Unlinkable and Strongly Accountable Sanitizable Signatures from Verifiable Ring Signatures

Xavier Bultel

Pascal Lafourcade

UNIVERSITÉ Clermont Auvergne & associés

LIMOS

CANS'17

Signature

signer

verifier

secrete key: *sk*

public key: *pk*

Signature

signer

verifier

secrete key: sk public key: pk

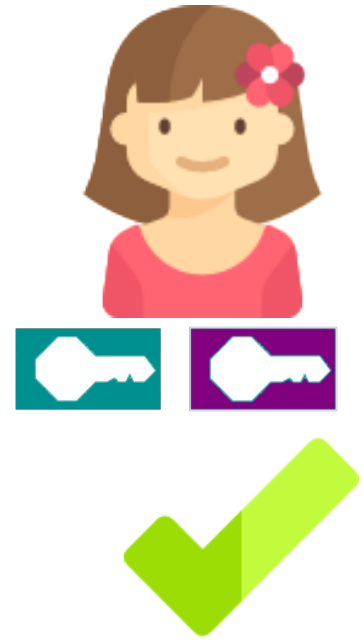
Schnorr's signature (1989)

- Key Generation: $pk = g^{sk}$
- Sign(m, sk): $\sigma = (g^r, z)$ where $z = r + sk \cdot H(g^r||m)$
- Verif(pk, m, σ): check $g^z = g^r \cdot pk^{H(g^r||m)}$

Sanitizable Signature, Atenise et al 2005

signer

sanitizer



Unforgeability

No unauthorized user can generate a valid signature

Immutability

signer

sanitizer

Sanitizer cannot modify unauthorized parts of a signature

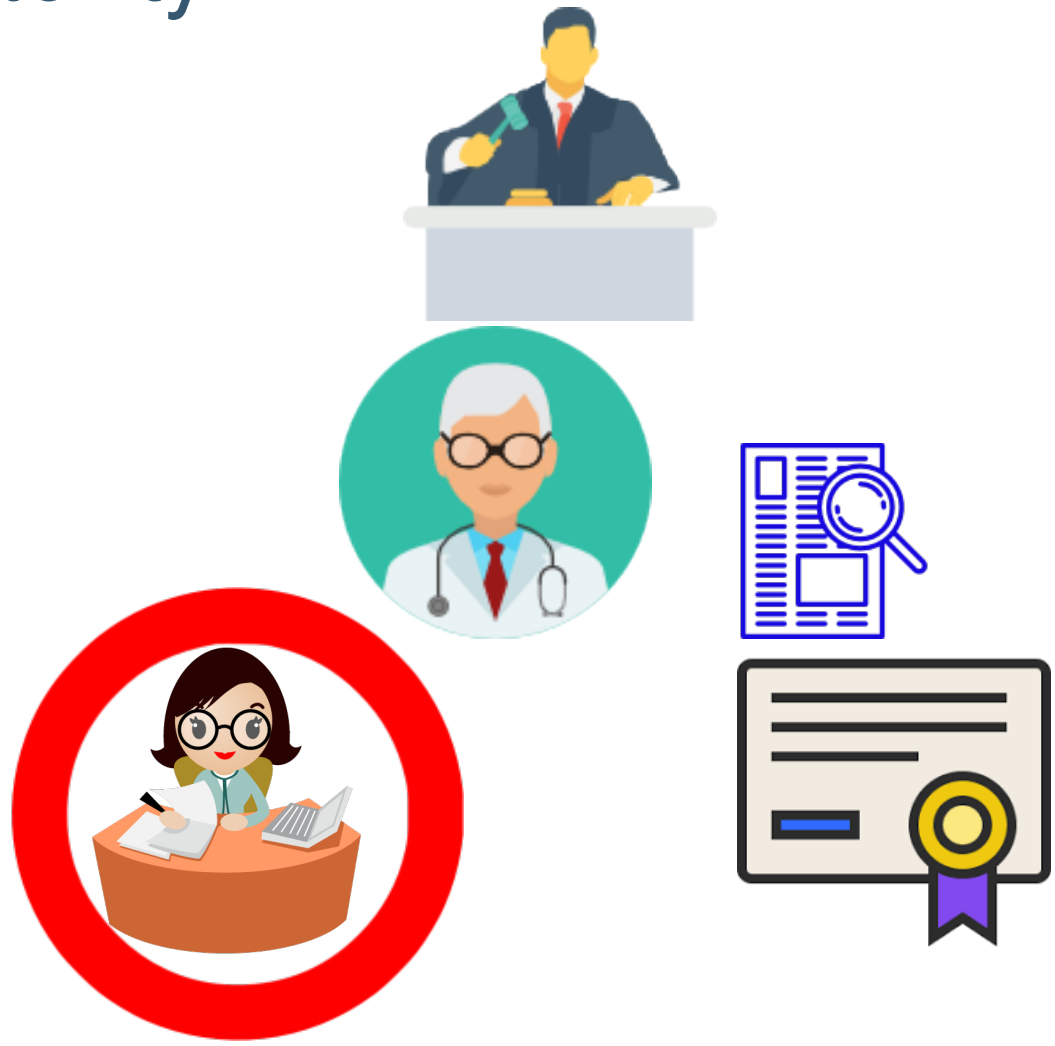
Privacy

All original sanitized information is unrecoverable

Transparency

Nobody can say if a signature is sanitized or not

Accountability



Signer can prove if a signature is sanitized or not

Unlinkability, Bruzuska et al 2010

Impossible to link a sanitized signature to the original one

Verifiable Ring Signature, Jiqiang et al 2003

Anonymous for the group but verifiable

Any signer can prove to the **judge** that he signed *m*

Contributions

New properties for VRS:

- Unforgeablity
- Anonymity
- Accountablitity
- Non-seizability

New property for USS:

- Strong Accountablitity

EVeR: Efficient Verifiable Ring signature

GUSS: Generic Unlinkable Sanitizable Signature
GUSS achieves strong accountability
EVeR + Schnorr = efficient GUSS

Comparison with Fleishhacker et al [PKC'16]

	SiGen	SaGen	Sig	San	Ver	SiProof	SiJudge	**Total**
PKC'16	7	1	15	14	17	23	6	**83**
GUSS	2	1	8	7	10	3	4	**35**

In number of exponentiations

	pk	spk	sk	ssk	σ	π	**Total**
PKC'16	7	1	14	1	14	4	**41**
GUSS	2	1	2	1	12	5	**23**

In number of group elements

Outline

Outline

Verifiable Ring Signature

Anonymous for the group but verifiable:
Any signer can prove to the **judge** that he signed m **or not**

Unforgeability

Only users of the group can generate a valid signature

Anonymity

Cannot distinguish who is the signer among the group

Accountability

User cannot sign *m* and prove that he did not sign it

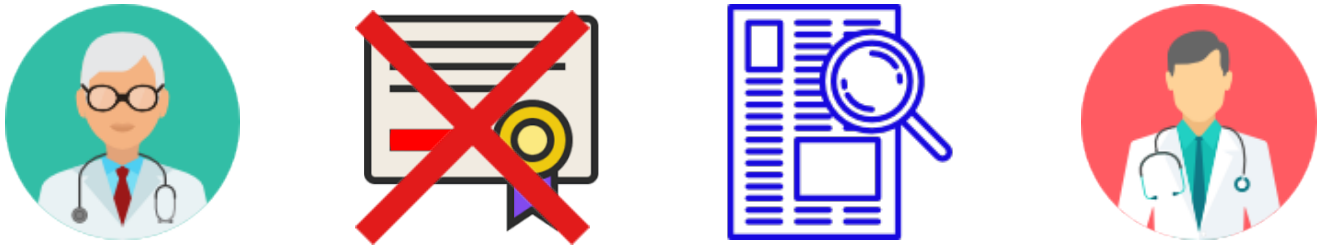
Non-seizability: (not pretend to be another)

- User cannot forge a signature with a proof that the signature has been produced by someone else

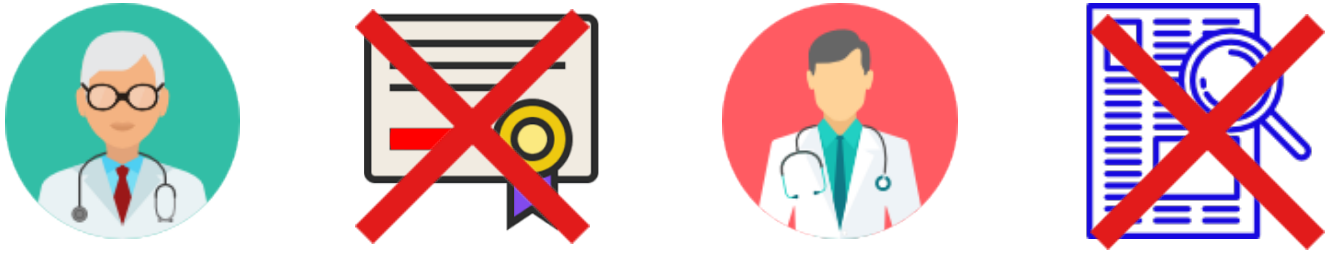



Non-seizability: (not pretend to be another)

- User cannot forge a signature with a proof that the signature has been produced by someone else



- User cannot forge σ s.t. others cannot prove that they do not forge it



Outline

Accountability like in PKC'16

Signer can prove the origin of a signature



Accountability like in PKC'16

Signer can prove the origin of a signature



1. Signer cannot forge a **false** signature for a sanitizer




Accountability like in PKC'16

Signer can prove the origin of a signature



1. Signer cannot forge a **false** signature for a sanitizer



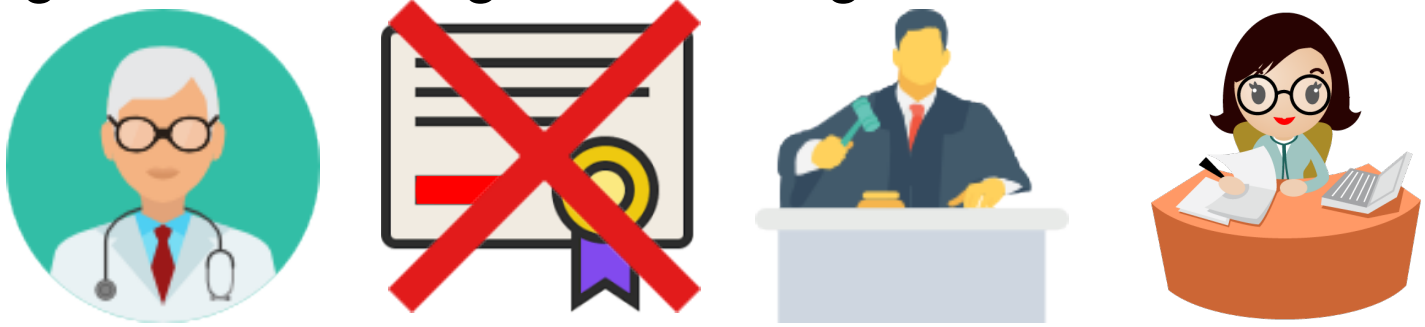

2. Sanitizer cannot forge a **false** signature for a signer




Strong Accountability

Signer or **sanitizer** can prove the origin of a signature

1. Signer cannot forge a **false** signature for a sanitizer



2. Sanitizer cannot forge a **false** signature for a signer

Outline

Efficient Verifiable Ring Signature: EVeR

Idea:

Signature

- anonymous commitment c of sk and m
- ZKP of the link between sk, c and one of several pks

Proof: ZKP that c was produced with sk associatd to pk

NIZKP Logarithm Equality (LogEq)

Prove $log_{g_j}(y_j) = log_{h_j}(z_j)$
among several y_i and z_i, $1 \leq i \leq n$

NIZKP Logarithm Equality (LogEq)

Prove $log_{g_j}(y_j) = log_{h_j}(z_j)$
among several y_i and z_i, $1 \leq i \leq n$

- $\pi = LEprove_n(\{h_i, z_i, g_i, y_i\}_{1 \leq i \leq n}, x)$
- $LEverif_n(\{(h_i, z_i, g_i, y_i)\}_{1 \leq i \leq n}, \pi)$:
  return true iff $\exists j, x = log_{g_j}(y_j) = log_{h_j}(z_j)$

EVeR

V.Gen($\mathbb{G}, p, g, H$): $\mathsf{pk} = g^{\mathsf{sk}}$ and return$(\mathsf{pk}, \mathsf{sk})$

V.Sig(L, m, sk): $h = H(m||r)$ and $z = h^{\mathsf{sk}}$
$P \leftarrow \mathsf{LEprove}_{|L|}(\{(h, z, g, \mathsf{pk}_l)\}_{\mathsf{pk}_l \in L}, \mathsf{sk})$ return (r, z, P)

V.Ver(L, m, σ): $h = H(m||r)$ and return
$b \leftarrow \mathsf{LEverif}_{|L|}(\{(h, z, g, \mathsf{pk}_l)\}_{\mathsf{pk}_l \in L}, P)$

V.Proof($L, m, \sigma, \mathsf{pk}, \mathsf{sk}$): $h = H(m||r)$ and
$\bar{P} \leftarrow \mathsf{LEprove}_1(\{(h, h^{\mathsf{sk}}, g, \mathsf{pk})\}, \mathsf{sk})$
return $\pi = (h^{\mathsf{sk}}, \bar{P})$

V.Judge($L, m, \sigma, \mathsf{pk}, \pi$): $h = H(m||r)$
$b \leftarrow \mathsf{LEverif}_1(\{(h, h^{\mathsf{sk}}, g, \mathsf{pk})\}, \pi)$
If $b = 1$ and $z = h^{\mathsf{sk}}$ then return 1 else 0

Result

Theorem

EVeR *is* **unforgeable, anonymous, accountable** *and* **non-seizable** *under the DDH assumption in the ROM.*

Outline

Generic Unlinkable Sanitizable Signature

Signature

Generic Unlinkable Sanitizable Signature


Signature




Sanitization



Verification and Judge for signer and sanitizer are straightforward

Result

Theorem

For any deterministic and unforgeable DS scheme D and any unforgeable, anonymous, accountable and non-seizable VRS scheme V, GUSS *instantiated with* (D, V) *is* **immutable, transparent, strongly accountable** *and* **unlinkable**.

Outline

Results

New properties for VRS:

- Unforgeablity
- Anonymity
- Accountablitity
- Non-seizability

New property for USS:

- Strong Accountablitity

EVeR: Efficient Verifiable Ring signature

GUSS: Generic Unlinkable Sanitizable Signature
GUSS achieves strong accountability
EVeR + Schnorr = efficient GUSS

Thank you for your attention!

Questions

Images designed by Freepik from www.flaticon.com

NIZKP LogEq

Prove $x = log_{g_j}(y_j) = log_{h_j}(z_j)$ among several y_i and z_i in L

$LEprove_n(\{h_i, z_i, g_i, y_i\}_{1 \leq i \leq n}, x)$:

$R_j = g_j^{r_j}$ and $S_j = h_j^{r_j}$

$\forall i \neq j : R_i = \frac{g_i^{\gamma_i}}{y_i^{c_i}}$ and $S_i = \frac{h_i^{\gamma_i}}{z_i^{c_i}}$

$$c_j = \frac{H(R_1||S_1||\ldots||R_n||S_n)}{\prod_{i=1;i\neq j}^{n} c_i} \qquad \gamma_j = r_j + c_j \cdot x$$

Return $\pi = \{(R_i, S_i, c_i, \gamma_i)\}_{1 \leq i \leq n}$

$LEverif_n(\{(h_i, z_i, g_i, y_i)\}_{1 \leq i \leq n}, \pi)$:

Return $H(R_1||\ S_1||\ldots||R_n||S_n) = \prod_{i=1}^{n} c_i$ and

$\forall i \in [1, n]$ such that $g_i^{\gamma_i} = R_i \cdot y_i^{c_i} (= g_j^{\gamma_j = r_j + c_j \cdot x})$

and $h_i^{\gamma_i} = S_i \cdot z_i^{c_i} = (= h_j^{\gamma_j = r_j + c_j \cdot x})$

Generic Unlinkable Sanitizable Signature

$D = (\mathsf{D.Init}, \mathsf{D.Gen}, \mathsf{D.Sig}, \mathsf{D.Ver})$ a deterministic signature
$V = (\mathsf{V.Init}, \mathsf{V.Gen}, \mathsf{V.Sig}, \mathsf{V.Ver}, \mathsf{V.Proof}, \mathsf{V.Judge})$ a VRS

$\mathsf{Sig}(m, \mathsf{sk}, \mathsf{spk}, \textsc{Adm})$: $\mathsf{sk} = (\mathsf{sk}_d, \mathsf{sk}_v)$, $M \leftarrow \textsc{Fix}_{\textsc{Adm}}(m)$
$\sigma_1 \leftarrow \mathsf{D.Sig}(\mathsf{sk}_d, (M||\textsc{Adm}||\mathsf{pk}||\mathsf{spk}))$ and
$\sigma_2 \leftarrow \mathsf{V.Sig}(\{\mathsf{pk}_v, \mathsf{spk}\}, \mathsf{sk}_v, (\sigma_1||m)))$
Return $\sigma = (\sigma_1, \sigma_2, \textsc{Adm})$

$\mathsf{San}(m, \textsc{Mod}, \sigma, \mathsf{pk}, \mathsf{ssk})$: $\mathsf{pk} = (\mathsf{pk}_d, \mathsf{pk}_v)$
$\sigma_2' \leftarrow \mathsf{V.Sig}(\{\mathsf{pk}_v, \mathsf{spk}\}, \mathsf{ssk}, (\sigma_1||\textsc{Mod}(m)))$
Return $\sigma' = (\sigma_1, \sigma_2', \textsc{Adm})$

$\mathsf{Ver}(m, \sigma, \mathsf{pk}, \mathsf{spk})$: $\sigma = (\sigma_1, \sigma_2, \textsc{Adm})$, $M \leftarrow \textsc{Fix}_{\textsc{Adm}}(m)$.
$b_1 \leftarrow \mathsf{D.Ver}(\mathsf{pk}_d, (M||\textsc{Adm}||\mathsf{pk}||\mathsf{spk}), \sigma_1)$
$b_2 \leftarrow \mathsf{V.Ver}(\{\mathsf{pk}_d, \mathsf{spk}\}, (\sigma_1||m), \sigma_2)$
Return $b = (b_1 \wedge b_2)$

Generic Unlinkable Sanitizable Signature

SiProof($\mathsf{sk}, m, \sigma, \mathsf{spk}$): $\sigma = (\sigma_1, \sigma_2, \mathrm{ADM})$, $\mathsf{sk} = (\mathsf{sk}_d, \mathsf{sk}_v)$

$\pi_{\mathsf{si}} \leftarrow \mathsf{V.Proof}(\{\mathsf{pk}_v, \mathsf{spk}\}, (m||\sigma_1), \sigma_2, \mathsf{pk}_v, \mathsf{sk}_v)$

Return π_{si}

SaProof($\mathsf{ssk}, m, \sigma, \mathsf{pk}$): $\sigma = (\sigma_1, \sigma_2, \mathrm{ADM})$

$\pi_{\mathsf{sa}} \leftarrow \mathsf{V.Proof}(\{\mathsf{pk}_v, \mathsf{spk}\}, (m||\sigma_1), \sigma_2, \mathsf{spk}, \mathsf{ssk})$

Return π_{sa}

SiJudge($m, \sigma, \mathsf{pk}, \mathsf{spk}, \pi_{\mathsf{si}}$): $\sigma = (\sigma_1, \sigma_2, \mathrm{ADM})$,

$\mathsf{pk} = (\mathsf{pk}_d, \mathsf{pk}_v)$

$b \leftarrow \mathsf{V.Judge}(\{\mathsf{pk}_v, \mathsf{spk}\}, (m||\sigma_1), \sigma_2, \mathsf{pk}_v, \pi_{\mathsf{si}})$

Return b

SaJudge($m, \sigma, \mathsf{pk}, \mathsf{spk}, \pi_{\mathsf{sa}}$): $\sigma = (\sigma_1, \sigma_2, \mathrm{ADM})$,

$\mathsf{pk} = (\mathsf{pk}_d, \mathsf{pk}_v)$

$b \leftarrow \mathsf{V.Judge}(\{\mathsf{pk}_v, \mathsf{spk}\}, (m||\sigma_1), \sigma_2, \mathsf{spk}, \pi_{\mathsf{sa}})$

Return $(1 - b)$