

# Privacy-Preserving Age Verification Using Zero-Knowledge Proofs

---

Ali Daher   Ali Khalil

Department of Computer Science, American University of Beirut

## Problem & Goal

- Online services need age checks without exposing personal data. (Privacy, tracking, over-collection)
- Goal: Prove “age  $\geq$  threshold” and validity, with **no identity leakage** and **unlinkability**.
- Approach: Combine Pedersen commitments, BBS+ signatures (BLS12-381), Bulletproofs, and RSA-accumulator revocation.

Goal: Prove knowledge of  $x$  such that  $X = g^x \pmod{p}$  without revealing  $x$ .

- **Setup:** Public parameters  $(g, p, q)$  where  $g$  generates a group of prime order  $q$ .
- **Prover:** Chooses random  $y \in \mathbb{Z}_q$ , sends  $Y = g^y$ .
- **Verifier:** Sends random challenge  $c \in \mathbb{Z}_q$ .
- **Prover:** Responds with  $r = y + cx \pmod{q}$ .
- **Verifier:** Checks  $g^r \stackrel{?}{=} Y X^c$ .

*Properties:* Zero-knowledge (no leak of  $x$ ), sound (cannot cheat without knowing  $x$ ), and efficient.

$$C = g^m h^r \quad \text{in a group of prime order } q.$$

- **Commit:** Choose random  $r \in \mathbb{Z}_q$  and compute  $C$  as above.
- **Reveal:** Send  $(m, r)$  to open the commitment.
- **Verify:** Check that  $C = g^m h^r$ .

*Properties:* Perfectly hiding (no information about  $m$ ), computationally binding (cannot change  $m$  once committed).

# Bulletproofs (Range Proofs)

Prove  $0 \leq v < 2^n$  without revealing  $v$ .

Compact (log-size) proof, no trusted setup.

## Overview

- **Bulletproofs** are short, non-interactive **range proofs**.
- Based on **inner product arguments** for efficiency.
- No trusted setup, fully transparent.

## Uses

- Confidential transactions (e.g., Monero, Bitcoin).
- Showing age or value in a valid range.
- Proving credentials (e.g., not expired).

# BBS+ Signatures (BLS12-381)

## Overview

- A **BBS+ signature** allows signing multiple messages at once.
- Later, the holder can reveal only selected messages and still prove the signature is valid known as **selective disclosure**.
- Built on **pairing-based cryptography** using the **BLS12-381** curve for 128-bit security.

## Core Idea

- The issuer signs a set of messages  $\{m_i\}$  producing a compact signature  $(A, e)$ .
- Verification uses a pairing equation that ensures integrity of all signed messages.
- During disclosure, the holder can generate a zero-knowledge proof that convinces a verifier the signature is valid—without revealing hidden messages.

# RSA Accumulator (Basic Idea)

## Overview

- An **RSA accumulator** is a compact way to represent a set of values using one number.
- Built on the **RSA assumption** — hard to factor large numbers.
- Used to prove that an item **is or is not** part of a set without revealing the whole set.
- Commonly used for **revocation lists** in anonymous credential systems.

## How It Works

- The issuer combines all elements into one value called the **accumulator**.
- Each user gets a small **witness** proving their element is included.
- Verification uses simple modular arithmetic and stays **constant-size**, no matter how large the set is.

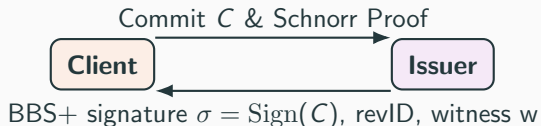
**Security Properties:** Unforgeable • Compact • Efficient • Privacy-friendly

- Three parties: **Issuer**, **Client**, **Verifier**.
- Core idea: single commitment with hidden attributes, certified by BBS+, and proven with ZK subproofs.

$$C = g_1^{m_{\text{birth}}} g_2^{m_{\text{exp}}} g_3^{x_d} h^r.$$



## Issuance Phase (Flow)



### Equations:

$$C = g_1^{m_{\text{birth}}} g_2^{m_{\text{exp}}} g_3^{x_d} h^r, \quad \sigma = \text{BBS+Sign}(C). ;$$

### Client stores:

- The issued credential  $(C, \sigma, \text{revID}, w)$
- The private randomness  $r$  and hidden attributes  $(m_{\text{birth}}, m_{\text{exp}}, x_d)$

## Verification Phase (1/2): Client-side preparation

Client

Verifier

### Randomize credential & initialize proofs

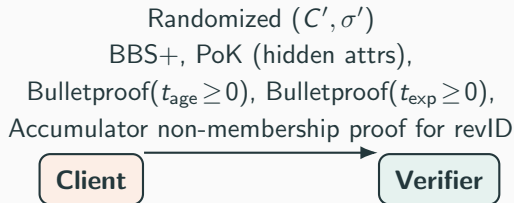
- Randomize commitment and signature:  $C' = C \cdot h^{r'}$ ,  $\sigma' = \text{ReRand}(\sigma, r')$
- Prepare BBS+ proof of knowledge (PoK) over hidden attributes  $(m_{\text{birth}}, m_{\text{exp}}, x_d, r)$ .
- Set up Bulletproof witnesses:

$$t_{\text{age}} = \text{age} - \text{threshold}, \quad t_{\text{exp}} = \text{expiry\_date} - \text{today}.$$

Target: prove  $t_{\text{age}} \geq 0$  and  $t_{\text{exp}} \geq 0$  via range proofs.

- Compute revocation handle  $e = \text{SHA-256}(x_d)$  and fetch its accumulator witness  $w$  to generate a non-membership proof yet with pedersen:  $U = g^e h^r$ .

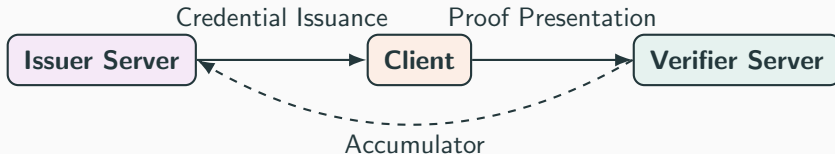
## Verification Phase (2/2): Build and send proofs



### Proof bundle:

- $\Pi_{\text{BBS}+}$ : PoK linking  $(C', \sigma')$  to hidden messages. (AND-composed)
- $\Pi_{\text{age}}$ : Bulletproof that  $t_{\text{age}} \geq 0$ .
- $\Pi_{\text{exp}}$ : Bulletproof that  $t_{\text{exp}} \geq 0$ .
- $\Pi_{\text{acc}}$ : Non-membership of revID in accumulator.

# System Overview (Entities & Roles)



## Roles:

- **Issuer:** Issues credentials, manages RSA accumulator (revocation list), publishes public parameters.
- **Client:** Holds credentials, generates ZK proofs (BBS+, Bulletproofs, revocation).
- **Verifier:** Validates proofs, checks non-revocation.

## Architecture

- All components in **Go**; lightweight REST APIs; thin JS UI.
- Client runs local Go backend; cryptography stays off the browser.
- Verifier caches issuer params; enforces freshness & replay protection.

## Packages

- pkg/crypto: Pedersen, BBS+ on BLS12-381.
- pkg/protocol: Bulletproofs, serialization, verification.
- pkg/revocation: RSA accumulator, witnesses.

## Security & Performance (Highlights)

- **Zero-knowledge:** proofs reveal no personal data.
- **Unlinkability:** re-randomization; no static identifiers.
- **Unforgeability:** BBS+ on BLS12-381.
- **Revocation privacy:** only revoked IDs in accumulator.

- Bind credentials to device secrets/biometrics for anti-sharing.
- Native mobile SDKs (Android, iOS).
- Threshold issuers and decentralized identifiers.
- Cross-authority scaling with common identifiers.

- Practical, privacy-preserving age checks without identity disclosure.
- Compact ZK composition: Pedersen + BBS+ + Bulletproofs + RSA accumulator.
- Efficient on BLS12-381 with a clean Go-based architecture.