A weaker notion of security
Lamport's one-time signature scheme

Foundations of Cryptography
Computer Science Department
Wellesley College

Fall 2016

Table of contents

Security of signature schemes

Let $\Pi = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ be a signature scheme.

The signature experiment $\text{Sig-forge}_{\mathcal{A},\Pi}(n)$:

1. $\mathsf{Gen}(1^n)$ is run to obtain keys (pk, sk).
2. Adversary $\mathcal{A}$ is given pk and oracle access to $\mathsf{Sign}_{sk}(\cdot)$. The adversary then outputs (m, α). Let $\mathcal{Q}$ denote the set of messages whose signatures were requested by $\mathcal{A}$ during its execution.
3. The output of the experiment is defined to be 1 if and only if (1) $\mathsf{Vrfy}_{pk}(m, \alpha) = 1$, and (2) $m \notin \mathcal{Q}$

Definition 12.2. A signature scheme $\Pi = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ is *existentially unforgeable under an adaptive chosen-message attack* if for all PPT adversaries $\mathcal{A}$, there exists a negligible function negl such that

$$\Pr[\text{Sig-forge}_{\mathcal{A},\Pi}(n) = 1] \leq \mathsf{negl}(n).$$

Good enough by golly

- Definition 12.2 is the gold standard of security for digital signature schemes.
- However, weaker notions of security may be appropriate for certain restricted application and as building blocks for scheme satisfying stronger notions of security.
- Today we study a that is secure as long as it's only used to sign one message.

One last experiment

The one-time signature experiment $\text{Sig-forge}^{1-time}_{\mathcal{A},\Pi}(n)$.

1. $\text{Gen}(1^n)$ is run to obtain keys (pk, sk).
2. Adversary $\mathcal{A}$ is given pk and asks a *single* query m' to oracle $\text{Sign}_{pk}(\cdot)$. $\mathcal{A}$ then outputs (m, α) where $m \neq m'$.
3. The output of the experiment is defined to be 1 if and only if $\text{Vrfy}_{pk}(m, \alpha) = 1$.

Definition 12.14 A signature scheme $\Pi = (\text{Gen}, \text{Sign}, \text{Vrfy})$ is existentially unforgeable under a single-message attack, or is a one-time signature scheme, if for all PPT adversaries $\mathcal{A}$, there exists a negligible function negl such that:

$$\Pr[\text{Sig-forge}^{1-time}_{\mathcal{A},\Pi}(n) = 1] \leq \text{negl}(n).$$

One-way functions: a formal introduction

The inverting experiment $\text{Invert}_{\mathcal{A},f}(n)$*:*

1. Choose input $x \leftarrow \{0,1\}^n$. Compute $y := f(x)$.
2. $\mathcal{A}$ is given 1^n and y as input, and outputs x'.
3. The output of the experiment is defined to be 1 if and only if $f(x') = y$.

Definition 8.72. A function $f : \{0,1\}^* \to \{0,1\}^*$ is one-way if the following two conditions hold:

1. (Easy to compute:) There exists a polynomial-time algorithm that on input x outputs $f(x)$.
2. (Hard to invert:) For all probabilistic polynomial-time algorithms $\mathcal{A}$ there exists a negligible function negl such that

$$\Pr[\text{Invert}_{\mathcal{A},f}(n) = 1] \leq \text{negl}(n).$$

Lamport scheme used to sign message the 3-bit message 011

Let f be a one-way function, that is, f is easy to compute, but difficult to invert.

Signing $m = 011$:

$$sk = \begin{pmatrix} \boxed{x_{1,0}} & x_{2,0} & x_{3,0} \\ x_{1,1} & \boxed{x_{2,1}} & \boxed{x_{3,1}} \end{pmatrix} \Rightarrow \alpha = (x_{1,0}, x_{2,1}, x_{3,1})$$

Verifying for $m = 011$ and $\alpha = (x_1, x_2, x_3)$:

$$pk = \begin{pmatrix} \boxed{y_{1,0}} & y_{2,0} & y_{3,0} \\ y_{1,1} & \boxed{y_{2,1}} & \boxed{y_{3,1}} \end{pmatrix} \Rightarrow \begin{array}{l} f(x_1) \stackrel{?}{=} y_{1,0} \\ f(x_2) \stackrel{?}{=} y_{2,1} \\ f(x_3) \stackrel{?}{=} y_{3,1} \end{array}$$

Construction 12.15 of Lamport scheme

Let f be a one-way function. Construct a signature scheme for messages of length $\ell = \ell(n)$ as follows:

- Gen: On input 1^n, proceed as follows for $i \in \{0, \dots \ell\}$:
  1. Choose random $x_{i,0}, x_{i,1} \leftarrow \{0,1\}^n$.
  2. Compute $y_{i,0} := f(x_{i,0})$ and $y_{i,1} := f(x_{i,1})$.

  The public key pk and the private key sk are

$$pk := \begin{pmatrix} y_{1,0} & y_{2,0} & \cdots & y_{\ell,0} \\ y_{1,1} & y_{2,1} & \cdots & y_{\ell,1} \end{pmatrix} \quad sk := \begin{pmatrix} x_{1,0} & x_{2,0} & \cdots & x_{\ell,0} \\ x_{1,1} & x_{2,1} & \cdots & x_{\ell,1} \end{pmatrix}$$

- Sign: On input a private key sk and a message $m \in \{0,1\}^\ell$ with $m = m_1 \dots m_\ell$, output the signature $(x_{1,m_1}, \dots, x_{\ell,m_\ell})$.
- Vrfy: On input a public key pk, a message $m \in \{0,1\}^\ell$ with $m = m_1 \dots m_\ell$, and a signature $\alpha = (x_1, \dots, x_\ell)$, output 1 if and only if $f(x_i) = y_{i,m_i}$ for all $1 \le i \le \ell$.

Security of Lamport scheme

Theorem 12.16. Let ℓ be any polynomial. If f is a one-way function, then Construction 12.7 is a one-time signature scheme for message of length ℓ.

Proof. Let Π denote the Lamport scheme. Let $\mathcal{A}$ be a PPT adversary, and define

$$\epsilon(n) \stackrel{\text{def}}{=} \Pr[\text{Sig-forge}^{1-time}_{\mathcal{A},\Pi}(n) = 1].$$

In a particular execution, let m' denote the message whose signature is requested by $\mathcal{A}$, and let (m, α) denote the final output of $\mathcal{A}$.

We say that $\mathcal{A}$ outputs a forgery at (i, b) if $\mathsf{Vrfy}_{pk}(m, \alpha) = 1$ and $m_i = b \neq m'_i$. Note that if $\mathcal{A}$ outputs a forgery, then it outputs a forgery at some (i, b).

The evil, wicked, mean, bad, and nasty adversary

Algorithm $\mathcal{I}$:
This algorithm is given y and 1^n as input.

1. Choose random $i^* \leftarrow \{1, \dots, \ell\}$ and $b^* \leftarrow \{0, 1\}$. Set $y_{i^*,b^*} := y$.
2. For all $i \in \{1, \dots, \ell\}$ and $b \leftarrow \{0, 1\}$ with $(i, b) \neq (i^*, b^*)$:
   - Choose $x_{i,b} \leftarrow \{0, 1\}^n$ and set $y_{i,b} := f(x_{i,b})$.
3. Run $\mathcal{A}$ on input $pk := \begin{pmatrix} y_{1,0} & y_{2,0} & \cdots y_{\ell,0} \\ y_{1,1} & y_{2,1} & \cdots y_{\ell,1} \end{pmatrix}$.
4. When $\mathcal{A}$ requests a signature on the message m':
   - If $m'_{i^*} = b^*$, stop.
   - Otherwise, return the correct signature $\alpha = (x_{1,m'_1}, \dots, x_{\ell,m'_\ell})$.
5. When $\mathcal{A}$ outputs (m, α) with $\alpha = (x_1, \dots, x_p)$:
   - If $\mathcal{A}$ outputs a forgery at (i^*, b^*), then output x_{i^*}.

Analysis of $\mathcal{A}$'s chances

$\mathcal{A}$'s changes of success. When $\mathcal{A}$ outputs a forgery at (i^*, b^*), algorithm $\mathcal{I}$ succeeds in inverting its given input y. What are the chances of that happening with x is chosen at random and $y := f(x)$?

Thought experiment. Imagine an experiment in which $\mathcal{I}$ is given x at the outset, sets $x_{i^*,b^*} := x$, and then always returns a signature to $\mathcal{A}$ in step 4.* Then the view of $\mathcal{A}$ being run as a subroutine by $\mathcal{I}$ is distributed identically to the view of $\mathcal{A}$ in experiment $\text{Sig-forge}^{1-time}_{\mathcal{A},\Pi}(n)$ and the probability $\mathcal{A}$ outputs a forgery is ϵ. Now the probability the $\mathcal{A}$ outputs a forgery at (i^*, b^*), conditioned on the fact that $\mathcal{A}$ outputs a forgery is at least $1/2\ell(n)$. We conclude that the probability $\mathcal{A}$ outputs a forgery at (i^*, b^*) in our thought experiment is at least $\epsilon/2\ell(n)$

*Even if $m'_{i^*} = b^*$.

Analysis of $\mathcal{A}$'s chances continued

Returning to the real experiment we note that *the probability that $\mathcal{A}$ outputs a forgery at i^*, b^* is unchanged* since the experiments only differ if $\mathcal{A}$ requests a signature on a message m' with $m'_{i^*} = b^*$. But in that case, $\mathcal{A}$ throws up its hands.

We conclude: In the real experiment

$$Pr[\text{Invert}_{\mathcal{I},f}(n) = 1] \geq \epsilon(n)/2\ell(n).$$

But f is one-way, so the left hand term is bounded by a negligible function, and hence so is ϵ. $\square$

*At last a provably secure signature scheme**

- *Corollary 12.17.* If one-way functions exist, then for any polynomial ℓ there exists a one-time signature scheme for message of length ℓ.
- Good time to stop?

*Well, provable mostly.