

# The Provable Security of Pairing-Based Protocols: Pairing-Based Short Signatures and their Extensions

Benoît Libert

Microelectronics Laboratory, Crypto Group  
Université Catholique de Louvain (Belgium)

`benoit.libert@uclouvain.be`

August 29<sup>th</sup> 2007

# Outline

- Bilinear map groups: properties and various configurations
- Security definition for signature schemes
- The Boneh-Lynn-Shacham short signature and its extensions
  - multisignatures
  - aggregate and verifiably encrypted signatures
- Short signatures in the standard model
  - short group signatures and beyond
- The Waters signature and its extensions
  - multisignatures, sequential aggregate signatures
  - forward-secure signatures with additional protection

# 1 Bilinear map groups

## Basic properties:

- Triple of groups  $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ , all of prime order  $p$ .
- There is a mapping  $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$  such that:
  - $e(g \cdot g', h) = e(g, h) \cdot e(g', h)$
  - $e(g, h \cdot h') = e(g, h) \cdot e(g, h')$
  - Hence

$$e(g^a, h^b) = e(g, h)^{ab} = e(g^b, h^a) = \dots$$

- Non-degeneracy:  $e(g, h) \neq 1_{\mathbb{G}_T}$  whenever  $g \neq 1_{\mathbb{G}_1}$  or  $h \neq 1_{\mathbb{G}_2}$ .
- Computability:  $e(g, h)$  can be efficiently computed.

## Various configurations

For prime order groups  $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$  admitting a bilinear mapping  $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ , we distinguish three cases:

- **Case I:** symmetric  $\mathbb{G}_1 = \mathbb{G}_2$  (can *only* be implemented with supersingular curves).
- **Case II :** asymmetric  $\mathbb{G}_1 \neq \mathbb{G}_2$  with an efficient isomorphism  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$  (possible with ordinary curves and the trace map).
- **Case III:** asymmetric  $\mathbb{G}_1 \neq \mathbb{G}_2$  but there is no known efficiently computable isomorphism  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ .

For more details, see “Pairings for Cryptographers”, by Galbraith-Paterson-Smart (<http://eprint.iacr.org/2006/165>)

## Consequences

Impact on the hardness of the Decision Diffie-Hellman problem (i.e. given  $g^a$  and  $g^b$ , distinguish  $g^{ab}$  from random):

- **Case I** ( $\mathbb{G}_1 = \mathbb{G}_2$ ): easy (Joux-Nguyen 2002). Given  $(g^a, g^b, h)$ ,

$$h = g^{ab} \iff e(g, h) = e(g^a, g^b).$$

- **Case II** ( $\mathbb{G}_1 \neq \mathbb{G}_2$  and a computable isomorphism  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$  is available):

- easy in  $\mathbb{G}_2$ : given  $(g_2^a, g_2^b, h)$ ,

$$h = g_2^{ab} \iff e(g_1, h) = e(\psi(g_2^a), g_2^b).$$

- hard in  $\mathbb{G}_1$  (eXternal Diffie-Hellman assumption).

- **Case III** ( $\mathbb{G}_1 \neq \mathbb{G}_2$  and no computable isomorphism  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$  is known): seemingly hard in both  $\mathbb{G}_1$  and  $\mathbb{G}_2$ .

## 2 Security Definition for Signature Schemes (Goldwasser-Micali-Rivest, 1988)

*Existential unforgeability against chosen-message attacks* (EUF-CMA) captured by the negligible advantage of  $\mathcal{A}$  in a game.

1. The challenger  $\mathcal{C}$  generates a key pair  $(SK, PK)$  and gives  $PK$  to the adversary  $\mathcal{A}$ .
2.  $\mathcal{A}$  gets  $\sigma_i = \text{Sign}(SK, M_i)$  for any messages  $M_i$  ( $i = 1, \dots, q$ ).
3.  $\mathcal{A}$  outputs  $(M, \sigma)$  and wins if  $\text{Verify}(PK, M, \sigma) = 1$  and  $M \neq M_i$ .

### 3 Pairing-Based Short Signatures (Boneh-Lynn-Shacham, Asiacrypt'01)

Uses groups  $\mathbb{G}_1 = \langle g_1 \rangle$ ,  $\mathbb{G}_2 = \langle g_2 \rangle$  and  $\mathbb{G}_T$  endowed with mappings  $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$  and  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$  so that  $g_2 = \psi(g_1)$ .

**Keygen:** generate public key  $PK = (X = g_2^x, H : \{0, 1\}^* \rightarrow \mathbb{G}_1)$   
and private key  $SK = x \xleftarrow{R} \mathbb{Z}_p^*$ .

**Sign:** to sign a message  $m$ , output  $\sigma = H(m)^x \in \mathbb{G}_1$ .

**Verify:** check

$$e(H(m), X) \stackrel{?}{=} e(\sigma, g_2).$$

Note that  $(g_1, X = g_2^x, H(m), \sigma = H(m)^x)$  is a co-CDH tuple when  $\sigma$  is a valid signature. Verification checks this relationship.

## Security of BLS signatures

**Definition 1** *In bilinear map groups  $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ , the **co-CDH** problem is, given  $(g_1, g_2, g_1^a, g_2^b) \in (\mathbb{G}_1 \times \mathbb{G}_2)^2$ , to compute  $g_1^{ab}$ .*

**Theorem 1** *The BLS scheme is EUF-CMA under the co-CDH assumption. A forger  $\mathcal{A}$  with advantage  $\varepsilon$  after  $q_s$  signing queries implies an algorithm solving co-CDH with probability  $O(\varepsilon/q_s)$ .*

The proof uses the random oracle methodology (Bellare-Rogaway, ACM-CCS'93).



## Security of BLS signatures (cont.)

**Idea of the proof:** use Coron's trick (Crypto'00) to answer random oracle queries and solve a co-CDH instance  $(g_1, g_2, g_1^a, g_2^b)$ .

Set  $X = g_2^b$  as a public key.

For each random oracle query  $H(M_i)$ :

- set  $H(M_i) = g_1^t$  with  $t \xleftarrow{R} \mathbb{Z}_p^*$  with probability  $\delta = q_s / (q_s + 1)$ .

$\Rightarrow$  Signatures are computable  $\sigma_i = \psi(g_2^b)^t$

- return  $H(M_i) = g_1^t \cdot (g_1^a)$  with  $t \xleftarrow{R} \mathbb{Z}_p^*$  with probability  $1 - \delta$ .

$\Rightarrow$  A valid signature  $\sigma^*$  leaks  $g_1^{ab} = \sigma^* / \psi(g_2^b)^t$

## Security of BLS signatures (cont.)

- The reduction features a “gap” of  $O(q_s)$  that can be avoided by slightly lengthening signatures (Coron, Eurocrypt’02).
- Only the proof requires the availability of an isomorphism  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ .
- The scheme remains secure without  $\psi$  (setting of case III) but under a less natural assumption

i.e. the hardness of finding  $g_1^{ab}$  given  $(g_1^a, g_1^b, g_2^b) \in (\mathbb{G}_1^2 \times \mathbb{G}_2)$ .

# Extensions of BLS Signatures

## Non-interactive multisignatures (Boldyreva, PKC'03)

- Consider  $n$  players  $\mathcal{P} = \{P_1, \dots, P_n\}$  having public keys  $X_1 = g_2^{x_1}, \dots, X_n = g_2^{x_n}$ .
- Members  $\{P_{i_1}, \dots, P_{i_t}\}$  of *any* subset of  $\mathcal{P}$  can independently issue shares  $\sigma_{i_j} = H(m)^{x_{i_j}} \in \mathbb{G}_1$  for  $j = 1, \dots, t$  ( $t \leq n$ ).
- Verification:

$$e(\sigma, g_2) \stackrel{?}{=} e(H(m), \prod_{j=1}^t X_{i_j}).$$

The security is reduced to that of standard BLS signatures in the *registered public key model*.

## Aggregate signatures

(Boneh-Gentry-Lynn-Shacham, Eurocrypt'03)

- Consider  $n$  BLS signatures  $\sigma_i = H(m_i)^{x_i} \in \mathbb{G}_1$  on  $n$  distinct messages  $m_i$  for parties with public keys  $X_i = g_2^{x_i} \in \mathbb{G}_2$ .
- Aggregation *by any party* to form a single signature  $\sigma = \prod_{i=1}^n \sigma_i \in \mathbb{G}_1$ .
- Verification via:

$$e(\sigma, g_2) \stackrel{?}{=} \prod_{i=1}^n e(H(m_i), X_i).$$

## Aggregate signatures (cont.)

- Security analysis is in the *chosen public key model* if the scheme is restricted to aggregate signatures on *distinct* messages.
- This constraint was lifted by Bellare-Namprempre-Neven (ICALP'07, <http://eprint.iacr.org/2006/285>).
- Lysyanskaya *et al.* (Eurocrypt'04) proposed *sequential* aggregate signatures from trapdoor permutations.

## Verifiably encrypted signatures (Boneh-Gentry-Lynn-Shacham, Eurocrypt'03)

- Let a signer with public key  $X = g_2^x \in \mathbb{G}_2$  and an adjudicator with public key  $Y = g_2^y \in \mathbb{G}_2$ .
- The signer generates an ElGamal-encrypted signature

$$(\sigma_1, \sigma_2) = (g_1^r, H(m)^x \cdot \psi(Y)^r)$$

for a random  $r \xleftarrow{R} \mathbb{Z}_p^*$ .

- Verification:  $e(\sigma_2, g_2) \stackrel{?}{=} e(H(m), X) \cdot e(\sigma_1, Y)$ .
- Adjudication: given  $y \in \mathbb{Z}_p^*$ , extract  $\sigma = H(m)^x = \sigma_2 / \sigma_1^y$ .

## Further Extensions of BLS Signatures

- Ring signatures (Boneh-Gentry-Lynn-Shacham, Eurocrypt'03)
- Blind signatures, threshold signatures (Boldyreva, PKC'03),
- Universal designated verifier signatures  
(Steinfeld-Bull-Pieprzyk-Wang, Asiacrypt'03),
- ....

## 4 Short signatures in the standard model (Boneh-Boyen, Eurocrypt'04)

Let groups  $\mathbb{G}_1 = \langle g_1 \rangle$ ,  $\mathbb{G}_2 = \langle g_2 \rangle$  and  $\mathbb{G}_T$  so that mappings  $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$  and  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$  with  $g_2 = \psi(g_1)$  are available.

**Keygen:** generate public key  $PK = (X = g_2^x, Y = g_2^y)$  and private key  $SK = (x, y) \xleftarrow{R} \mathbb{Z}_p^* \times \mathbb{Z}_p^*$ .

**Sign:** to sign  $m \in \mathbb{Z}_p^*$ , pick  $r \xleftarrow{R} \mathbb{Z}_p^*$  and output

$$(r, \sigma) = \left( r, g_1^{1/(m+x+yr)} \right) \in \mathbb{Z}_p^* \times \mathbb{G}_1.$$

**Verify:** check

$$e(\sigma, X \cdot Y^r \cdot g_2^m) \stackrel{?}{=} e(g_1, g_2).$$



## Security of BB signatures

**Definition 2** *The  $q$ -Strong Diffie-Hellman problem ( $q$ -SDH) is, given  $(g_1, g_2, g_2^a, g_2^{(a^2)}, \dots, g_2^{(a^q)}) \in \mathbb{G}_1 \times \mathbb{G}_2^{q+1}$ , to find a pair  $(c, g_1^{1/(c+a)}) \in \mathbb{Z}_p \times \mathbb{G}_1$ .*

N.B.: all solutions  $(c, A) \in \mathbb{Z}_p \times \mathbb{G}_1$  satisfy

$$e(A, (g_2^a) \cdot g_2^c) = e(g_1, g_2).$$

**Theorem 2** *The scheme is secure under the  $q$ -SDH assumption in the standard model. A forger  $\mathcal{A}$  with advantage  $\epsilon$  after  $q_s$  signing queries allows solving  $(q_s + 1)$ -SDH with advantage  $\epsilon' \approx \epsilon$ .*

## Security of BB signatures (cont.)

Security proof starts from simplified scheme in the random oracle model.

**Keygen:** generate  $PK = X = g_2^x$  and  $SK = x \xleftarrow{R} \mathbb{Z}_p^*$ .

**Sign:** to sign  $m \in \mathbb{Z}_p^*$ , output  $\sigma = g_1^{1/(m+x)} \in \mathbb{G}_1$ .

**Verify:** check

$$e(\sigma, X \cdot g_2^m) \stackrel{?}{=} e(g_1, g_2).$$

**Idea of the proof:** in a group  $\mathbb{G}$ , given  $(g, g^a, g^{(a^2)}, \dots, g^{(a^q)})$ , one can compute  $(g', g'^a) \in \mathbb{G} \times \mathbb{G}$  so that  $q$  pairs  $(c_i, g'^{1/(a+c_i)})$  are known and another one  $(c^*, g'^{1/(a+c^*)})$ , with  $c^* \neq c_i$ , yields  $(c^*, g^{1/(c^*+a)}) \in \mathbb{Z}_p \times \mathbb{G}_1$ .

## Extensions: short group signatures (Boneh-Boyen-Shacham, Crypto'04)

Assume a group manager with public key  $\text{gpk} := w = g_2^\gamma \in \mathbb{G}_2$ . Each member  $i$  gets a pair  $\text{gsk}[i] := (A_i = g_1^{\frac{1}{\gamma+x_i}}, x_i)$  for  $i = 1, \dots, n$ .

To sign a message  $M$ , member  $i$  encrypts  $A_i$  into

$$T_1 = u^a \quad T_2 = v^b \quad T_3 = A_i \cdot h^{a+b}$$

using the opener's public key  $\text{opk} := (u = h^{\xi_1}, v = h^{\xi_2}, h)$  and provide a NIZK proof that  $(T_1, T_2, T_3)$  encrypts of a valid certificate  $(A_i, x_i)$ .

## Short group signatures (cont.)

- Signatures of 1533 bits (192 bytes) are obtained.
- Security holds in the random oracle model in a relaxation of the Bellare-Micciancio-Warinschi (Eurocrypt'03) security model.
- Anonymity under the Decision Linear Assumption: given  $(u, v, h, u^a, v^b, T)$ , deciding whether  $T \stackrel{?}{=} h^{a+b}$  is hard.
- Full-traceability under the  $q$ -SDH assumption (where  $q$  stands for the group size).

## Group Signatures with Verifier-Local Revocation (Boneh-Shacham, ACM CCS'04)

Group manager has public key  $\text{gpk} := w = g_2^\gamma \in \mathbb{G}_2$ . Member  $i$  gets a pair  $\text{gsk}[i] := (A_i = g_1^{\frac{1}{\gamma+x_i}}, x_i)$  for  $i = 1, \dots, n$ .

To sign  $M$ , user  $i$  picks  $r \xleftarrow{R} \mathbb{Z}_p^*$ , sets  $(u, v) = H_0(M, r) \in \mathbb{G}_2 \times \mathbb{G}_2$  and

$$T_1 = \psi(u)^a \quad T_2 = A_i \cdot \psi(v)^a$$

and gives a proof that  $(T_1, T_2)$  encrypts of a valid certificate  $A_i$ .

Given a CRL =  $(A_1^*, A_2^*, \dots, A_n^*)$ , accept the signature as unrevoked if

$$e(T_1, v) \neq e(T_2/A_j^*, u)$$

for all  $j \in \text{CRL}$ .

## Efficiency Improvements (Libert-Vergnaud, 2007)

GM has public key  $\mathbf{gpk} := (w = g_2^\gamma, h, \tilde{g}) \in \mathbb{G}_2^2 \times \mathbb{G}_1$ . Member  $i$  gets a triple  $\mathbf{gsk}[i] := (A_i, x_i, y_i) \in \mathbb{G}_1 \times \mathbb{Z}_p^* \times \mathbb{Z}_p^*$  where  $A_i = (g_1 \cdot \tilde{g}^{y_i})^{\frac{1}{\gamma+x_i}}$ .

To sign  $M$ , user  $i$  picks  $\alpha, \delta \xleftarrow{R} \mathbb{Z}_p^*$  and sets

$$T_1 = A_i \cdot \tilde{g}^\alpha \quad T_2 = \psi(h)^\delta \quad T_3 = H_0(M, T_1, T_2)^{x_i} \cdot g_1^\delta$$

and gives a proof of well-formedness for  $(T_1, T_2, T_3)$  and  $A_i$ .

Given a CRL =  $(\mathbf{gsk}^*[1], \dots, \mathbf{gsk}^*[n])$ , accept the signature as unrevoked if

$$\frac{e(T_3, h)}{e(T_2, g_2)} \neq e(H_0(M, T_1, T_2), h)^{x_j^*}$$

for all  $j \in CRL$ .

## Efficiency improvements (cont.)

- We obtain signatures of 1363 bits.
- Boneh-Shacham was estimated to 1192 bits but is difficult to implement in asymmetric settings (yields 1808-bit signatures with symmetric pairings on supersingular curves over  $\mathbb{F}_{3^{163}}$ ).
- Constant number of pairings at verification:
  - 4 pairings +  $|CRL|$  exp. in  $\mathbb{G}_T$  instead of  $3+2|CRL|$  pairings.
- Anonymity and traceability under DLIN and SDH assumptions in both schemes.
- A variant gives backward unlinkability with the same signature size (but linear number of pairings at verification)

## Other applications of BB signatures and the SDH assumption

- Verifiable random functions (Dodis-Yampolskiy, PKC'05)
- Fast identity-based signature/encryption (Barreto-Libert-McCullagh-Quisquater, Asiacrypt'05)
- Blind and partially blind signatures in the standard model (Okamoto, TCC'06) using a variant of BB
- $k$ -times anonymous authentication (Nguyen-Safavi-Naini, ACNS'05; Nguyen, Vietcrypt'06)
- ...



## 5 Waters Signatures (Eurocrypt'05)

### With symmetric pairings

Consider groups  $\mathbb{G} = \langle g \rangle$  and  $\mathbb{G}_T$  with a mapping  $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ .

**Setup:** generate  $PK = (Z = e(g, g)^x, \bar{U})$  where  $x \xleftarrow{R} \mathbb{Z}_p^*$ ,  
 $\bar{U} = (u', u_1, \dots, u_n) \xleftarrow{R} \mathbb{G}^{n+1}$  and  $SK = g^x \in \mathbb{G}$ .

**Sign:** to sign  $\mathbf{m} = m_1 \dots m_n \in \{0, 1\}^n$ , select  $r \xleftarrow{R} \mathbb{Z}_p^*$  and output

$$\sigma = \left( g^x \cdot H_W(\mathbf{m})^r, g^r \right) \quad \text{where} \quad H_W(\mathbf{m}) = u' \cdot \prod_{i=1}^n u_i^{m_i}$$

**Verify:** given  $\sigma = (\sigma_1, \sigma_2)$ , check

$$e(\sigma_1, g) / e(\sigma_2, H_W(\mathbf{m})) \stackrel{?}{=} Z.$$

## With asymmetric pairings

Let groups  $\mathbb{G}_1 = \langle g_1 \rangle$ ,  $\mathbb{G}_2 = \langle g_2 \rangle$  and  $\mathbb{G}_T$  and mappings  $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$  and  $\psi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$  with  $g_2 = \psi(g_1)$ .

**Setup:** generate  $PK = (Z = (g_1, g_2)^x, \bar{U})$  where  $x \xleftarrow{R} \mathbb{Z}_p^*$ ,  
 $\bar{U} = (u', u_1, \dots, u_n) \xleftarrow{R} \mathbb{G}_2^{n+1}$  and  $SK = g_1^x \in \mathbb{G}_1$ .

**Sign:** to sign  $\mathbf{m} = m_1 \dots m_n \in \{0, 1\}^n$ , select  $r \xleftarrow{R} \mathbb{Z}_p^*$  and output

$$\sigma = \left( g_1^x \cdot \psi(H_W(\mathbf{m}))^r, g_1^r \right) \quad \text{where} \quad H_W(\mathbf{m}) = u' \cdot \prod_{i=1}^n u_i^{m_i}$$

**Verify:** given  $\sigma = (\sigma_1, \sigma_2)$ , check

$$e(\sigma_1, g_2) / e(\sigma_2, H_W(\mathbf{m})) \stackrel{?}{=} Z.$$

## Security of Waters signatures

**Theorem 3** *The scheme is secure under the CDH assumption in the standard model. A forger  $\mathcal{A}$  with advantage  $\epsilon$  after  $q_s$  signing queries implies an algorithm solving CDH with advantage  $O(\epsilon/q_s)$ .*

- With asymmetric pairings, security relies on the co-CDH assumption.
- The scheme is not strongly unforgeable since signatures can be randomized.

## Security of Waters signatures (cont.)

**Proof idea:** given a CDH instance  $(g^a, g^b)$ , set  $Z = e(g^a, g^b)$  and choose  $\bar{U} = (u', u_1, \dots, u_n)$  s.t. for any  $\mathbf{m} = m_1 \dots m_n \in \{0, 1\}^n$ ,

$$H_W(\mathbf{m}) = u' \cdot \prod_{i=1}^n u_i^{m_i} = (g^b)^{F(\mathbf{m})} \cdot g^{K(\mathbf{m})}$$

for functions  $F : \{0, 1\}^n \rightarrow \mathbb{Z}$  (with  $|F(\cdot)| \ll p$ ),  $K : \{0, 1\}^n \rightarrow \mathbb{Z}_p$ .

The private key is implicitly set as  $SK = g^{ab}$ .

- Signing queries  $\mathbf{m}_i$  can be answered whenever  $F(\mathbf{m}_i) \neq 0$
- At the forgery stage,  $F(\mathbf{m}^*) = 0$  with probability  $O(1/q_s)$ . Then,  $\mathcal{A}$ 's output

$$(\sigma_1^*, \sigma_2^*) = \left( g^{ab} \cdot (g^{K(\mathbf{m}^*)})^r, g^r \right)$$

allows retrieving  $g^{ab} = \sigma_1^* / \sigma_2^{*K(\mathbf{m}^*)}$ .

# Extensions of Waters signatures

## Multisignatures (Lu *et al.* – Eurocrypt'06)

All signers use groups  $(\mathbb{G}, \mathbb{G}_T)$  with a map  $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ .

**Keygen:** each signer generates  $PK_j = Z_j = e(g, g)^{x_j}$  where  
 $SK_j = g^{x_j} \in \mathbb{G}$ . They all share  $\bar{U} = (u', u_1, \dots, u_n) \xleftarrow{R} \mathbb{G}^{n+1}$ .

**M-Sign:** given  $\mathbf{m} = m_1 \dots m_n \in \{0, 1\}^n$ , each signer outputs

$$(\sigma_{j,1}, \sigma_{j,2}) = \left( g^{x_j} \cdot H_W(\mathbf{m})^{r_j}, g^{r_j} \right) \text{ where } H_W(\mathbf{m}) = u' \cdot \prod_{i=1}^n u_i^{m_i}$$

and signatures are combined into  $(\sigma_1 = \prod_{j=1}^t \sigma_{1,j}, \sigma_2 = \prod_{j=1}^t \sigma_{2,j})$ .

**M-Verify:** given  $\sigma = (\sigma_1, \sigma_2)$ , check

$$e(\sigma_1, g) / e(\sigma_2, H_W(\mathbf{m})) \stackrel{?}{=} \prod_{j=1}^t Z_j.$$

## Sequential aggregate signatures

**Keygen:** each signer generates  $SK_j = (g^{x_j}, y'_j, y_{j,1}, \dots, y_{j,n}) \in \mathbb{G}$  and

$$PK_j = \left( Z_j = e(g, g)^{x_j}, \bar{U}_j = \left( u'_j = g^{y'_j}, u_{j,1} = g^{y_{j,1}}, \dots, u_{j,n} = g^{y_{j,n}} \right) \right).$$

**A-Sign:** given  $\mathbf{m}_j = m_{j,1} \dots m_{j,n} \in \{0, 1\}^n$  and

$$(\sigma'_1, \sigma'_2) = \left( g^{x_1 + \dots + x_{j-1}} \dots \left( (u'_1 \prod_{i=1}^n u_{1,i}^{m_{1,i}}) \dots (u'_{j-1} \prod_{i=1}^n u_{j-1,i}^{m_{j-1,i}}) \right)^r, g^r \right)$$

which is an aggregate-so-far on  $\mathbf{m}_1, \dots, \mathbf{m}_{j-1}$ , signer  $j$  computes

$$(\tilde{\sigma}_1, \tilde{\sigma}_2) = \left( \sigma'_1 \cdot g^{x_j} \cdot \sigma'_2^{y'_j + \sum_{i=1}^n m_{j,i} y_j}, \sigma'_2 \right)$$

which is re-randomized

$$(\sigma_1, \sigma_2) = \left( \tilde{\sigma}_1 \cdot \left( (u'_1 \prod_{i=1}^n u_{1,i}^{m_{1,i}}) \dots (u'_{j-1} \prod_{i=1}^n u_{j-1,i}^{m_{j-1,i}}) \right)^{r'}, \tilde{\sigma}_2 \cdot g^{r'} \right).$$

**A-Verify:** accept  $(\sigma_1, \sigma_2)$  as an aggregate-so-far for  $m_1, \dots, m_j$  and  $PK_1, \dots, PK_j$  if

$$\frac{e(\sigma_1, g)}{e(\sigma_2, \prod_{t=1}^j u'_t \prod_{i=1}^n u_{t,i}^{m_{t,i}})} = \prod_{t=1}^j Z_j.$$

- Constant number of pairing at verification (unlike BGLS).
- Aggregation is sequential but the aggregation order does not matter at verification.
- Security in the registered public key model for both multi and sequential aggregate signatures.

## Forward-Secure Signatures with Untrusted Updates (Boyen-Shacham-Shen-Waters, ACM-CCS'06)

- Forward-secure signatures (FSS) prevent forgeries for past time periods after a break-in.
- Many security architectures additionally encrypt keys under a human-chosen password.
- Problem with FSS schemes where the update algorithm should access the key in clear.
- Boyen *et al.* used pairings to construct a FSS where updates can be made on password-blinded keys.



## FSS with Untrusted Updates (cont.)

- The public key is

$$PK = \left( Z = e(g, g)^x, V = e(g, g)^v, \bar{U} = (u', u_1, \dots, u_n), \bar{H} = (h', h_1, \dots, h_\ell) \right)$$

At period  $t = t_1 \dots t_\ell$ , the signer holds a blinded private key

$$SK_t = \left( g^{x+v} \cdot (h' \cdot h_1^{t_1} \dots h_\ell^{t_\ell})^r, g^r \right) \text{ and a } 2^{nd} \text{ factor } DK = g^{-v}.$$

- To sign  $M = m_1 \dots m_n$  using  $SK_t = (s_1, s_2)$  and  $DK = g^{-v}$ , return

$$\sigma = (\sigma_1, \sigma_2, \sigma_3) = \left( DK \cdot s_1 \cdot (u' \cdot \prod_{j=1}^n u_j^{m_j})^s, s_2, g^s \right) \text{ with } r, s \xleftarrow{R} \mathbb{Z}_p^*$$

- To check  $\sigma$  for period  $t = t_1 \dots t_\ell$ , the verifier tests whether

$$e(\sigma_1, g) \stackrel{?}{=} Z \cdot e\left(h' \cdot \prod_{i=1}^{\ell} h_i^{t_i}, \sigma_2\right) \cdot e\left(u' \cdot \prod_{j=1}^n u_j^{m_j}, \sigma_3\right)$$

## FSS with Untrusted Updates (end)

- The scheme provides
  - Forward security (as defined by Bellare-Miner at Crypto'99)
  - Update security (i.e. the blinded key alone is useless to attackers)in the *standard model* assuming that computing  $g^{(a^{\ell+1})}$  is hard given  $(g, g^a, g^{(a^2)}, \dots, g^{(a^\ell)})$ .
- Features constant-size signatures and at most log-squared complexity in other metrics
- But FSS with Untrusted Updates can be efficiently achieved without pairings (Libert-Quisquater-Yung, ACM-CCS'07):
  - Generically from any FSS
  - From 2-party multi-signatures and extending Malkin-Micciancio-Miner (Eurocrypt'02)

## Further extensions of Waters signatures

- Strongly unforgeable signature (Boneh-Shen-Waters, PKC'06)
- Ring signatures (Shacham-Waters, PKC'07)
- Blind signatures (Okamoto, TCC'06)
- Group signatures (Boyen-Waters, Eurocrypt'06 and PKC'07)
- Identity-based signatures (Paterson-Schuldt, ACISP'06)
- Universal designated verifier signatures  
(Laguillaumie-Libert-Quisquater, SCN'06)
- ....

## 6 Conclusions

- Pairings allow for short signatures
- Random oracles may now be avoided in many primitives:
  - Group and ring signatures
  - Multisignatures and sequential aggregate signatures
- But several desirable properties remain elusive for those primitives in the standard model:
  - Constant-size ring signatures
  - Practical and fully (i.e. CCA) anonymous group signatures under well-studied assumptions
  - (Non-sequential) aggregate signatures
  - Aggregate and multisignatures in the chosen public key model