Strengthened Security for Blind Signatures

David Pointcheval
David.Pointcheval@info.unicaen.fr

GREYC
Université de Caen

Summary

- Blind Signatures
  - Definition
  - Security
- Previous Results
- New Scheme
  - Presentation
  - Security
- Conclusion

Blind Signatures

An authority helps a user to get a valid signature

the message and the signature
must remain unknown for the authority

$\Longrightarrow$ (revokable) anonymity

- electronic cash schemes
- electronic voting
- ...

Security Properties

- $(\ell, \ell+1)$-**forgery:** after ℓ interactions with the authority the attacker can forge $\ell+1$ message–signature valid pairs.

Attacks

- **Sequential attack:** the attacker interacts sequentially with the signer.
- **Parallel attack:** the attacker can initiate several interactions at the same time with the signer, in any order he wants.

Previous Results

- **Complexity-Based Security:** at last Crypto, [JLO-97] proved the existence of secure schemes using secure signature schemes
  and multi-party computation

  $\Longrightarrow$ totally inefficient, and even impractical.

- **Random Oracle Model:** [PS-96] proposed first proofs for witness-indistinguishable-based schemes (WI is needed for simulation of the signer).

Okamoto–Schnorr Blind Scheme

Authority – Σ		Alice
public : p, q, g, h, y		
secret : $y = g^{-r} h^{-s} \bmod p$		
$t, u \in (\mathbb{Z}/q\mathbb{Z})^\star$		
$a = g^t h^u \bmod p$	$\xrightarrow{\quad a \quad}$	
		$\beta, \gamma, \delta \in \mathbb{Z}/q\mathbb{Z}$
		$\alpha = a g^\beta h^\gamma y^\delta \bmod p$
		$\varepsilon = H(m, \alpha)$
	$\xleftarrow{\quad e \quad}$	$e = \varepsilon - \delta \bmod q$
$R = t + er \bmod q$		
$S = u + es \bmod q$	$\xrightarrow{\quad R, S \quad}$	
		$g^R h^S y^e \stackrel{?}{=} a \bmod p$
		$\rho = R + \beta \bmod q$
		$\sigma = S + \gamma \bmod q$

$(m, \alpha, \varepsilon, \rho, \sigma)$ s.t. $\alpha = g^\rho h^\sigma y^\varepsilon \bmod p$ with $\varepsilon = H(m, \alpha)$.

Security Result [PS-96]

If $\mathcal{A}$ is a PPTM which can perform an $(\ell, \ell+1)$-forgery, under a parallel attack,

- after Q queries to the random oracle,
- with probability $\varepsilon \geq 4Q^{\ell+1}/q$.

The Discrete Logarithm Problem can be solved

- after 2 calls to $\mathcal{A}$
- with probability greater than

$$\frac{1}{4\ell} \times \left(\frac{\varepsilon}{12\ell Q^{\ell+1}}\right)^3.$$

Remark: there are less than $Q^{\ell+1}$ possibilities to choose $\ell+1$ hash values among Q.

Extension

(Extension of the non-uniform reduction of [P-96])

If $\mathcal{A}$ is a PPTM which can perform an $(\ell, \ell+1)$-forgery, under a parallel attack,

- after Q queries to the random oracle,
- after R initiated interactions, (but only ℓ ended ones),
- with probability $\varepsilon \geq 4Q^{\ell+1}R^{\ell}/q$.

The Discrete Logarithm Problem can be solved

- after $33Q\ell/\varepsilon$ calls to $\mathcal{A}$
- with probability greater than $\frac{1}{72\ell^2}$.

Remark: there are less than $Q^{\ell+1} \times R^{\ell}$ possibilities to choose
$\ell+1$ hash values among Q
and ℓ ended interactions among R initiated ones.

Asymptotically

k is the security parameter.

If $|q| = k$ and $\ell \ll k/\log k$,
for any polynomial P, Q and A,

$4Q^{\ell+1}R^{\ell}/q \leq 1/A$, for k large enough.

$\Longrightarrow$ ℓ poly-logarithmically bounded.

Generic Transformation

It is a kind of "cut-and-choose":

- we duplicate everything except the final answer;
- we ask the user to commit its "blinding" factors;
- after the 2 queries:
  the authority randomly chooses one, $I \in_R \{0, 1\}$
  and checks its well-construction
  then answers the other query, e_{1-I}.

Authority		Alice
public: $p,\ q,\ g,\ h,\ y$ secret: $y = g^{-r}h^{-s} \bmod p$		
		for $i = 0, 1$, $\beta_i, \gamma_i, \delta_i \in \mathbb{Z}/q\mathbb{Z}$ ϕ_i, ψ_i random, $\mu_i = H(m, \phi_i)$ $h_i = H(\beta_i, \gamma_i, \delta_i, \mu_i, \psi_i)$
	$\xleftarrow{h_0, h_1}$	
for $i = 0, 1$, $t_i, u_i \in \mathbb{Z}/q\mathbb{Z}$ $a_i = g^{t_i}h^{u_i} \bmod p$	$\xrightarrow{a_0, a_1}$	
		for $i = 0, 1$, $\alpha_i = a_i g^{\beta_i} h^{\gamma_i} y^{\delta_i} \bmod p$ $e_i = H(\mu_i, \alpha_i) - \delta_i \bmod q$
	$\xleftarrow{e_0, e_1}$	
$I \in \{0, 1\}$	$\xrightarrow{I}$	
	$\xleftarrow{\beta_I, \gamma_I, \delta_I, \mu_I, \psi_I}$	
Verification of h_I and e_I $R = t_{1-I} + e_{1-I} \cdot r \bmod q$ $S = u_{1-I} + e_{1-I} \cdot s \bmod q$	$\xrightarrow{R, S}$	
		$a_{1-I} \stackrel{?}{=} g^R h^S y^{e_{1-I}} \bmod p$ $\rho = R + \beta_{1-I} \bmod q$ $\sigma = S + \gamma_{1-I} \bmod q$

Then $\alpha = g^\rho h^\sigma y^\varepsilon \bmod p$, $\mu = H(m, \phi)$ and $\varepsilon = H(\mu, \alpha)$
where $\alpha = \alpha_{1-I}$ and $\phi = \phi_{1-I}$

Claim

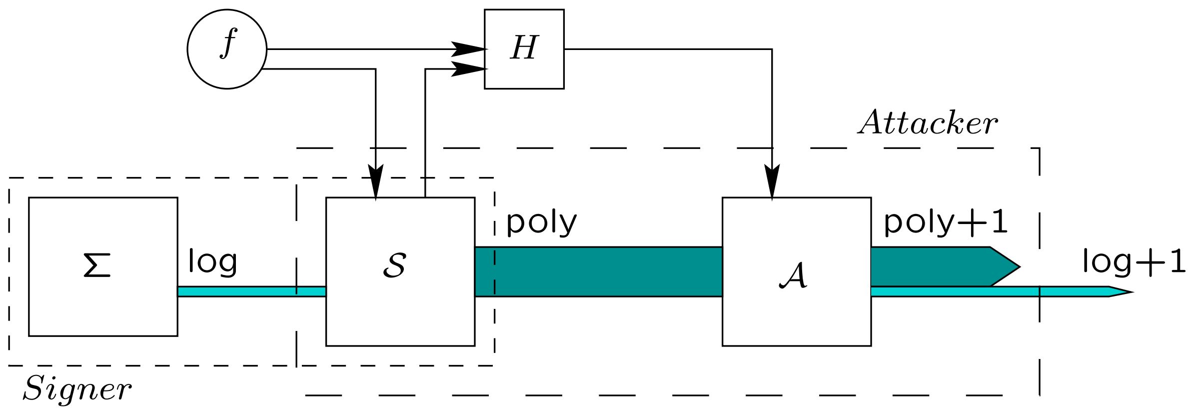
- **Synchronized Parallel Attack:** the attacker can initiate several interactions at the same time with the signer, but for each round, indexes follow the same order.

  seq. attack $<$ synchr. parallel attack $<$ parallel attack

- **Security:** If there exist polynomials ℓ, Q and P, and a PPTM $\mathcal{A}$ which can perform an $(\ell, \ell+1)$-forgery, under a **synchronized parallel attack**,
  - after Q queries to the random oracle,
  - with probability $\varepsilon \geq \mathbf{1/P}$.

  The Discrete Logarithm Problem can be solved
  - after $\mathcal{O}(\log \mathbf{k})\mathbf{Q}/\varepsilon$ calls to $\mathcal{A}$
  - with probability greater than $\mathbf{\Omega(1/(\log k)^2)}$.

Reduction



- New scheme — *Signer*: signer; $\mathcal{A}$: attacker
- OS scheme — Σ: signer; *Attacker*: attacker
- $\mathcal{S}$ Simulator
- f random oracle
- H $\mathcal{S}$-controled random oracle

$\mathcal{S}$:

- $\mathcal{A}$ sends h_0 and h_1;
- $\mathcal{S}$ randomly chooses $i \in \{0, 1\}$:
  1. $\mathcal{S}$ begins an alone simulation: a_{1-i}, challenge w
     $\mathcal{S}$ looks, in the table of f, for j: $h_{1-i} = \rho_j$.
     j exists: $\mathcal{Q}_j = (\beta, \gamma, \delta, \mu, \psi) \implies \alpha$
     $\mathcal{S}$ defines $H(\mu, \alpha) = w + \delta$ and $E_{1-i} = w$.
     Otherwise, it lets $E_{1-i} = \infty$;
  2. $\mathcal{S}$ asks to Σ: a_i
     As above: $\mathcal{Q}_j = (\beta, \gamma, \delta, \mu, \psi)$, $\implies \alpha$
     and define $E_i = f(\mu, \alpha) - \delta$, or $E_i = \infty$;
  3. It sends a_0 and a_1 to $\mathcal{A}$;
- $\mathcal{A}$ sends the challenges e_0 and e_1;
- If $(e_0, e_1) = (E_0, E_1)$ then $\mathcal{S}$ defines $I = i$, asks I;
  else it lets $I = 1 - i$.
- $\mathcal{A}$ answers $\beta', \gamma', \delta', \mu', \psi'$;
- $\mathcal{S}$ checks whether $h_I = f(\beta', \gamma', \delta', \mu', \psi')$.
  `False`: $\mathcal{S}$ stops the game;
  `True`: if $I = i$
  then $\mathcal{S}$ ends its simulation
  else $\mathcal{S}$ sends $\Sigma(e_{1-I}) = (R, S)$.

Properties

Let us assume that $\mathcal{A}$ can perform an $(\ell, \ell+1)$-forgery against $Signer$ under a **synchronized parallel attack** for ℓ polynomially bounded.

The number of initiated interactions with Σ is equal to ℓ.
We denote by λ the number of complete interactions with Σ.

1. $\mathcal{A}$ cannot distinguish $\mathcal{S} \cup \Sigma$ from $Signer$;
2. The number of valid signatures (w.r.t. f) is greater than $\lambda + 1$;
3. With probability greater than 1/16, $\lambda \leq \log(4/\varepsilon)$

Property 1

$\mathcal{A}$ cannot distinguish $\mathcal{S} \cup \Sigma$ from $Signer$:

- a_0 and a_1 follow an identical distribution;
- H looks like a random oracle,
  except if some (μ, α) has yet been asked to f.
  This occurs with probability less than $Q\ell/q$;
- the challenge "I" is equal to $i \oplus v$,
  where $i \in_R \{0, 1\}$ and $v = [(e_0, e_1) = (E_0, E_1)]$.
  (v is independent of i).

Property 2

The number of really valid signatures is greater than $\lambda + 1$:

$$\varepsilon_i = H(\mu_i, \alpha_i) \neq f(\mu_i, \alpha_i) \implies \mathcal{S} \text{ imposed } \varepsilon_i = w + \delta$$
$$\text{Then } g^{\rho_i - \beta} h^{\sigma_i - \gamma} = a y^{-w} = g^u h^v$$

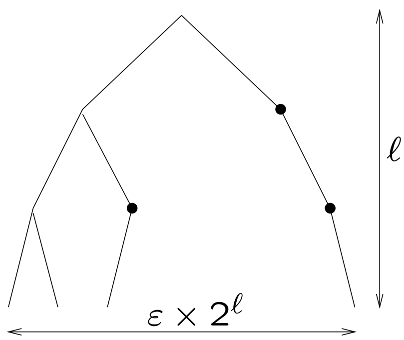
- either $\mathcal{A}$ received (u, v) from $\mathcal{S}$;
- or $\mathcal{A}$ had computed ρ_i and σ_i from ay^{-w}:
  with probability greater than $1/q$, $\rho_i \neq u + \beta \implies \log_g h$

$\implies \mathcal{S}$ has simulated everything (otherwise we have $\log_g h$).

$$\#\{\text{valid signatures}\} = \ell + 1 - \#\{\varepsilon_i \neq f(\mu_i, \alpha_i)\} \geq \ell + 1 - (\ell - \lambda) \geq \lambda + 1.$$

Property 3

λ is logarithmically bounded:



$2^3 = 2^0 \cdot 1 + 2^1 \cdot 1 + 2^2 \cdot 1$

$2^\ell = \sum_i 2^i \times \#\{\text{paths with } i \bullet\}$

Then $\#\{\text{paths } \geq s \bullet\} \leq 2^{\ell - s}$
$\implies$ Pr[more than s $\bullet$ $|OK] \leq 2^{-s}/\varepsilon$

Help of $\Sigma \implies (e_0, e_1) \neq (E_0, E_1)$
$\implies$ single node (or collision for f).

So Pr[less than $\log(2/\varepsilon)$ $\bullet$ $|OK] \geq 1/2$.

Consequences

- Assumption: $\mathcal{A}$ can perform an $(\ell, \ell+1)$-forgery
  against $Signer$ under a synchronized parallel attack
  - after Q queries to the random oracle,
  - with probability ε.
- Consequence: $\mathcal{S} \cup \mathcal{A}$ can perform an $(\lambda, \lambda+1)$-forgery
  against Σ under a parallel attack
  - after Q queries to the random oracle,
  - after ℓ initiated interactions
    but only $\lambda \leq \log(4/\varepsilon)$ ended ones
  - with probability $\varepsilon' \geq \varepsilon/16$.

As soon as $\varepsilon \geq 1/P$, for any k large enough,

$$\varepsilon' \geq \varepsilon/16 \geq 4Q^{\lambda+1}\ell^{\lambda}/q$$

Then the DLP can be solved

- with probability greater then $\Omega(1/(\log k)^2)$
- after less than $\mathcal{O}(\log k)Q/\varepsilon$ steps.

Conclusion

With a kind of cut-and-choose,
we impose the user to play honestly.

A dishonest user will be detected
before it is too late.

We have presented a generic transformation which

- makes secure:
  after poly. many synchronized interactions
  with poly-log. many attackers.
- lets practical and efficient.
  the output signature is an OS signature

This transformation can be adapted
to any other WI-based blind signature schemes.