end{pmatrix}$$

- $\mathsf{Enc}_{\mathsf{sk}}(m)$ outputs $\sigma = r^{(m_1,1)} \dots r^{(m_n,n)}$
- $\mathsf{Ver}_{\mathsf{pk}}(\widetilde{m}, \widetilde{\sigma})$ outputs 1 if and only if all the following tests pass

$$y^{(\widetilde{m}_i,i)} = f(\widetilde{\sigma}_i), where$$

Security Proof

- Suppose $\mathcal{A}^*$ breaks the Lamport Scheme with probability ε
- Following is the code of $\widetilde{\mathcal{A}}$ on input y:
  - Prepare sk and pk of Lamport Scheme
  - Pick random $i^* \xleftarrow{\$} [n]$ and $b^* \xleftarrow{\$} \{0,1\}$
  - Replace $y^{(i^*,b^*)}$ by y in the public-key pk
  - Send pk to $\mathcal{A}^*$
  - Receive m from $\mathcal{A}^*$
  - If $b^* = m_{i^*}$, then stop executing $\mathcal{A}^*$ and return 0 to the external honest challenger of OWF experiment (this corresponds to the case where we need to know an inverse of y to prepare the signature of m)
  - Otherwise, generate the signature σ on m using the secret key sk
  - Obtain (m', σ') from $\mathcal{A}^*$
  - If (m', σ') is not a valid message and signature pair, then return 0 to the external honest challenger of OWF experiment (this corresponds to the case that the adversary $\mathcal{A}^*$ failed to produce a forgery)
  - Otherwise, if $b^* = m'_{i^*}$, i.e. $\mathcal{A}^*$ has helped us invert y (think why this is the case), then return σ'_{i^*} (This is the inverse of y)

Security Proof

Note that we succeed if we satisfy the following conditions:

- $b^* = m'_{i^*}$ but $b^* \neq m_{i^*}$
- $\mathcal{A}^*$ successfully breaks the signature scheme.

Conditioned on $\mathcal{A}^*$ successfully breaking the signature scheme, the probability that random (i^*, b^*) satisfy the first condition is $1/2n$. So, overall probability of successfully inverting the OWF f is $\varepsilon/2n$

General Signatures

Think & Read about the following:

- Signing Arbitrary length messages: Use CRHF family that hash arbitrary length messages (Merkle-Damgård Construction) and the "Hash-then-sign" Paradigm
- Signatures where an adversary can ask for arbitrary-polynomial number of signatures of messages of its choice