Sigs. with special properties

Fast one-time signatures and applications
One-time signatures: definition

Suppose signing key is used to sign a single message
Can we give a simple (fast) construction $SS=(\text{Gen}, S, V)$?

Chal. (pk, sk)$ \leftarrow \text{Gen}$

pk $\rightarrow$

$m_1 \in M$

$\sigma_1 \leftarrow S(sk,m_1)$

Adv. A

(m, σ)$ \rightarrow$

A wins if $V(pk, m, \sigma) = \text{`accept’}$ and $m \neq m_1$

Security: for all “efficient” A, $\text{Adv}_{1-SIG}[A, SS] = Pr[ A \text{ wins }] \leq \text{negl}$
Application: fast online signatures

1. Next section: secure one-time sigs $\Rightarrow$ secure many-time sigs

2. Fast online signatures: signing can be slow on a weak device

   Goal:
   – Do heavy signature computation **before** message is known
   – Quickly output signature once user supplies message
Fast online signing using one-time sigs

(\text{Gen}, S, V): secure many-time signature (slow)

(\text{Gen}_{1T}, S_{1T}, V_{1T}): secure one-time signature (fast)

- Gen $\rightarrow$ (pk,sk)
- PreSign(sk): $(pk_{1T}, sk_{1T}) \leftarrow \text{Gen}_{1T}$, $\sigma \leftarrow S(sk, pk_{1T})$
- $S_{\text{online}}(\sigma, sk_{1T}, pk_{1T}, m)$: $\sigma_{1T} \leftarrow S_{1T}(sk_{1T}, m)$
  
  output $\sigma^* \leftarrow (pk_{1T}, \sigma, \sigma_{1T})$

- $V_{\text{online}}(pk, m, \sigma^*=(pk_{1T}, \sigma, \sigma_{1T}))$:
  
  accept if $V(pk, pk_{1T}, \sigma) = V_{1T}(pk_{1T}, m, \sigma_{1T}) = \text{“accept”}$

Dan Boneh
Sigs. with special properties

Constructing fast one-time signatures
One-time signatures

Goal: one-time sigs from fast **one-way functions** (OWF)

- $f: X \rightarrow Y$ is a OWF if
  1. $f(x)$ is efficiently computable,
  2. hard to invert on random $f(x)$

- Examples: 
  1. $f(x) = \text{AES}(x, 0^{128})$, 
  2. $f(x) = \text{SHA256}(x)$
Lamport one-time signatures

\[ f: X \rightarrow Y \text{ a one-way function.} \quad \text{Msg space: } M = \{0,1\}^{256} \]

\[ \text{Gen: generate } 2 \times 256 \text{ random elements in } X \]

\[ f \]

\[ f \]

\[ f \]

\[ f \]
Lamport one-time signatures (simple)

\[ f : X \rightarrow Y \text{ a one-way function.} \quad \text{Msg space: } M = \{0,1\}^{256} \]

Gen: generate 2\times256 random elements in X

\[ \text{sk} \]

\[ m = 0 \quad 1 \quad 1 \quad \ldots \quad 0 \quad 1 \]

\[ S(\text{sk}, \, m): \quad \sigma = \text{(pre-images corresponding to bits of } m\text{)} \]
Lamport one-time signatures (simple)

\[ f: X \rightarrow Y \text{ a one-way function.} \quad \text{Msg space: } M = \{0,1\}^{256} \]

Gen: generate $2 \times 256$ random elements in $X$

\[ \sigma = \left[ \begin{array}{cccc} \text{ } & \text{ } & \text{ } & \text{ } \\ \text{ } & \text{ } & \text{ } & \text{ } \\ \text{ } & \text{ } & \text{ } & \text{ } \\ \text{ } & \text{ } & \text{ } & \text{ } \\ \text{ } & \text{ } & \text{ } & \text{ } \\ \end{array} \right] \in X^{256} \quad (2\text{KB}) \]

\[ m = 0 \quad 1 \quad 1 \quad \cdots \quad 0 \quad 1 \]

\[ S(sk, m): \quad \sigma = (\text{pre-images corresponding to bits of } m) \]

Dan Boneh
Lamport one-time signatures (simple)

\[
f: X \rightarrow Y \text{ a one-way function.} \quad \text{Msg space: } M = \{0,1\}^{256}
\]

**Gen:** generate 2×256 random elements in X

\[
\sigma = \left[ \begin{array}{c}
\vdots \\
\sigma_0 \sigma_1 \sigma_2 \sigma_3 \vdots
\end{array} \right] 
\in X^{256} \quad (2\text{KB})
\]

\[
m = \left[ \begin{array}{c}
0 \\
1 \\
1 \\
\vdots \\
0 \\
1
\end{array} \right]
\]

\[
V( pk, m, \sigma ) : \text{ accept if all pre-images in } \sigma \text{ match values in } pk
\]
Very fast signature system. Will prove one-time security in a bit.

Not two-time secure:

The attacker can ask for a signature on $0^{128}$ and on $1^{128}$. He gets all of $sk$ which he can use to sign new messages.
Abstraction: cover free set systems

Sets: $S_1, S_2, \ldots, S_{2^{256}} \subseteq \{1, \ldots, n\}$

Def: $S = \{S_1, S_2, \ldots, S_{2^{256}}\}$ is cover-free if $S_i \not\subseteq S_j$ for all $i \neq j$

Example: if all sets in $S$ have the same size $k$ then $S$ is cover free
Abstract Lamport signatures

\( f: X \rightarrow Y \) a one-way function.  

Msg space: \( M = \{0,1\}^{256} \)

\( S = \{S_1, S_2, \ldots, S_{2^{256}}\} \) is **cover-free** over \( \{1,\ldots,n\} \)

\( H: \{0,1\}^{256} \rightarrow S \) a bijection (one-to-one)

**Gen:** generate \( n \) random elements in \( X \)

\[ \begin{align*}
\text{sk} & \in X^n \\
\text{pk} & \in Y^n
\end{align*} \]
Abstract Lamport signatures

f: X → Y a one-way function. Msg space: M = \{0,1\}^{256}

\( S = \{S_1, S_2, \ldots, S_{2^{256}}\} \) is cover-free over \{1,..,n\}

H: \{0,1\}^{256} → S a bijection (one-to-one)

Gen: generate n random elements in X

\[ \sigma = \]

\[ 1 \quad \ldots \quad n \quad pk \in Y^n \]

S(sk, m): \( \sigma = \left( \text{pre-images corresponding to elements of } H(m) \right) \)
Why cover free?

Suppose $S$ were not cover free

$\Rightarrow$ exists $m_1, m_2$ such that $H(m_1) \subset H(m_2)$

$\Rightarrow$ signature on $m_2$ gives signature on $m_1$

$S(sk, m): \sigma = \left( \text{pre-images corresponding to elements of } H(m) \right)$
Security statement

**Thm:** if \( f: X \rightarrow Y \) is one-way and \( S \) is cover-free then Lamport signatures (Lam) are one-time secure.

\[
\forall A \ \exists B: \ \text{Adv}_{1-SIG}[A, \text{Lam}] \leq n \cdot \text{Adv}_{\text{OWF}}[B, f]
\]

**Proving security:**

Diagram:
- **y = f(x)**
- **pk**
- **m_1**
- **σ_1**
- **(m, σ)**
Proving security

$y = f(x)$

choose: $i \leftarrow \{1, \ldots, n\}$

$x_1, \ldots, x_n \leftarrow X$

$pk = f(x_1) \ldots f(x_{i-1}) y f(x_{i+1}) \ldots f(x_n)$

\[
\begin{cases}
  i \notin H(m_1) & \implies \text{we (alg. B) can generate } \sigma_i \\
  i \in H(m) & \implies \sigma \text{ from adv. reveals pre-image } x
\end{cases}
\]

$\implies$ B wins if $i \in H(m)$ but $i \notin H(m)$
Proving security

\[ y = f(x) \]

choose: \( i \leftarrow \{1, \ldots, n\} \)
\[ x_1, \ldots, x_n \leftarrow X \]

\[ \text{pk} = \begin{array}{c}
  f(x_1) \\
  \cdots \\
  f(x_{i-1}) \\
  y \\
  f(x_{i+1}) \\
  \cdots \\
  f(x_n)
\end{array} \]

\[ S \text{ cover free} \implies \exists i^* \text{ s.t. } i^* \notin H(m) \text{ but } i^* \in H(m) \]
\[ \Pr[i = i^*] \geq \frac{1}{n}. \]

So: \( \text{Adv}_{\text{orf}}[B, f] = \Pr[i = i^*] \cdot \text{Adv}_{\text{sig}}[A, \text{Lamport}] \)
\[ \geq \frac{1}{n} \cdot \text{Adv}_{\text{sig}}[A, \text{Lamport}] \]

Dan Boneh
Parameters \((f: X \rightarrow Y \text{ where } X = Y)\)

\[ S = \{ S_1, S_2, \ldots, S_{2^{256}} \} \text{ is cover-free over } \{1,..,n\} \]

In particular: \( S = \{ \text{all subsets of } \{1,..,n\} \text{ of size } k \} \)

\[ pk \in Y^n \implies \text{pk size} = (n \text{ elements of } Y) \]
\[ \text{sig. size} = (k \text{ elements of } X) \]

\[ \text{Msg-space} = \{0,1\}^{256} \implies |S| = (n \text{ choose } k) \geq 2^{256} \]

- To shrink signature size, choose small \(k\)
  
  example: \(k=32 \implies n \geq 3290\)

- For optimal \((\text{sig-size} + \text{pk-size})\) choose \(n = 261, k = 123\)
  
  \((\text{sig-size} + \text{pk-size}) \approx 1.5 \times 256 \text{ elements of } X \) \(3\text{KB}\)
Further improvement: Winternitz

Gen: generate $n$ random elements in $X$: 

$$\text{Gen}: \text{generate } n \text{ random elements in } X: \quad (f: X \rightarrow X)$$

depth $d=4$

$$\begin{cases}
\text{for } i = 1, \ldots, n:
\quad \text{choose } x_i \in X \\
\quad f(x_i) \leftarrow x_i \\
\quad \text{for } j = 1, \ldots, d:
\quad \text{compute } x_{i+j} = f(x_i)
\end{cases}$$

$$\begin{cases}
\text{for } i = 1, \ldots, n:
\quad x_i \in X \\
\quad f(x_i) \leftarrow x_i \\
\quad \text{compute } x_{i+d} = f(x_i)
\end{cases}$$

$$\begin{cases}
\text{for } i = 1, \ldots, n:
\quad x_i \in X \\
\quad f(x_i) \leftarrow x_i \\
\quad \text{compute } x_{i+2d} = f(x_i)
\end{cases}$$

$$\begin{cases}
\text{for } i = 1, \ldots, n:
\quad x_i \in X \\
\quad f(x_i) \leftarrow x_i \\
\quad \text{compute } x_{i+3d} = f(x_i)
\end{cases}$$

$$\begin{cases}
\text{for } i = 1, \ldots, n:
\quad x_i \in X \\
\quad f(x_i) \leftarrow x_i \\
\quad \text{compute } x_{i+4d} = f(x_i)
\end{cases}$$

sk $\in X^n$

pk $\in X^n$
Further improvement: Winternitz

\[ H: \{0,1\}^{256} \rightarrow \{0,1,\ldots,d-1\}^n \]

\[
\begin{align*}
S(\sk, \m): \quad & \sigma = \left( \text{pre-images indicated by } H(\m) \right) \\
\end{align*}
\]
Further improvement: Winternitz

\[ H: \{0,1\}^{256} \rightarrow \{0,1,\ldots,d-1\}^n \]

ex: \[ H(0^{256}) = (2, 1, 3, 0, \ldots, 0, 1) \]

\[
\begin{align*}
S(sk, m): \quad \sigma &= \left( \text{pre-images indicated by } H(m) \right) \\
\end{align*}
\]
For what $H$ is this a secure one-time signature?

Suppose $H(0^{256}) = (2, 1, 3, 0, 0, 1)$
$H(1^{256}) = (2, 2, 3, 1, 1, 2)$

Is the signature one-time secure?

- No, from a sig. on $0^{256}$ one can construct a sig. on $1^{256}$
- No, from a sig. on $1^{256}$ one can construct a sig. on $0^{256}$
- Yes, the signature is one-time secure
- It depends on how $H$ behaves at other points
Optimized parameters

For one-time security need that:

for all \( m_0 \neq m_1 \) we have \( H(m_0) \) does not “cover” \( H(m_1) \)

**Parameters:**

- Time(sign) = Time(verify) = \( O(n \cdot d) \)
- \( pk \) size = sig. size = (n elements in X)
- msg-space = \( \{0,1\}^{256} \Rightarrow n > 256 / \log_2(d) \) (approx.)

\[(pk \text{ size})+(\text{sig. size}) \approx 256 \times (2/\log_2(d)) \text{ elems. of } X\]

For Lamport: \( (pk \text{ size})+(\text{sig. size}) \approx 256 \times (1.5) \text{ elems. of } X\)
Sigs. with special properties

One-time signatures \Rightarrow many-time signatures
Review

One-time signatures need not be 2-time secure
example: Lamport signatures

Goal: convert any one-time signature into a many-time signature

Main tool: collision resistant hash functions
Construction

$(\text{Gen}_{1T}, S_{1T}, V_{1T})$: secure one-time signature (fast)

Four-time signature: (stateful version)

- **Gen:**

\[
\text{Gen}_{1T} \rightarrow (\text{pk}_{0123}, \text{sk}_{0123})
\]

\[
(\text{pk}_{01}, \text{sk}_{01}) \quad (\text{pk}_{23}, \text{sk}_{23})
\]

\[
(\text{pk}_{0}, \text{sk}_{0}) \quad (\text{pk}_{1}, \text{sk}_{1}) \quad (\text{pk}_{2}, \text{sk}_{2}) \quad (\text{pk}_{3}, \text{sk}_{3})
\]
Construction

(Gen_{1T}, S_{1T}, V_{1T}): secure one-time signature (fast)

Four-time signature: (stateful version)

- **Gen:**

  \[
  \begin{align*}
  & (pk_{0123}, \sigma_{0123}) \\
  \rightarrow & S_{1T}(sk, (pk_{01}, pk_{23}))
  \end{align*}
  \]

  \[
  \begin{align*}
  & (pk_{01}, sk_{01}) \quad (pk_{23}, sk_{23}) \\
  \quad & (pk_{0}, sk_{0}) \quad (pk_{1}, sk_{1}) \quad (pk_{2}, sk_{2}) \quad (pk_{3}, sk_{3})
  \end{align*}
  \]

Dan Boneh
Construction

\((\text{Gen}_{1T}, S_{1T}, V_{1T})\): secure one-time signature (fast)

Four-time signature: (stateful version)

- **Gen:**

\[
\begin{align*}
(pk_{01}, \sigma_{01}) & \quad (pk_{23}, \sigma_{23}) \\
(pk_0, sk_0) & \quad (pk_1, sk_1) & \quad (pk_2, sk_2) & \quad (pk_3, sk_3)
\end{align*}
\]
Construction

\((\text{Gen}_{1T}, S_{1T}, V_{1T})\): secure one-time signature (fast)

Four-time signature: (stateful version)

Sig. on msg \(m_0\):

\[
(\sigma_{0123}, \sigma_{01}, \sigma_0, \text{pk}_0, \text{pk}_{23}, \text{pk}_0, \text{pk}_1)
\]
Construction

\((\text{Gen}_{1T}, S_{1T}, V_{1T})\): secure one-time signature (fast)

Four-time signature: (stateful version)

Sig. on msg \(m_1\):

\[
(\sigma_{0123}, \sigma_{01}, \sigma_1, \text{pk}_{01}, \text{pk}_{23}, \text{pk}_0, \text{pk}_1)
\]
Construction

(\text{Gen}_{1T}, S_{1T}, V_{1T})$: secure one-time signature (fast)

Four-time signature: (stateful version)

Sig. on msg $m_2$:

\[
(\sigma_{0123}, \sigma_{23}, \sigma_2, pk_{01}, pk_{23}, pk_2, pk_3)
\]
Construction

$(\text{Gen}_{1T}, S_{1T}, V_{1T})$: secure one-time signature (fast)

Four-time signature: (stateful version)

Sig. on msg $m_3$:

$$(\sigma_{0123}, \sigma_{23}, \sigma_3, pk_{01}, pk_{23}, pk_2, pk_3)$$
More generally: $2^d$-time signature

Tree of depth $d$:
- Every signature contains $d+1$ one-time signatures along with associated pk’s

Tree is generated on-the-fly:
- Signer stores only $d$ secret keys at a time

Stateful signature:
- Signer maintains a counter indicating which leaf to use for signature
- Every leaf must only be used once!
Optimized $2^d$-time signatures

Combined with Lamport signatures:
• collision resistant hash funs $\Rightarrow$ many-time signature

With further optimizations:
• For $2^{40}$ signatures: (stateful) signature size is $\approx$ 5KB
  ... signing time is about the same as RSA signatures

• Recall: RSA sig size is 256 bytes  (2048 bit RSA modulus)
THE END