Short Accountable Ring Signatures
Based on DDH

Jonathan Bootle, Andrea Cerulli, Pyrros Chaidos, Essam Ghadafi, Jens Groth, and Christophe Petit

University College London
Signature Schemes

Link message to single entity

– Signer
– Verifier

\[ S \xrightarrow{m, \sigma} V \]
Signature Schemes

Link message to single entity

– Signer
– Verifier

• Link message to multiple entities:
Signature Schemes

Link message to single entity

- Signer
- Verifier

• Link message to multiple entities:

Ring Signatures

- Users
- Verifier
Signature Schemes

Link message to single entity

- Signer
- Verifier

\[ S \rightarrow V \]

• Link message to multiple entities:

Ring Signatures

- Users
- Verifier

\[ S \rightarrow V \]

\[ S', \sigma', R' \]
Signature Schemes

Link message to single entity

- Signer
- Verifier

Link message to multiple entities:

Ring Signatures

- Users
- Verifier

Group Signatures

- Manager
- Users
- Verifier
Signature Schemes

Link message to single entity

- Signer
- Verifier

- Link message to multiple entities:

**Ring Signatures**

- Users
- Verifier

\[ m, \sigma, R \]

**Group Signatures**

- Manager
- Users
- Verifier

\[ m, \sigma \]

\[ m', \sigma', R' \]
Signature Schemes

Link message to single entity

- Signer
- Verifier

• Link message to multiple entities:

Ring Signatures

- Users
- Verifier

Group Signatures

- Manager
- Users
- Verifier
Accountable Ring Signatures [Xu and Yung]

Link message to multiple entities

• Users
• Opener(s)
• Verifier
Accountable Ring Signatures

- Setup, OpenerKeyGen, UserKeyGen
- Sign, Vfy
- Open, Judge

Security:
- Correctness
- Full Unforgeability
- Anonymity
- Traceability with Tracing Soundness
Components for Accountable Ring Signatures

- One-way functions ($g^x$)
- Homomorphic Commitments (Pedersen)
  - $C_{ck}(m_1) \cdot C_{ck}(m_2) = C_{ck}(m_1 \cdot m_2)$
- IND-CPA Encryption (ElGamal)
- Non-Interactive Zero Knowledge Proofs
- Signatures of Knowledge
**Σ-Protocols**

- 3-move protocols for some NP relation $R$
- Prover demonstrates a statement $x \in L_R$: there exists $w$ s.t. $(x, w) \in R$

![Diagram of protocol interaction]

- Completeness: outputs 1 for $x \in L_R$
- $n$-Special Soundness: $n$ accepting $e, z$ pairs for same $x, a$: we obtain $w$
- Special Honest Verifier Zero Knowledge: Transcripts between and honest can be efficiently simulated for any challenge $e$
Non-Interactive Zero Knowledge Proofs

- 1-move protocols for some NP relation $R$
- Fiat-Shamir: challenge is hash of the transcript

• Completeness: outputs 1 for $x \in L_R$
• Soundness: If $x \notin L_R$, almost never outputs 1
• Zero Knowledge: Proofs can be efficiently simulated
Signatures of Knowledge

- 1-move protocols for some NP relation $R$, given common reference string $crs$
- Prover demonstrates, w.r.t. message $m$, knowledge of $w$ for statement $x \in L_R$: $(x, w) \in R$

- Extractability: If produces good signatures, extract $w$ by rewinding
- Straightline $f$-Extractability: we can extract $f(w)$ without rewinding
- Simulatability: signatures can be efficiently simulated

- Extractor, Simulator is given control of $crs$ creation
Construction

• Setup: Choose discrete log group $G$, generator $g$ and common reference string $crs$

• OpenerKeyGen: Create ElGamal keypair, publish $pk$

• UserKeyGen: Pick secret key $sk$, output verification key $vk = g^{sk}$
Signing

- Choose ring $R = \{vk_0, vk_1, \ldots, vk \ldots, vk_k\}$

- Prove $vk \in R$
- Attach encryption $c$ of $vk$ so opener can trace
- Prove knowledge of $sk = \log(vk)$
- Prove knowledge, correctness of $c$
- Bind $\sigma$ to message $m$ via Fiat-Shamir

$$R_{sig} = \left\{ (R, c), (sk, r) : (vk \in R \land vk = g^{sk} \land c = E(vk; r)) \right\}$$
Signing

• Choose ring \( R = \{vk_0, vk_1, \ldots, vk \ldots, vk_k\} \)

• Prove \( vk \in R \)

• Attach encryption \( c \) of \( vk \) so opener can trace

• Prove knowledge of \( sk = \log(vk) \)

• Prove knowledge, correctness of \( c \)

• Bind \( \sigma \) to message \( m \) via Fiat-Shamir

\[
R_{\text{sig}} = \left\{ (R, c), (sk, r) : \begin{array}{l}
vk \in R \land vk = g^{sk} \land c = E(vk; r) \\
\end{array} \right\}
\]
Signing

- Choose ring $R = \{vk_0, vk_1, \ldots, vk \ldots, vk_k\}$
- Prove $vk \in R$

Could prove: $vk = vk_0$ OR $vk = vk_1$ OR $\ldots$ OR $vk = vk_k$

- Linear size: too big for large rings

Use One-out-of-Many proof by Groth and Kohlweiss

- Take $c_i = c/E(vk_i ; 0)$
- Use modified GK to show one node encrypts 1
GK idea

• We want to open $c_l$ without revealing $l$

• $c_l = \prod c_i^{\Delta_i}$, where $\Delta_i = 1 \iff i = l$

• Commit to $\Delta_i$. Also commit to blinders $a_i$

• Given challenge $x$, reply with $f_i = x \cdot \Delta_i + a_i$

• $\prod c_i^{f_i} = c_l^x \cdot \prod c_i^{a_i}$
GK idea

- We want to open $c_l$ without revealing $l$
- $c_l = \prod c_i^{\Delta_i}$, where $\Delta_i = 1 \iff i = l$
- Commit to $\Delta_i$. Also commit to blinders $a_i$
- Given challenge $x$, reply with $f_i = x \cdot \Delta_i + a_i$
- $\prod c_i^{f_i} = c_l^x \cdot \prod c_i^{a_i}$
- $G = \prod c_i^{a_i}$ does not depend on $x$. Rerandomize as $G'$
GK idea

• We want to open $c_l$ without revealing $l$

• $c_l = \prod c_i^{\Delta_i}$, where $\Delta_i = 1 \iff i = l$

• Commit to $\Delta_i$. Also commit to blinders $a_i$, reveal $G'$

• Given challenge $x$, reply with $f_i = x \cdot \Delta_i + a_i$

• $\prod c_i^{f_i} = c_l^x \cdot \prod c_i^{a_i}$

• $G = \prod c_i^{a_i}$ does not depend on $x$. Rerandomize as $G'$
GK idea

• We want to open $c_l$ without revealing $l$

• $c_l = \prod c_i^{\Delta_i}$, where $\Delta_i = 1 \iff i = l$

• Commit to $\Delta_i$. Also commit to blinders $a_i$, reveal $G'$

• Given challenge $x$, reply with $f_i = x \cdot \Delta_i + a_i$

• $\prod c_i^{f_i} = c_l^x \cdot \prod c_i^{a_i}$

• $G = \prod c_i^{a_i}$ does not depend on $x$. Rerandomize as $G'$

• $\prod c_i^{f_i} / G' = c_l^x \cdot E(1; r) = E(1; r')$
n-tree GK

Split $i, \Delta_i$ by level:

$$i = \sum i_j \cdot n^j$$

$$\delta_{i,j} : \Delta_i = \prod \delta_{i,j}$$
n-tree GK

\[ \delta_0 = [0,1,0] \]

\[ \delta_1 = [1,0,0] \]

- Split \( i, \Delta_i \) by level: 
  \[ i = \sum i_j \cdot n^j \]
  \[ \delta_{i,j} : \Delta_i = \prod \delta_{i,j} \]
- Commit to \( \delta_{i,j} \), prove 0/1, for each \( j \) exactly one \( \delta_{i,j} \) is 1
n-tree GK

- Commit to $\delta_{j,i_j}$. Also commit to blinders $a_{i,j}$
- Given challenge $x$, reply with $f_{j,i_j} = x \cdot \delta_{j,i_j} + a_{j,i_j}$
- Let $p_i(x) = \prod f_{j,i_j}$
- Key point: $x^m$ appears only if all $\delta_{j,i_j}$ are 1 i.e $i = l$
- $p_i(x) = \Delta_i x^m + \sum_{k=0}^{m-1} p_{i,k} x^k$ where $p_{i,k}$ depend on $l, a_{j,i_j}$
- $\prod c_i^{p_i(x)} = c_l \cdot \prod_{k=0}^{m-1} P_k x^k$
- $P_k$ do not depend on $x$. 
n-tree GK

- $P_k$ do not depend on $x$
- We commit beforehand as $G_k$

- What is $\prod c_i \prod f_{i,j} \prod_{k=0}^{m-1} G_k x^{-k}$?

- If $c_l$ is an encryption of 1, result is encryption of 1

- Otherwise, with overwhelming probability it’s an encryption of a value $\neq 1$, so can’t be opened to 1
Opening

• Open
  – Check if $\sigma$ actually verifies
  – Decrypt ciphertext $c$ attached in signature
  – Prove correctness of decryption in Zero Knowledge

• Judge
  – Check decryption correctness
Simulated Opening & Straightline Extractability

• To prove anonymity, we do an IND-CCA style proof
  – Need to extract $vk$ from sigs
  – Can’t see the key

• Adversary can obstruct rewinding
  – Adversary’s signatures related to each other
  – Rewinding to open one changes previous $\Rightarrow$ more rewinding

• We need to extract $vk = g^{sk}$ with no rewinding
  – Cheap solution: Attach 2nd encryption of $vk$ to proof [NY]
  – Simulator has 2nd key in simulation
  – Nobody has the key in real world
Efficiency

- \( \log N + 12 \) Group Elements
- \( \frac{3}{2} \log N + 6 \) Field Elements
- Competitive vs sRSA/DDH schemes

| Scheme                        | \( |R| = 128 \) | \( |R| = 1024 \) | \( |R| = 1 \text{Mi} \) |
|-------------------------------|--------------|--------------|-----------------|
| [CG05] – 2048 sRSA + d.Log    | 10 Kib       | 10 Kib       | 10 Kib          |
| This – 192 ECC                | 6.7 Kib      | 8.1 Kib      | 12.75 Kib       |
| This – 192 ECC                | 7.8 Kib      | 9.4 Kib      | 14.875 Kib      |

- Linear expos (or worse) to Sign
- Linear expos to Verify
Summary

• Accountable Ring Signatures can be best of both worlds
  – Tracing functionality of Group sigs
  – Free choice of ring
  – Free choice of opener
  – Can derive Ring and Group signatures

• Signature size:
  – Competitive vs sRSA/DDH schemes
  – Better than 50% size improvement over original GK construction: binary $\rightarrow n$-tree, mixed Com+Enc
Thanks!