The BBS Signatures Scheme

Tobias Looker, Vasilis Kalos, Andrew Whitehead, Mike Lodder
BBS Signatures Recap

- Deterministic multi message signature.
- Unlikable proofs (zk-proof of knowledge) supporting selective disclosure of messages.
- Header: always revealed value (e.g., alg identifier, token type etc.).
- Presentation header: value bound to the proof (e.g., random nonce etc.).
Updates in v4

- Separate Proof operations to subroutines.
- Separate high-level API, from core operations.
Updates: Separate Proof operations to subroutines.

Proof Generation

- Proof Init
  \[ \text{initRes} \leftarrow \text{ProofInit}(\ldots\text{inputs}) \]
- Challenge Calculation
  \[ \text{challenge} \leftarrow \text{ChallengeCalc}(\text{initRes}) \]
- Proof Finalize
  \[ \text{proof} \leftarrow \text{ProofFinalize}(\text{challenge}, \text{initRes}) \]

Examples:

2. Anonymous revocation with accumulators, e.g., [VB22]
Updates: Separate high-level API, from core operations.

- Removed Create Generators and Process Messages operations from the ciphersuite.
- Interface operations manage creating the generators and processing the messages.
- Core operations accept generators and processed messages as input.
- Allows application specific generators and pre-processing of messages.
New BBS Proof
BBS Proof Generation Process

- Type 2 pairings
- Not as efficient

- Extended on type 3 pairings
- Uses extra scalar on signature

- SUF without the extra scalar
- PoK soundness reduced to q-SDH

- Uses alternative PoK based on [CDL16]
- Soundness does not rely on computational assumptions
- Easier to combine with other ZKPs

References:
- [ASM06]
- [CDL16, BBDT17]
- [TZ23]
- [BBS CFRG Draft]

Symbols:
- $(A, e)$
- $(A, e, s)$
Blind BBS Signatures
BBS Blind Signatures

1. Commitment on messages
2. Proof of correctness of

Messages known to the Issuer

Signature on and

Proof presentation

Issuer

Prover

Verifier
BBS Blind Signatures

- No “unblinding” needed. The first signed message becomes the commitments “blinding factor” (never disclosed).

- Used for User binding, hiding pseudonyms from the Issuer etc.

- The Verifier should not know if a message is committed by the Prover or the Issuer.

- Better suited for a different document or should be merged with the “core” draft?

**Diagram:**

- **Issuer**
  - Commitment on messages

- **Prover**
  - Proof presentation

- **Verifier**
  - sig
References


Thank you!

Questions?