One-Shot Fiat-Shamir-based NIZK Arguments of Composite Residuosity and Logarithmic-Size Ring Signatures in the Standard Model

Benoît Libert[1] **Khoa Nguyen**[2] **Thomas Peters**[3] **Moti Yung**[4]

[1]CNRS, Laboratoire LIP (CNRS, ENSL, U. Lyon, Inria, UCBL)
ENS de Lyon (France)

[2]University of Wollongong (Australia)

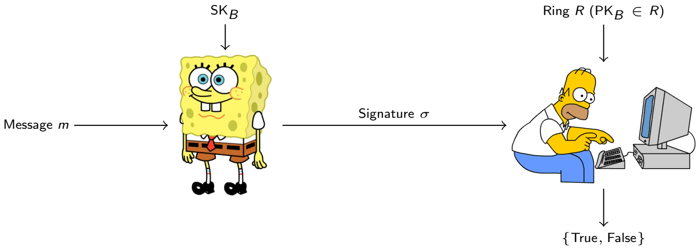
[3]FNRS & UCLouvain, ICTEAM (Belgium)

[4]Google & Columbia University (USA)

June 1, 2022

Ring-Signature (informal) (Rivest-Shamir-Tauman; Asiacrypt'01)

$RSig = (\text{KeyGen}, \text{Sign}, \text{Verify})$.

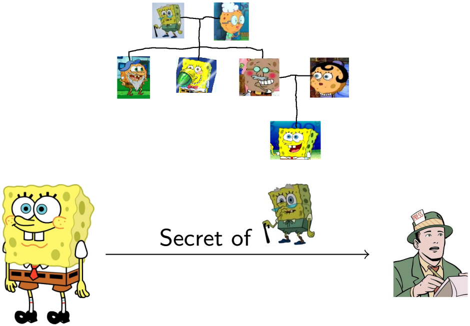


Correctness: If $\text{VK} \in R$, $\text{Verify}(R, \text{Sign}(SK, R, m)) = \text{True}$

Applications: Leak secrets anonymously:

- Whistleblowing
- Cryptocurrencies

Whistleblower



Anonymity under full key exposure: Signatures remain anonymous, even if the adversary knows all secret keys of the ring.

Unforgeability w.r.t. insider corruption: Infeasibility of signing without a ring member's secret key.

Σ-protocols

3-move protocols with transcripts (a, Chall, z)

Common input: P and V both have a statement x

Private input: P has a witness w showing that $x \in L$

1. P sends a *commitment* a to V
2. V sends a random *challenge* $\mathsf{Chall} \in_R \{0,1\}^\lambda$
3. P sends a *response* z

Given (a, Chall, z), V outputs 0 or 1

- **Special-Soundness:** transcripts $(a, \mathsf{Chall}_1, z_1)$, $(a, \mathsf{Chall}_2, z_2)$ reveal a witness
- $(n+1)$-**Special-Soundness:** transcripts $\{(a, \mathsf{Chall}_i, z_i)\}_{i=1}^{n+1}$ reveal a witness

$\Rightarrow$ For a false statement $x \notin L$, up to n bad challenges may exist

Fiat-Shamir: From Σ-protocol to Non-Interative Proof

- Compiles Σ-protocols into NIZK proofs in the ROM [BR91]
  1. Compute a *commitment* a
  2. Compute a random *challenge* $\mathsf{Chall} = H(x, a) \in_R \{0,1\}^\lambda$
  3. Compute a *response* z, and output $\pi = (\mathsf{Chall}, z)$
- Does not guarantee soundness in the standard model [Bar01,GT03]

Fiat-Shamir: From Σ-protocol to Non-Interative Proof

- Compiles Σ-protocols into NIZK proofs in the ROM [BR91]
  1. Compute a *commitment* a
  2. Compute a random *challenge* Chall $= H(x, a) \in_R \{0, 1\}^\lambda$
  3. Compute a *response* z, and output $\pi = (\text{Chall}, z)$
- Does not guarantee soundness in the standard model [Bar01,GT03]
- Instantiable for some protocols and **correlation intractable** hash functions [CGH98]:

  For some relation R, finding x s.t. $(x, H(x)) \in R$ is hard

- Canetti *et al.* (STOC'19): CIH functions for efficiently searchable relations

  For any $y \in \mathcal{Y}$, at most one (efficiently computable) $x \in \mathcal{X}$ satisfies $(x, y) \in R$

  $\Rightarrow$ Compiles **trapdoor** Σ-protocols into non-interactive FS proofs

Fiat-Shamir from Trapdoor Σ-protocols

Trapdoor Σ-protocols [CLW19]: for unique/enumerable relations

- Assume a CRS with a trapdoor τ
- For a false statement $x \notin L$ and a first prover message a
  - τ allows computing bad challenges $\{\mathsf{Chall}_i\}_{i=1}^{n}$ for which a valid response z_i exists
  - Allows applying CI hash functions [CLW19,PS19] when $n \in \mathsf{poly}(\lambda)$

Fiat-Shamir from Trapdoor Σ-protocols

Trapdoor Σ-protocols [CLW19]: for unique/enumerable relations

- Assume a CRS with a trapdoor τ
- For a false statement $x \notin L$ and a first prover message a
  - τ allows computing bad challenges $\{\mathsf{Chall}_i\}_{i=1}^n$ for which a valid response z_i exists
  - Allows applying CI hash functions [CLW19,PS19] when $n \in \mathsf{poly}(\lambda)$
- Generically implied by Σ-protocols with binary challenges (Ciampi *et al.*, SCN'20)
- So far, all instantiations use $O(\lambda)$ parallel repetitions

Fiat-Shamir from Trapdoor Σ-protocols

Trapdoor Σ-protocols [CLW19]: for unique/enumerable relations

- Assume a CRS with a trapdoor τ
- For a false statement $x \notin L$ and a first prover message a
  - τ allows computing bad challenges $\{\mathsf{Chall}_i\}_{i=1}^n$ for which a valid response z_i exists
  - Allows applying CI hash functions [CLW19,PS19] when $n \in \mathsf{poly}(\lambda)$
- Generically implied by Σ-protocols with binary challenges (Ciampi *et al.*, SCN'20)
- So far, all instantiations use $O(\lambda)$ parallel repetitions

Warning: number of bad challenges blows up with parallel repetitions if the basic Σ-protocol has $(n+1)$-special soundness with $n > 1$

Fiat-Shamir from Trapdoor Σ-protocols

Trapdoor Σ-protocols [CLW19]: for unique/enumerable relations

- Assume a CRS with a trapdoor τ
- For a false statement $x \notin L$ and a first prover message a
  - τ allows computing bad challenges $\{\mathsf{Chall}_i\}_{i=1}^{n}$ for which a valid response z_i exists
  - Allows applying CI hash functions [CLW19,PS19] when $n \in \mathsf{poly}(\lambda)$
- Generically implied by Σ-protocols with binary challenges (Ciampi *et al.*, SCN'20)
- So far, all instantiations use $O(\lambda)$ parallel repetitions

Warning: number of bad challenges blows up with parallel repetitions if the basic Σ-protocol has $(n+1)$-special soundness with $n > 1$

Our goal

Standard-model instantiations in **one shot** under standard assumptions

Trapdoor Σ-protocol and one-shot NIZK for DCR

Let $N = pq$ and $L := \{x = w^N \bmod N^2 \mid w \in \mathbb{Z}_N^*\}$

$$\begin{tikzcd}
P \arrow[rrr, "a := r^N \bmod N^2"] \arrow[d, Rightarrow] & & & V \arrow[d, Rightarrow] \\
\bullet \arrow[d, Rightarrow] & & & \bullet \arrow[lll, "\mathsf{Chall} \in \{0, \ldots, 2^\lambda - 1\}"'] \arrow[d, Rightarrow] \\
\bullet \arrow[rrr, "z := r \cdot w^{\mathsf{Chall}} \bmod N"] & & & \bullet
\end{tikzcd}$$

V accepts iff $a \cdot x^{\mathsf{Chall}} = z^N \bmod N^2$

Trapdoor Σ-protocol and one-shot NIZK for DCR

Let $N = pq$ and $L := \{x = w^N \bmod N^2 \mid w \in \mathbb{Z}_N^*\}$

$$\begin{tikzcd}
P \arrow[rrr, "a := r^N \bmod N^2"] \arrow[d, Rightarrow] & & & V \arrow[d, Rightarrow] \\
\bullet \arrow[d, Rightarrow] & & & \bullet \arrow[lll, "\mathsf{Chall} \in \{0, \ldots, 2^\lambda - 1\}"'] \arrow[d, Rightarrow] \\
\bullet \arrow[rrr, "z := r \cdot w^{\mathsf{Chall}} \bmod N"] & & & \bullet
\end{tikzcd}$$

V accepts iff $a \cdot x^{\mathsf{Chall}} = z^N \bmod N^2$

BadChallenge(x, a): (cf. `Lipmaa, FC'17`)

1. Compute $\alpha_x = \mathcal{D}_{p,q}(x) \in \mathbb{Z}_N$ and $\alpha_a = \mathcal{D}_{p,q}(a) \in \mathbb{Z}_N$
2. Let the congruence

$$\alpha_a + \alpha_x \cdot \mathsf{Chall} \equiv 0 \pmod{N/\gcd(\alpha_x, N)}$$

3. Output $\mathsf{Chall} = \alpha_x^{-1} \cdot \alpha_a \bmod \frac{N}{\gcd(\alpha_x, N)}$ if it fits in $\{0, \ldots, 2^\lambda - 1\}$ and $\perp$ otherwise.

State-of-the-art ring signatures in the standard model

- Bender-Katz-Morselli (TCC'06): generic construction from ZAPs
- Shacham-Waters (PKC'07): efficiently using a CRS, $O(R)$-size signatures
- Chandran-Groth-Sahai (ICALP'07): using a CRS, $O(R^{1/2})$-size signatures
- Gonzalez (PKC'19): using a CRS, $O(R^{1/3})$-size signatures
- Backes *et al.* (Eurocrypt'19): no CRS, $O(\log R)$-size signatures
- Chatterjee *et al.* (Crypto'21): no CRS, $O(\log R)$-size signatures from LWE

This paper

- Assumes a CRS; $O(\log R)$-size signatures
- Concretely short signatures (comparable to ROM-based schemes) from DCR+LWE

Short Log-Size Ring signatures in the CRS model

Adaptation of Groth-Kohlweiss (`Eurocrypt'15`) which gives $O(\log R)$-size in the ROM

GK15 at a high level

- Each public key is an additively homomorphic commitment to 0
- $O(\log R)$-communication protocol showing that one-out-R commitment opens to 0

Short Log-Size Ring signatures in the CRS model

Adaptation of Groth-Kohlweiss (`Eurocrypt'15`) which gives $O(\log R)$-size in the ROM

GK15 at a high level

- Each public key is an additively homomorphic commitment to 0
- $O(\log R)$-communication protocol showing that one-out-R commitment opens to 0

Difficulty #1: computing bad challenges in the DLOG setting

- Repeating $O(\lambda/\log\lambda)$ times a small-challenge Σ-protocol fails:

  Parallel repetitions yield $O((\log R)^{\lambda/\log\lambda})$ bad challenges ($O(\log R)$ per iteration)

- **First idea:** adapt GK15 to the DCR setting; use the DCR structure to compute bad challenges in a $O(2^{\lambda})$-size space

Short Log-Size Ring signatures: High-level ideas

Our adaptation of GK15

- Each public key is a DCR commitment $vk = \mathsf{com}(0\,;w)$
- Trapdoor Σ-protocol showing that one-out-R commitment opens to 0
- **BadChallenge** computes the roots of a polynomial of degree $r = O(\log R)$ over $\mathbb{Z}_N$ and outputs those in $\{0,\ldots,2^\lambda - 1\}$

  $\Rightarrow$ Efficiently enumerable relation, compatible with LWE-based CI hash functions

Short Log-Size Ring signatures: High-level ideas

Our adaptation of GK15

- Each public key is a DCR commitment $vk = \mathrm{com}(0\,; w)$
- Trapdoor Σ-protocol showing that one-out-R commitment opens to 0
- **BadChallenge** computes the roots of a polynomial of degree $r = O(\log R)$ over $\mathbb{Z}_N$ and outputs those in $\{0, \ldots, 2^\lambda - 1\}$

  $\Rightarrow$ Efficiently enumerable relation, compatible with LWE-based CI hash functions

Difficulty $\#2$**:** How to prove unforgeability without the ROM?

- GK15 uses the forking lemma (not an option in the standard model)
- **Second idea:** argue membership instead of knowledge (no need to rewind)
  - Use (unbounded) simulation-sound arguments

    $\Rightarrow$ We give new DCR-based USS arguments from lossy encryption
  - Force a forgery to argue a false statement

Short Log-Size Ring signatures (cont.)

Our adaptation of GK15

- Each public key is a DCR commitment $vk = \mathsf{com}(0\,;w)$
- $O(\log R)$-communication protocol showing that one-out-R commitment opens to 0

Difficulty $\#3$: How to define a true/false statement?

- In GK15, signatures commit to the signer's position $\ell^\star \in [R]$ in the ring
- We use dual-mode (instead of perfectly hiding) commitments
  - True statement: Ring $\{vk_1, \ldots, vk_R\}$ such that $vk_{\ell^\star} = \mathsf{com}(0\,;w)$
  - Security proof guesses $\ell^\star \in [R]$ with proba $1/R$ and uses DCR to reach a game where $vk_{\ell^\star} = \mathsf{com}(1\,;w)$

    $\Rightarrow$ Forgery breaks simulation-soundness

Short Log-Size Ring signatures (cont.)

Our adaptation of GK15

- Each public key is a DCR commitment $vk = \mathrm{com}(0\,; w)$
- $O(\log R)$-communication protocol showing that one-out-R commitment opens to 0
  - GK15: $vk_\ell \in \{vk_1, \ldots, vk_R\}$, commit to the bits of $\ell = \ell_1\ell_2\cdots\ell_r$

Short Log-Size Ring signatures (cont.)

Our adaptation of GK15

- Each public key is a DCR commitment $vk = \mathsf{com}(0\,;w)$
- $O(\log R)$-communication protocol showing that one-out-R commitment opens to 0
  - GK15: $vk_\ell \in \{vk_1, \ldots, vk_R\}$, commit to the bits of $\ell = \ell_1\ell_2\cdots\ell_r$

Difficulty $\#$4**:** How to handle corruptions without erasures?

- Reduction is stuck when it has to explain NIZK-simulated signatures
- We only simulate signatures involving $vk_{\ell^\star}$ (index $\ell^\star$ is guessed in advance)

  $\Rightarrow$ With probability $1/R$, $vk_{\ell^\star}$ never gets corrupted
- **Problem:**
  - Decoding $\ell^\star$ from forgery requires extractable commitments
  - We need statistical NIZK in signing queries involving $vk_{\ell^\star}$ to keep guess for $\ell^\star$ hidden

    $\Rightarrow$ We build sometimes extractable perfectly hiding commitments from DCR (commitment key programmed using admissible hash functions)

Short Log-Size Ring signatures (cont.)

Difficulty $\#5$**:** How to rely on DCR for $vk_{\ell^\star}$ and extract $\ell^\star$?

- Reduction is stuck when $vk_{\ell^\star}$: $\mathsf{com}(0\,;w^\star) \to \mathsf{com}(1\,;w^\star)$
- We always need to extract $\ell^\star = \ell_1^\star \ell_2^\star \cdots \ell_r^\star$ thanks to a DCR membership trapdoor

  $\Rightarrow$ Works in distinct groups: i.e., makes use of distinct moduli
- **Problem:**
  - GK15 works by "carrying" the bits ℓ_j over the vk's to securely select vk_ℓ
    More precisely: the vk's are raised to the power of the responses $z_j = \ell_j \mathsf{Chall} + r_j$

Short Log-Size Ring signatures (cont.)

Difficulty $\#5$**:** How to rely on DCR for $vk_{\ell^\star}$ and extract $\ell^\star$?

- Reduction is stuck when $vk_{\ell^\star}$: $\mathsf{com}(0\,; w^\star) \rightarrow \mathsf{com}(1\,; w^\star)$
- We always need to extract $\ell^\star = \ell_1^\star \ell_2^\star \cdots \ell_r^\star$ thanks to a DCR membership trapdoor

  $\Rightarrow$ Works in distinct groups: i.e., makes use of distinct moduli
- **Problem:**
  - GK15 works by "carrying" the bits ℓ_j over the vk's to securely select vk_ℓ
    More precisely: the vk's are raised to the power of the responses $z_j = \ell_j \mathsf{Chall} + r_j$

Our adaptation of GK15

- Each public key is a DCR commitment $vk = \mathsf{com}(0\,; w)$
- $O(\log R)$-communication protocol showing that one-out-R commitment opens to 0
  - Manage to "carry" the bits of $\ell = \ell_1 \ell_2 \cdots \ell_r$ over the integers

Additional difficulty: Proving anonymity when rings contain malformed keys

- Need dual-mode commitments where the statistically hiding mode is dense in $\mathbb{Z}^*_{N^2}$
- Use $\mathsf{com}(m; (y, w)) = (1 + N)^m \cdot h^y \cdot w^N \bmod N^2$ with $h \sim U(\mathbb{Z}^*_{N^2})$

Additional difficulty: Proving anonymity when rings contain malformed keys

- Need dual-mode commitments where the statistically hiding mode is dense in $\mathbb{Z}_{N^2}^*$
- Use $\mathsf{com}(m; (y, w)) = (1 + N)^m \cdot h^y \cdot w^N \bmod N^2$ with $h \sim U(\mathbb{Z}_{N^2}^*)$

Conclusion

- First one-shot Trapdoor Σ-protocols (i.e., with negligible soundness error)
- Ring signature with short keys: $vk = \mathsf{com}(0\,; (y, w))$ and $sk = (y, w)$
- Signature size: $15 \log R + 7$ group elements (v.s. $5 \log R + 1$ in the ROM)

$\longrightarrow$

*"Concretely short privacy-preserving signature
in the standard model without pairing is feasible"*

Thank you!

Questions?