ID-based signatures: review

Alexey Vyskubov

16.02.2003

- ID-based cryptography: Shamir, 1985
- ID-based encryption scheme, based on pairings: Boneh and Franklin, 2001
- ID-based signature scheme, based on pairings: Sakai, Ohgishi and Kasahara, 2000 — but no security argument was given

$\boxed{\text{ID, TTP} \longrightarrow \text{Signature function}}$

To be able to sign a message one should use trusted third party services to get the signature function...

$\boxed{\text{ID} \longrightarrow \text{Signature verification function}}$

...but signature verification function can be easily obtained from signer ID.

Article 1:

Kenneth G. Paterson

ID-based signatures from pairings on elliptic curves

G_1 is an additive group of order q.

G_2 is a multiplicative group of order q.

$\hat{e} : G_1 \times G_1 \to G_2$ is a bilinear map.

$P \in G_1$ is a known element such that $\hat{e}(P, P) \neq 1_{G_2}$.

Discrete logarithm problem should be hard in G_1 and in G_2.

H_1, H_2, H_3 are public cryptographic hash functions:

$$
\begin{tikzcd}
ID \in \{0,1\}^* \arrow[r, "H_1"] \arrow[d, "H_2"] & Q_{ID} \in G_1 \arrow[d, "H_3"] \\
Z_q \arrow[r, equal] & Z_q
\end{tikzcd}
$$

- $Q_{ID} = H_1(ID)$ — public key for signature verification.
- $s \in Z_q$ — master key known to TTP.
- $D_{ID} = sQ_{ID}$ — secret key for signature generation.
- $M \in \{0,1\}^*$ is a message to sign.

We also assume that $P_{pub} = sP$ is publicly known.

$$
\begin{array}{ccccc}
M \in \{0,1\}^* & \xrightarrow{a=H_2(M)} & Z_q \times Z_q & \xleftarrow{b=H_3(R)} & R = kP \in G_1 \\
& & {\scriptstyle S=aP+bD_{ID}}\Big\downarrow & & \Big\uparrow \\
& & (kP, k^{-1}S) \in G_1 \times G_1 & \longleftarrow & k \in Z_q^*
\end{array}
$$

k is chosen at random in Z_q^*.

Reminder:

$$D_{ID} = sQ_{ID} = sH_2(M)$$

$$P_{pub} = sP$$

Let us suppose that verifier wants to check the signature (U, V).
Verifier calculates $\hat{e}(U, V)$.

$$\hat{e}(U, V) = \hat{e}(kP, k^{-1}S) = \hat{e}(P, S) =$$

$$= \hat{e}(P, aP + bD_{ID}) = \hat{e}(P, aP) \cdot \hat{e}(P, bD_{ID}) =$$

$$= \hat{e}(P, P)^a \cdot \hat{e}(P, D_{ID})^b =$$

$$= \hat{e}(P, P)^{H_2(M)} \cdot \hat{e}(P, sQ_{ID})^b =$$

$$= \hat{e}(P, P)^{H_2(M)} \cdot \hat{e}(sP, Q_{ID})^{H_3(kP)} =$$

$$= \hat{e}(P, P)^{H_2(M)} \cdot \hat{e}(P_{pub}, Q_{ID})^{H_3(kP)}$$

$$\hat{e}(U, V) = \hat{e}(P, P)^{H_2(M)} \cdot \hat{e}(P_{pub}, Q_{ID})^{H_3(kP)}$$

If (U, V) is a valid signature then $U = kP$ and

$$\hat{e}(U, V) = \hat{e}(P, P)^{H_2(M)} \cdot \hat{e}(P_{pub}, Q_{ID})^{H_3(U)}$$

(U, V) is a valid signature $\Longleftrightarrow \forall l \in Z_q^* \ (lU, l^{-1}V)$ is a valid signature.

Signature generation

$$\boxed{\mathsf{signature} = (kP; k^{-1}(H_2(M)P + H_3(kP)D_{ID})}$$

- two hash function evaluations
- computation in G_1
- inversion modulo q

Signature verification

$$\boxed{\hat{e}(U, V) = \hat{e}(P, P)^{H_2(M)} \cdot \hat{e}(P_{pub}, Q_{ID})^{H_3(U)}}$$

$\hat{e}(P, P)$ can be precalculated.

$\hat{e}(P_{pub}, Q_{ID})$ can be precalculated for given ID.

- $\hat{e}(U, V)$ — one pairing
- two hash function evaluations
- two exponentiations in G_2
- one multiplication in G_2

"... the security of our identity-based scheme is linked to the security of an ordinary signature scheme which resembles a well-known scheme."

Article 2:

Jae Choon Cha and Jung Hee Cheon

An identity-based signature from Gap Diffie-Hellman Groups

$G =< P >$; order of G is a prime q; group operations can be computed in a unit time.

G is a **Gap Diffie-Hellman group** if DDHP can be solved in a polynomial time but no probabilistic algorithm can solve CDHP with non-negligible advantage within polynomial time.

Choose random $s \in Z_q^*$ — master key.

$P_{pub} = sP$

$H_1 : \{0,1\}^* \times G \to Z_q$

$H_2 : \{0,1\}^* \to G$

H_1 and H_2 are viewed as random oracles.

(P, P_{pub}, H_1, H_2) — system parameters.

$D_{ID} = sH_2(ID)$ — private key, associated with ID.

$Q_{ID} = H_2(ID)$ — plays the role of public key.

M — a message to sign.

$$\begin{tikzcd}
M \in \{0,1\}^* \arrow[r] & h = H_1(M,U) \in Z_q \arrow[r] & V = (k+h)D_{ID} \\
& U = kQ_{ID} \in G \arrow[u] & k \in Z_q^* \arrow[l] \arrow[u]
\end{tikzcd}$$

$$\textsf{signature} = (U, V)$$

k is chosen at random in Z_q^*.

Verifier checks that $(P, P_{pub}, U + H_1(M, U)Q_{ID}, V)$ is a valid Diffie-Hellman tuple.

Let $h = H_1(M, U)$. Signature correct $\Rightarrow$

$$U = kQ_{ID}$$

$$V = (k + h)D_{ID} = s(k + h)Q_{ID}$$

$$(P, P_{pub}, U + H_1(M, U)Q_{ID}, V) = (P, sP, (k + h)Q_{ID}, s(k + h)Q_{ID})$$

The game

1. $\mathcal{C}$ runs Setup and gives system parameters to $\mathcal{A}$.
2. $\mathcal{A}$ issues the following queries as it wants:
   (a) Hash query.
   (b) $\mathsf{Extract}$ query.
   (c) Sign query.
3. $\mathcal{A}$ outputs (ID, M, σ).

Given ID model

$\mathcal{A}$ is given ID and should forge the signature for it.

Reduction to the given ID case:

$\mathcal{A}_0$ runs in time t_0 and has advantage ε_0 making q_{H_2} queries to H_2.

$\mathcal{A}_1$ runs in time $t_1 \leq t_0$ and has advantage

$$\varepsilon_1 \geq \frac{\varepsilon_0 \left(1 - \frac{1}{q}\right)}{q_{H_2}}$$

The idea: to substitute the given ID instead of one (random) ID_i in $\mathcal{A}_0$'s queries.

Reduction to CDHP:

$\mathcal{A}_1$ runs in time t_1 making at most q_{H_1}, q_{H_2}, q_S and q_E queries to H_1, H_2, Sign and $\mathsf{Extract}$ and has an advantage

$$\varepsilon_1 \geq \frac{10\,(q_S + 1)\,(q_S + q_{H_1})}{q}.$$

CDHP can be solved with probability $\varepsilon_2 \geq 1/9$ within running time $t_2 \leq 23 q_{H_1} t_1 / \varepsilon_1$.