Universal Ring Signatures in the Standard Model

Pedro Branco[1] Nico Döttling[2,*] Stella Wohnig[2,3]

[1]Johns Hopkins University, partially done at IST University of Lisbon

[2]Helmholtz Center for Information Security (CISPA)

[3]Universität des Saarlandes

[*]Funded by an ERC grant

Asiacrypt 2022

Ring signatures

Semantics

- $\mathsf{Gen}(1^\lambda) \to (\mathsf{vk}, \mathsf{sk})$
- $\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, \underbrace{(\mathsf{vk}_1, \ldots \mathsf{vk}_\ell)}_{\text{Ring R}}) \to \Sigma$
- $\mathsf{Verify}(\Sigma, m, R) \to \{0, 1\}$

Ring signatures

Semantics

- $\mathsf{Gen}(1^\lambda) \rightarrow (\mathsf{vk}, \mathsf{sk})$
- $\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, \underbrace{(\mathsf{vk}_1, \ldots \mathsf{vk}_\ell)}_{\text{Ring R}}) \rightarrow \Sigma$
- $\mathsf{Verify}(\Sigma, m, R) \rightarrow \{0, 1\}$

Anonymity

Does not reveal who exactly created a signature ...

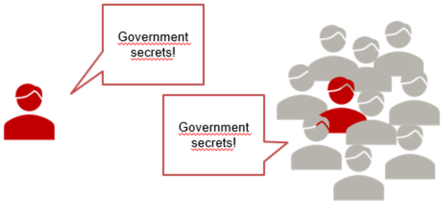
Unforgeability

... only that it was someone from the ring

Ring signatures

Semantics

- $\mathsf{Gen}(1^\lambda) \to (\mathsf{vk}, \mathsf{sk})$
- $\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, \underbrace{(\mathsf{vk}_1, \ldots \mathsf{vk}_\ell)}_{\text{Ring R}}) \to \Sigma$
- $\mathsf{Verify}(\Sigma, m, R) \to \{0, 1\}$



Anonymity

Does not reveal who exactly created a signature ...

Unforgeability

... only that it was someone from the ring

Governments don't like whistleblowers, duh

Governments don't like whistleblowers, duh

The situation:

Employee

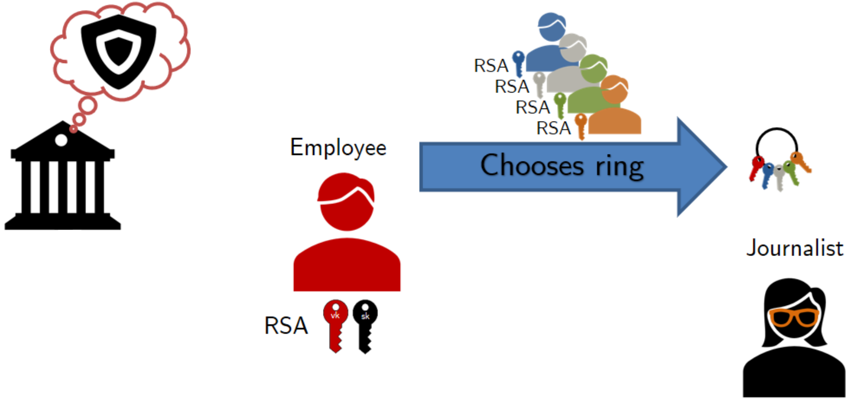
Journalist

Governments don't like whistleblowers, duh

The situation:

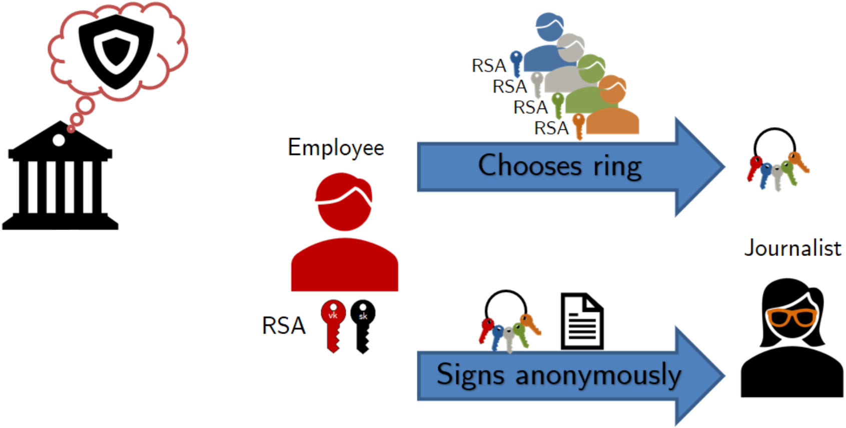
Governments don't like whistleblowers, duh

The situation:



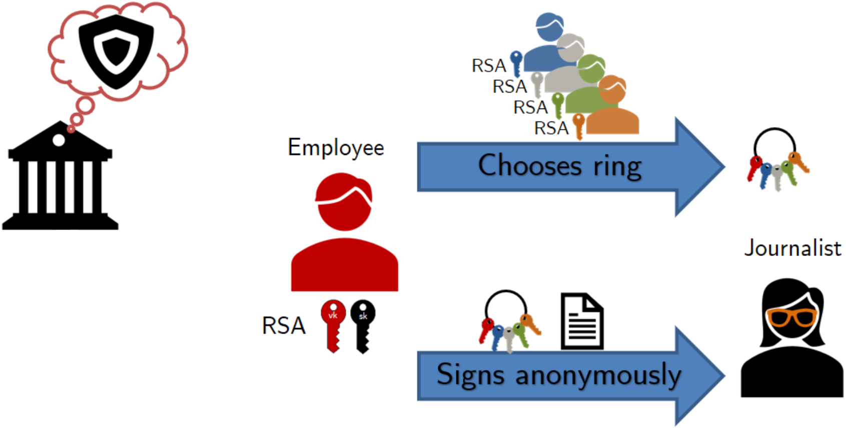
Governments don't like whistleblowers, duh

The situation:



Governments don't like whistleblowers, duh

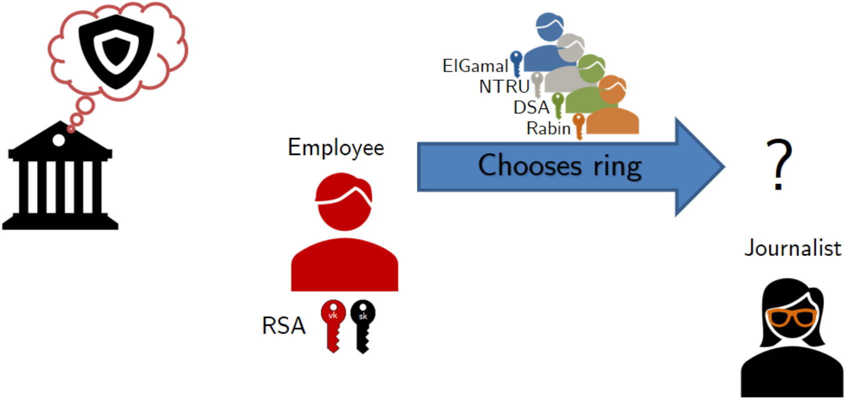
The situation:



Things are never so easy...

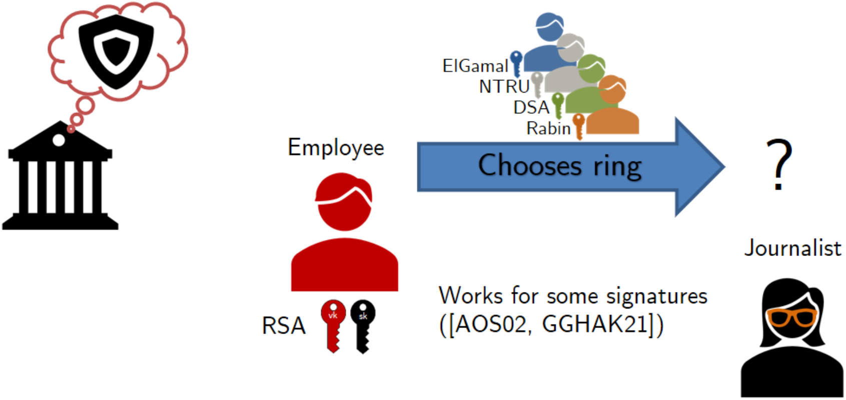
Governments don't like whistleblowers, duh

The situation:

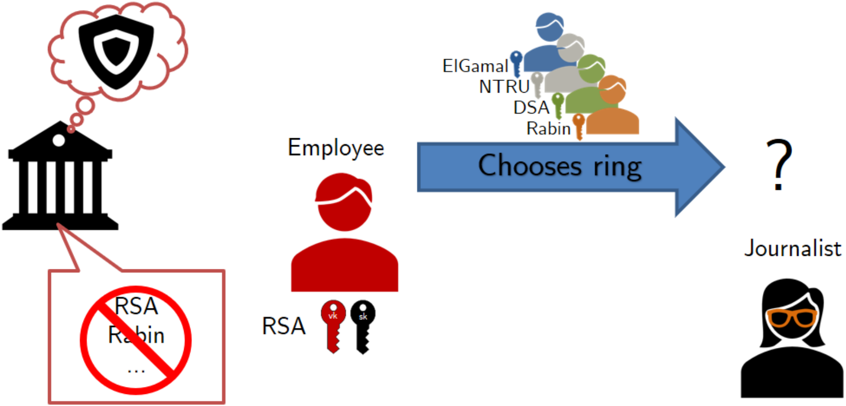


Governments don't like whistleblowers, duh

The situation:

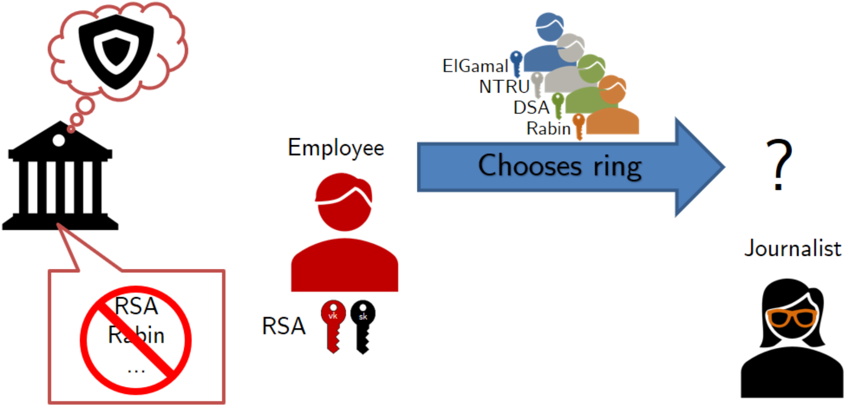


The situation:



Governments don't like whistleblowers, duh

The situation:

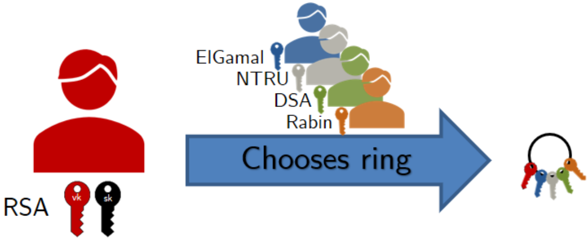


Is there protection from being used in anonymity ring?

NO

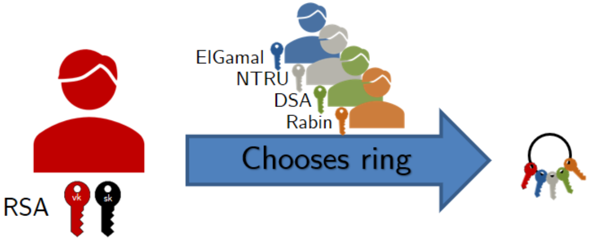
NO*

*terms and conditions apply



Ring signatures for rings of keys from ANY set of signing schemes.

Universal Ring Signatures



Ring signatures for rings of keys from ANY set of signing schemes.

Semantics

- $\mathsf{Gen}(1^\lambda) \to (\mathsf{vk}, \mathsf{sk})$ \\ same as in underlying signature
- $\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots \mathsf{vk}_\ell), \underbrace{(\mathsf{Sig}_1, \ldots, \mathsf{Sig}_M)}_{\text{List of schemes } S}) \to \Sigma$
- $\mathsf{Verify}(\Sigma, m, R, S) \to \{0, 1\}$

The (too) obvious solution...

We could use a CRS or the ROM. NIZKPoK construction exists.

The (too) obvious solution...

We could use a CRS or the ROM. NIZKPoK construction exists.

No! Why?

The (too) obvious solution...

We could use a CRS or the ROM. NIZKPoK construction exists.

No! Why?

- Who chooses CRS? No trusted setting.

The (too) obvious solution...

We could use a CRS or the ROM. NIZKPoK construction exists.

No! Why?

- Who chooses CRS? No trusted setting.
- RO must be instantiated. Soundness issues.

The (too) obvious solution...

We could use a CRS or the ROM. NIZKPoK construction exists.

No! Why?

- Who chooses CRS? No trusted setting.
- RO must be instantiated. Soundness issues.

Standard model it is - why is this hard?

The challenge

For any secure schemes $\mathsf{Ls} = \{\mathsf{Sig}_i\}_i$ we need:

Unforgeability

Experiment Adversary $\mathcal{A}$

$$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$$

$$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow \mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$$

Repeatedly:

$$\xleftarrow{\text{may ask to corrupt honest keys}}$$

$$\xleftarrow{\text{or ask for (universal ring) signatures}}$$

When done: $\xleftarrow{(\Sigma^*, m^*, R^*, S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

The challenge

For any secure schemes $\mathsf{Ls} = \{\mathsf{Sig}_i\}_i$ we need:

Unforgeability

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$

Repeatedly:

$\xleftarrow{\text{may ask to corrupt honest keys}}$

$\xleftarrow{\text{or ask for (universal ring) signatures}}$

When done: $\xleftarrow{(\Sigma^*,m^*,R^*,S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

Must reduce to unforgeability of Sig.

The challenge

For any secure schemes $\mathsf{Ls} = \{\mathsf{Sig}_i\}_i$ we need:

Unforgeability

Experiment — Adversary $\mathcal{A}$

$$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$$

$$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$$

Repeatedly:

$$\xleftarrow{\text{may ask to corrupt honest keys}}$$

$$\xleftarrow{\text{or ask for (universal ring) signatures}}$$

When done: $\xleftarrow{(\Sigma^*,m^*,R^*,S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

Must reduce to unforgeability of Sig.

$\Rightarrow$ Reduction must extract a signature from Σ^*, that is forge for some Sig_i.

The challenge

For any secure schemes $\mathsf{Ls} = \{\mathsf{Sig}_i\}_i$ we need:

Unforgeability

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$

Repeatedly:

$\xleftarrow{\text{may ask to corrupt honest keys}}$

$\xleftarrow{\text{or ask for (universal ring) signatures}}$

When done: $\xleftarrow{(\Sigma^*,m^*,R^*,S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

Must reduce to unforgeability of Sig.

$\Rightarrow$ Reduction must extract a signature from Σ^*, that is forge for some Sig_i.

Anonymity

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$

$\xrightarrow{\text{Additionally outputs randomness}(r_i)_i}$

$\xleftarrow{(m^*,R^*,S^*,(ind_{\mathbf{0}},ind_{\mathbf{1}})),\mathsf{vk}^*_{ind_{\mathbf{0}/\mathbf{1}}}\in R}$

$\xrightarrow{\Sigma^*=\mathsf{URS.Sign}(1^\lambda,\mathsf{sk}^*_{ind_b},m^*,R^*,S^*)\text{ for } b\leftarrow_\$\{0,1\}}$

$\xleftarrow{\text{Guess } b'.}$

Wins if $b = b'$.

The challenge

For any secure schemes $\mathsf{Ls} = \{\mathsf{Sig}_i\}_i$ we need:

Unforgeability

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$

Repeatedly:

$\xleftarrow{\text{may ask to corrupt honest keys}}$

$\xleftarrow{\text{or ask for (universal ring) signatures}}$

When done: $\xleftarrow{(\Sigma^*,m^*,R^*,S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

Must reduce to unforgeability of Sig.
$\Rightarrow$ Reduction must extract a signature from Σ^*, that is forge for some Sig_i.

Anonymity

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$

$\xrightarrow{\text{Additionally outputs randomness}(r_i)_i}$

$\xleftarrow{(m^*,R^*,S^*,(ind_{\mathbf{0}},ind_{\mathbf{1}})),\mathsf{vk}^*_{ind_{\mathbf{0}/\mathbf{1}}}\in R}$

$\xrightarrow{\Sigma^*=\mathsf{URS.Sign}(1^\lambda,\mathsf{sk}^*_{ind_b},m^*,R^*,S^*)\text{ for } b\leftarrow_\$\{0,1\}}$

$\xleftarrow{\text{Guess } b'.}$

Wins if $b = b'$.

$\mathcal{A}$ may NOT extract a forge for any Sig_i from Σ^*.

The challenge

For any secure schemes $\mathsf{Ls} = \{\mathsf{Sig}_i\}_i$ we need:

Unforgeability

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow \mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$

Repeatedly:

$\xleftarrow{\text{may ask to corrupt honest keys}}$

$\xleftarrow{\text{or ask for (universal ring) signatures}}$

When done: $\xleftarrow{(\Sigma^*,m^*,R^*,S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

Anonymity

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow \mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$

$\xrightarrow{\text{Additionally outputs randomness}(r_i)_i}$

$\xleftarrow{(m^*,R^*,S^*,(ind_{\mathbf{0}},ind_{\mathbf{1}})),\mathsf{vk}^*_{ind_{\mathbf{0}/\mathbf{1}}}\in R}$

$\xrightarrow{\Sigma^*=\mathsf{URS.Sign}(1^\lambda,\mathsf{sk}^*_{ind_b},m^*,R^*,S^*) \text{ for } b\leftarrow_\$\{0,1\}}$

$\xleftarrow{\text{Guess } b'.}$

Wins if $b = b'$.

How can the reduction get an edge over the adversary?

The challenge

For any secure schemes $\mathsf{Ls} = \{\mathsf{Sig}_i\}_i$ we need:

Unforgeability

Experiment | Adversary $\mathcal{A}$

$$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$$

$$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$$

Repeatedly:

$$\xleftarrow{\text{may ask to corrupt honest keys}}$$

$$\xleftarrow{\text{or ask for (universal ring) signatures}}$$

When done: $\xleftarrow{(\Sigma^*,m^*,R^*,S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

Anonymity

Experiment | Adversary $\mathcal{A}$

$$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$$

$$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$$

$$\xrightarrow{\text{Additionally outputs randomness}(r_i)_i}$$

$$\xleftarrow{(m^*,R^*,S^*,(ind_{\mathbf{0}},ind_{\mathbf{1}})),\mathsf{vk}^*_{ind_{\mathbf{0}/\mathbf{1}}}\in R}$$

$$\xrightarrow{\Sigma^*=\mathsf{URS.Sign}(1^\lambda,\mathsf{sk}^*_{ind_b},m^*,R^*,S^*)\text{ for }b\leftarrow_\$\{0,1\}}$$

$$\xleftarrow{\text{Guess }b'.}$$

Wins if $b = b'$.

How can the reduction get an edge over the adversary WITHOUT changing the key structure?

Overview

Need to hide a signature, where only the reduction can find it...

Overview

Need to hide a signature, where only the reduction can find it...

- Allow the reduction more run-time: Complexity leveraging

Overview

Need to hide a signature, where only the reduction can find it...

- Allow the reduction more run-time: Complexity leveraging
- Allow the reduction more information: Witness encryption

The terms and conditions

Construction	Size in R	Model	Assumptions/Caveats
[AOS02]	Linear	ROM	structured signatures[1]
[GGHAK21]	Log	ROM, CRS	structured signatures[2]
[BDW22] 1	Log	Standard	signatures are superpoly secure
[BDW22] 2	Linear	Standard	Witness Encryption, t-anonymity
[BDW22] 2b	Log	Standard	iO, t-anonymity

[1] hash-then-trapdoor-sign or three-move

[2] Σ-protocols

The terms and conditions

Construction	Size in R	Model	Assumptions/Caveats
[AOS02]	Linear	ROM	structured signatures[1]
[GGHAK21]	Log	ROM, CRS	structured signatures[2]
[BDW22] 1	Log	Standard	signatures are superpoly secure
[BDW22] 2	Linear	Standard	Witness Encryption, t-anonymity
[BDW22] 2b	Log	Standard	iO, t-anonymity

t-anonymity: Anonymity holds if $\geq t$ honest keys in R; commonly $(t = 3, 4)$

[1] hash-then-trapdoor-sign or three-move

[2] Σ-protocols

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.

The solution

Complexity Leveraging

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$.

The solution

Complexity Leveraging

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$.
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$

The solution
Complexity Leveraging

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$. Secure for superpoly T Adversary
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ Keyless, broken in T

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$. Secure for superpoly T Adversary
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ Keyless, broken in T
- Compute a NIWI proof $\pi \leftarrow \mathsf{NIWI.Prove}(\cdot, \cdot)$ that the above was done correctly for one $\mathsf{vk}_i \in R$ and $\mathsf{Sig}_{\mathsf{ind}} \in S$.
- Output $\Sigma = (\mathsf{com}_0, \pi)$.

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$. Secure for superpoly T Adversary
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ Keyless, broken in T
- Compute a NIWI proof $\pi \leftarrow \mathsf{NIWI.Prove}(\cdot, \cdot)$ that the above was done correctly for one $\mathsf{vk}_i \in R$ and $\mathsf{Sig}_{\mathsf{ind}} \in S$.
- Output $\Sigma = (\mathsf{com}_0, \pi)$.

Unforgeable ✓

The solution

Complexity Leveraging

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$.
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ and $(\mathsf{com}_1, \gamma_1) \leftarrow \mathsf{CS.Commit}(1^\lambda, 0)$.
- Compute a NIWI proof $\pi \leftarrow \mathsf{NIWI.Prove}(\cdot, \cdot)$ that (com_0, σ) OR (com_1, σ) are correct for one $\mathsf{vk}_i \in R$ and $\mathsf{Sig}_{\mathsf{ind}} \in S$.
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeable ✓

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$.
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ and $(\mathsf{com}_1, \gamma_1) \leftarrow \mathsf{CS.Commit}(1^\lambda, 0)$.
- Compute a NIWI proof $\pi \leftarrow \mathsf{NIWI.Prove}(\cdot, \cdot)$ that (com_0, σ) OR (com_1, σ) are correct for one $\mathsf{vk}_i \in R$ and $\mathsf{Sig}_{\mathsf{ind}} \in S$.
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeable ✓ 2-Anonymous ✓

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$.
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ and $(\mathsf{com}_1, \gamma_1) \leftarrow \mathsf{CS.Commit}(1^\lambda, 0)$.
- Compute a NIWI proof $\pi \leftarrow \mathsf{NIWI.Prove}(\cdot, \cdot)$ that (com_0, σ) OR (com_1, σ) are correct for one $\mathsf{vk}_i \in R$ and $\mathsf{Sig}_{\mathsf{ind}} \in S$.
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeable ✓ 2-Anonymous ✓
Size?

The solution

Complexity Leveraging

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig.Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$.
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ and $(\mathsf{com}_1, \gamma_1) \leftarrow \mathsf{CS.Commit}(1^\lambda, 0)$.
- Compute a NIWI proof $\pi \leftarrow \mathsf{NIWI.Prove}(\cdot, \cdot)$ that (com_0, σ) OR (com_1, σ) are correct for one $\mathsf{vk}_i \in R$ and $\mathsf{Sig}_{\mathsf{ind}} \in S$.
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeable ✓ 2-Anonymous ✓

Size? Standard trick: SPB digest ([BDH+19])

$\mathsf{Sign}(1^\lambda, \mathsf{sk}_i, m, R = (\mathsf{vk}_1, \ldots, \mathsf{vk}_\ell), S = \{\mathsf{Sig}_i\}_{i \in [M]})$

- Let vk_i, $\mathsf{Sig}_{\mathsf{ind}}$ be corresponding to sk_i.
- $\sigma \leftarrow \mathsf{Sig}.\mathsf{Sign}_{\mathsf{ind}}(\mathsf{sk}_i, m)$.
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS}.\mathsf{Commit}(1^\lambda, \sigma)$ and $(\mathsf{com}_1, \gamma_1) \leftarrow \mathsf{CS}.\mathsf{Commit}(1^\lambda, 0)$.
- Compute a NIWI proof $\pi \leftarrow \mathsf{NIWI}.\mathsf{Prove}(\cdot, \cdot)$ that (com_0, σ) OR (com_1, σ) are correct for one $\mathsf{vk}_i \in R$ and $\mathsf{Sig}_{\mathsf{ind}} \in S$.
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeable ✓ 2-Anonymous ✓

Size? Standard trick: SPB digest ([BDH+19])

Superpoly-Assumption!

Solution 2 - rough idea

Want: Commitment $\xrightarrow{\textit{replace}}$ WE

Solution 2 - rough idea

Want: Commitment $\xrightarrow{\textit{replace}}$ WE

Witness Encryption

Let $\mathcal{L}$ be an NP language defined by $x \in \mathcal{L} \Leftrightarrow \exists w : (x, w) \in \mathcal{R}$. A witness encryption WE has the following algorithms:

- $\mathsf{ct} \leftarrow \mathsf{WE.Enc}(1^\lambda, x, m)$
- $m \leftarrow \mathsf{WE.Dec}(w, \mathsf{ct})$

It is correct & soundness secure.

The challenge - Recall

Unforgeability

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i, (\mathsf{vk}_i, \mathsf{sk}_i)\leftarrow \mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda; r_i)}$

Repeatedly:

$\xleftarrow{\text{may ask to corrupt honest keys}}$

$\xleftarrow{\text{or ask for (universal ring) signatures}}$

When done: $\xleftarrow{(\Sigma^*, m^*, R^*, S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

t-Anonymity

Experiment | Adversary $\mathcal{A}$

$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$

$\xrightarrow{R=(vk_i)_i, (\mathsf{vk}_i, \mathsf{sk}_i)\leftarrow \mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda; r_i)}$

$\xrightarrow{\text{Additionally outputs randomness}(r_i)_i}$

$\xleftarrow{(m^*, R^*, S^*, (ind_{\mathbf{1}}, \dots, ind_t)), \mathsf{vk}^*_{ind_k}\in R}$

$\xrightarrow{\Sigma^* = \mathsf{URS.Sign}(1^\lambda, \mathsf{sk}^*_{ind_k}, m^*, R^*, S^*) \text{ for } k \leftarrow_\$ [t]}$

$\xleftarrow{\text{Guess } k'.}$

Wins if $k = k'$.

The challenge - Recall

Unforgeability

Experiment — Adversary $\mathcal{A}$

$$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$$

$$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$$

Repeatedly:

$$\xleftarrow{\text{may ask to corrupt honest keys}}$$

$$\xleftarrow{\text{or ask for (universal ring) signatures}}$$

When done: $\xleftarrow{(\Sigma^*,m^*,R^*,S^*)}$

Wins if all schemes/keys are honest, m^* not queried to sign and signature verifies.

t-Anonymity

Experiment — Adversary $\mathcal{A}$

$$\xleftarrow{\{\mathsf{ind}_i\}_{i\in[\ell]}}$$

$$\xrightarrow{R=(vk_i)_i,(\mathsf{vk}_i,\mathsf{sk}_i)\leftarrow\mathsf{Sig.KeyGen}_{\mathbf{ind}_i}(1^\lambda;r_i)}$$

$$\xrightarrow{\text{Additionally outputs randomness}(r_i)_i}$$

$$\xleftarrow{(m^*,R^*,S^*,(ind_1,\ldots,ind_t)),\mathsf{vk}^*_{ind_k}\in R}$$

$$\xrightarrow{\Sigma^*=\mathsf{URS.Sign}(1^\lambda,\mathsf{sk}^*_{ind_k},m^*,R^*,S^*)\text{ for } k\leftarrow_\$[t]}$$

$$\xleftarrow{\text{Guess } k'.}$$

Wins if $k=k'$.

Cannot change Gen, only degree of freedom is choice of randomness r_i!

Solution 2 - rough idea

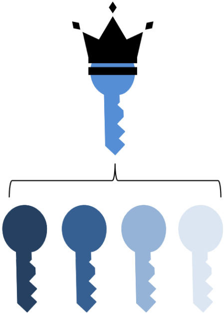
Want: Commitment $\xrightarrow{\textit{replace}}$ WE

Independently chosen keys

Solution 2 - rough idea

Want: Commitment $\xrightarrow{\textit{replace}}$ WE

Independently chosen keys

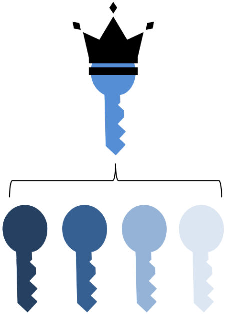


Malformed, correlated keys

Solution 2 - rough idea

Want: Commitment $\xrightarrow{\textit{replace}}$ WE

PRF key



Independently chosen keys

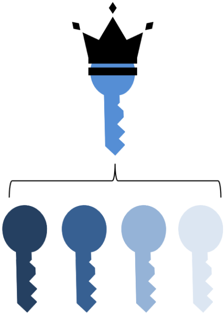
Malformed, correlated keys

Solution 2 - rough idea

Want: Commitment $\xrightarrow{\textit{replace}}$ WE

Independently chosen keys

PRF key



Malformed, correlated keys

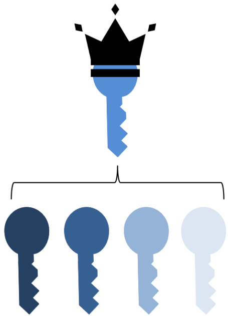
Malformed keys are a much more sparse key distribution.

Solution 2 - rough idea

Want: Commitment $\xrightarrow{\textit{replace}}$ WE

PRF key

Independently chosen keys

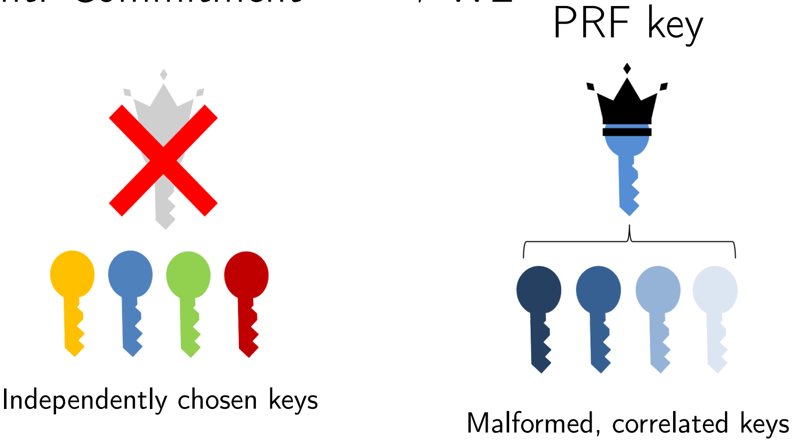


Malformed, correlated keys

Malformed keys are a much more sparse key distribution.
Indistinguishable, but far.

Solution 2 - rough idea

Want: Commitment $\xrightarrow{\textit{replace}}$ WE



Malformed keys are a much more sparse key distribution.
Indistinguishable, but far. PRF key = witness of malformedness

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$, $r_i = \mathsf{PRF}(K, i)$.

Indistinguishable by pseudorandomness. PRF key K is witness!

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$, $r_i = \mathsf{PRF}(K, i)$.

Indistinguishable by pseudorandomness. PRF key K is witness!

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$, $r_i = \mathsf{PRF}(K, i)$.

Indistinguishable by pseudorandomness. PRF key K is witness!
What WE statement to pick?

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$,
$r_i = \mathsf{PRF}(K, i)$.

Indistinguishable by pseudorandomness. PRF key K is witness!

What WE statement to pick?

Malformed in unforgeability, reduction can control all but one honest key

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$,
$r_i = \mathsf{PRF}(K, i)$.

Indistinguishable by pseudorandomness. PRF key K is witness!

What WE statement to pick?

Malformed in unforgeability, reduction can control all but one honest key

$x = R = (\mathsf{vk}_i)_i \in \mathcal{L} \Leftrightarrow \exists$ randomness K s.t. all but one key are malformed

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$,
$r_i = \mathsf{PRF}(K, i)$.

Indistinguishable by pseudorandomness. PRF key K is witness!

What WE statement to pick?

Malformed in unforgeability, reduction can control all but one honest key

$x = R = (\mathsf{vk}_i)_i \in \mathcal{L} \Leftrightarrow \exists$ randomness K s.t. all but one key are malformed

In anonymity: Adversary must use t honest keys, chance of $t-1$ of them being malformed?

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.
Min-Entropy of $t-1$ keys:
$\geq (t-1)\kappa$ (κ lowest key entropy)

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$,
$r_i = \mathsf{PRF}(K, i)$.
Min-Entropy of $t-1$ keys:
λ, as we only choose K

Indistinguishable by pseudorandomness. PRF key K is witness!

What WE statement to pick?

Malformed in unforgeability, reduction can control all but one honest key

$x = R = (\mathsf{vk}_i)_i \in \mathcal{L} \Leftrightarrow \exists$ randomness K s.t. all but one key are malformed

In anonymity: Adversary must use t honest keys, chance of $t-1$ of them being malformed?

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.
Min-Entropy of $t-1$ keys:
$\geq (t-1)\kappa$ (κ lowest key entropy)

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$, $r_i = \mathsf{PRF}(K, i)$.
Min-Entropy of $t-1$ keys:
λ, as we only choose K

Indistinguishable by pseudorandomness. PRF key K is witness!

What WE statement to pick?

Malformed in unforgeability, reduction can control all but one honest key

$x = R = (\mathsf{vk}_i)_i \in \mathcal{L} \Leftrightarrow \exists$ randomness K s.t. all but one key are malformed

In anonymity: Adversary must use t honest keys, chance of $t-1$ of them being malformed? Information theoretically negligible, if $(t-1)\kappa > \lambda$.

Solution 2 - More concretely!

Standard keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
All $r_i \leftarrow_\$ \{0,1\}^\lambda$ are freshly random.
Min-Entropy of $t-1$ keys:
$\geq (t-1)\kappa$ (κ lowest key entropy)

Malformed keys

$(\mathsf{vk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Sig.KeyGen}_{\mathsf{ind}_i}(1^\lambda; r_i)$
Choose random $K \leftarrow_\$ \{0,1\}^\lambda$,
$r_i = \mathsf{PRF}(K, i)$.
Min-Entropy of $t-1$ keys:
λ, as we only choose K

Indistinguishable by pseudorandomness. PRF key K is witness!
What WE statement to pick?
Malformed in unforgeability, reduction can control all but one honest key

$x = R = (\mathsf{vk}_i)_i \in \mathcal{L} \Leftrightarrow \exists$ randomness K s.t. all but one key are malformed

In anonymity: Adversary must use t honest keys, chance of $t-1$ of them being malformed? Information theoretically negligible, if $(t-1)\kappa > \lambda$.
Assume $\kappa = \Theta(\lambda)$ e.g. if $\kappa = \lambda/2$, $t = 4$ sufficient

Putting it together

Sign

- $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_i, m)$.
- $(\mathsf{com}_0, \gamma_0) \leftarrow \mathsf{CS.Commit}(1^\lambda, \sigma)$ and $(\mathsf{com}_1, \gamma_1) \leftarrow \mathsf{CS.Commit}(1^\lambda, 0)$.
- Compute a NIWI proof of correctness
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Putting it together

Sign

- $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_i, m)$.
- $\mathsf{com}_0 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, \sigma)$ and $\mathsf{com}_1 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, 0)$.
- Compute a NIWI proof of correctness
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Putting it together

Sign

- $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_i, m)$.
- $\mathsf{com}_0 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, \sigma)$ and $\mathsf{com}_1 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, 0)$.
- Compute a NIWI proof of correctness
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeability:

challenge vk from Sig-unforgeability,
rest malformed.

Putting it together

Sign

- $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_i, m)$.
- $\mathsf{com}_0 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, \sigma)$ and $\mathsf{com}_1 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, 0)$.
- Compute a NIWI proof of correctness
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeability:

challenge vk from Sig-unforgeability,

rest malformed.

$\Rightarrow$ can extract σ.

Putting it together

Sign

- $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_i, m)$.
- $\mathsf{com}_0 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, \sigma)$ and $\mathsf{com}_1 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, 0)$.
- Compute a NIWI proof of correctness
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeability:
challenge vk from Sig-unforgeability,
rest malformed.
$\Rightarrow$ can extract σ.

t-Anonymity: Honest keys.
$\mathcal{A}$ must include t, needs $\geq t-1$ of
them technically malformed.

Putting it together

Sign

- $\sigma \leftarrow \mathsf{Sig.Sign}(\mathsf{sk}_i, m)$.
- $\mathsf{com}_0 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, \sigma)$ and $\mathsf{com}_1 \leftarrow \mathsf{WE.Enc}(1^\lambda, R, 0)$.
- Compute a NIWI proof of correctness
- Output $\Sigma = (\mathsf{com}_0, \mathsf{com}_1, \pi)$.

Unforgeability:
challenge vk from Sig-unforgeability, rest malformed.
$\Rightarrow$ can extract σ.

t-Anonymity: Honest keys.
$\mathcal{A}$ must include t, needs $\geq t-1$ of them technically malformed.
Negligible extraction chance

Putting it together

Sign

- $\sigma \leftarrow \text{Sig.Sign}(\text{sk}_i, m)$.
- $\text{com}_0 \leftarrow \text{WE.Enc}(1^\lambda, R, \sigma)$ and $\text{com}_1 \leftarrow \text{WE.Enc}(1^\lambda, R, 0)$.
- Compute a NIWI proof of correctness
- Output $\Sigma = (\text{com}_0, \text{com}_1, \pi)$.

Unforgeability:
challenge vk from Sig-unforgeability,
rest malformed.
$\Rightarrow$ can extract σ.

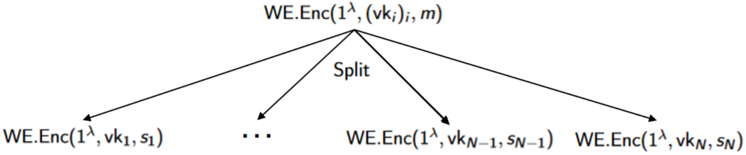
t-Anonymity: Honest keys.
$\mathcal{A}$ must include t, needs $\geq t - 1$ of
them technically malformed.
Negligible extraction chance

WE Size!

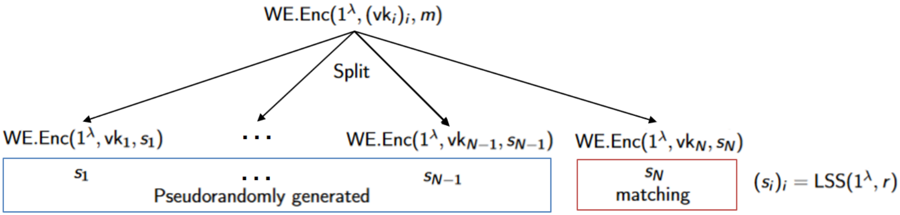
Reducing size

$\mathsf{WE.Enc}(1^\lambda, (\mathsf{vk}_i)_i, m)$

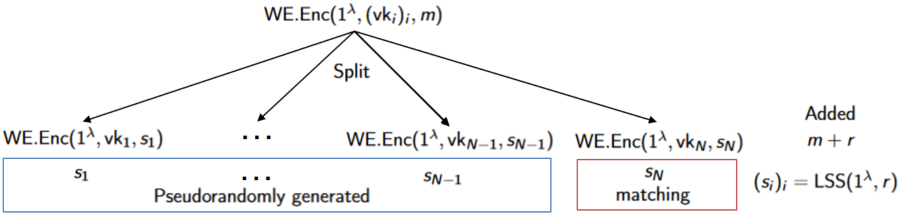
Reducing size



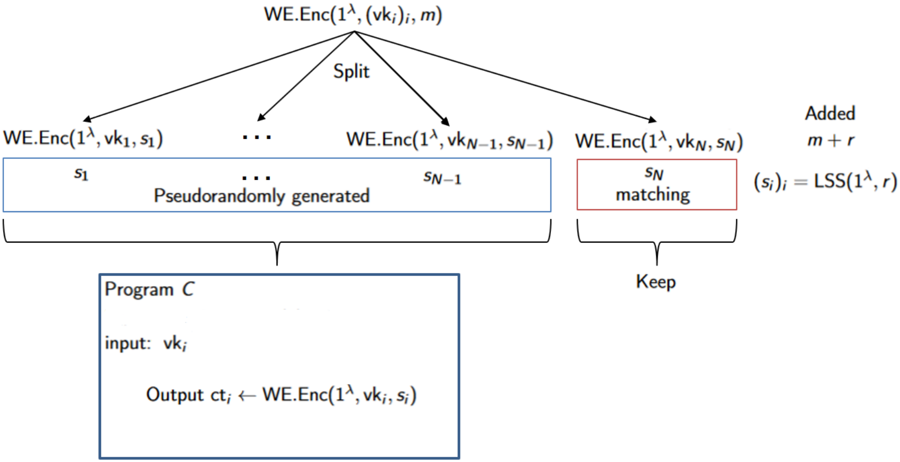
Reducing size



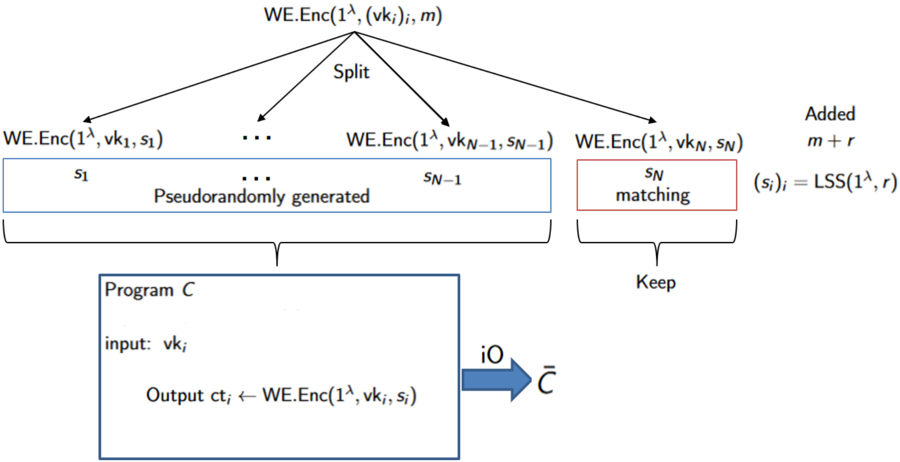
Reducing size



Reducing size



Reducing size



Recap

Result

We have shown, that we can always generate a ring signature for any ad hoc ring, without collaboration and regardless of the signing schemes used.

Constructions

1. Complexity leveraging, superpoly secure schemes (log size)
2. Correlated keys, Witness encryption (linear size in ring)
3. Correlated keys, special WE from iO (log size)

THANKS