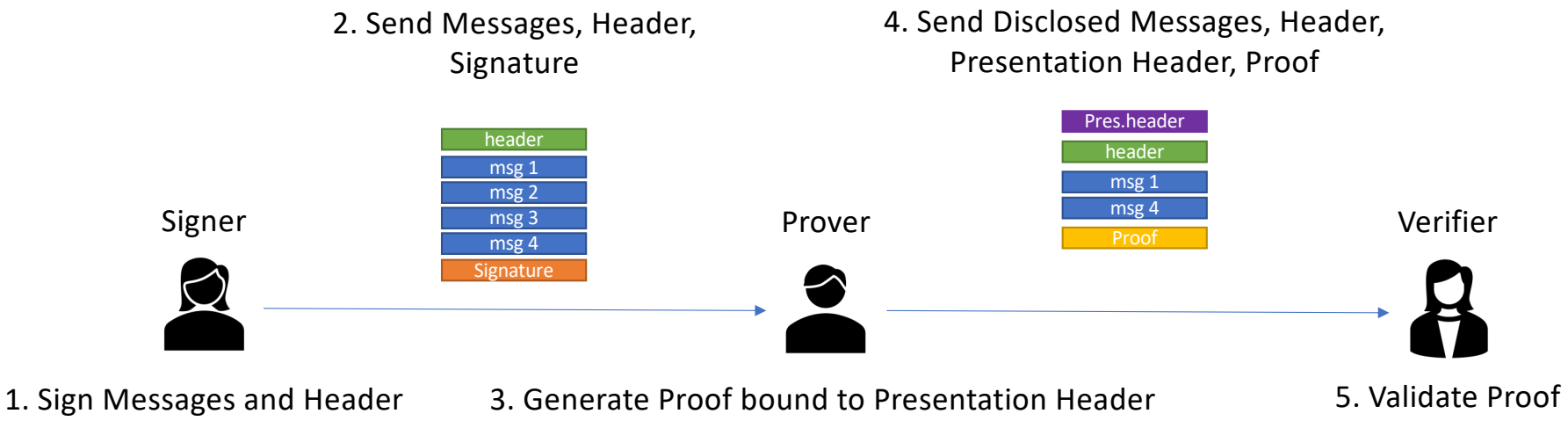
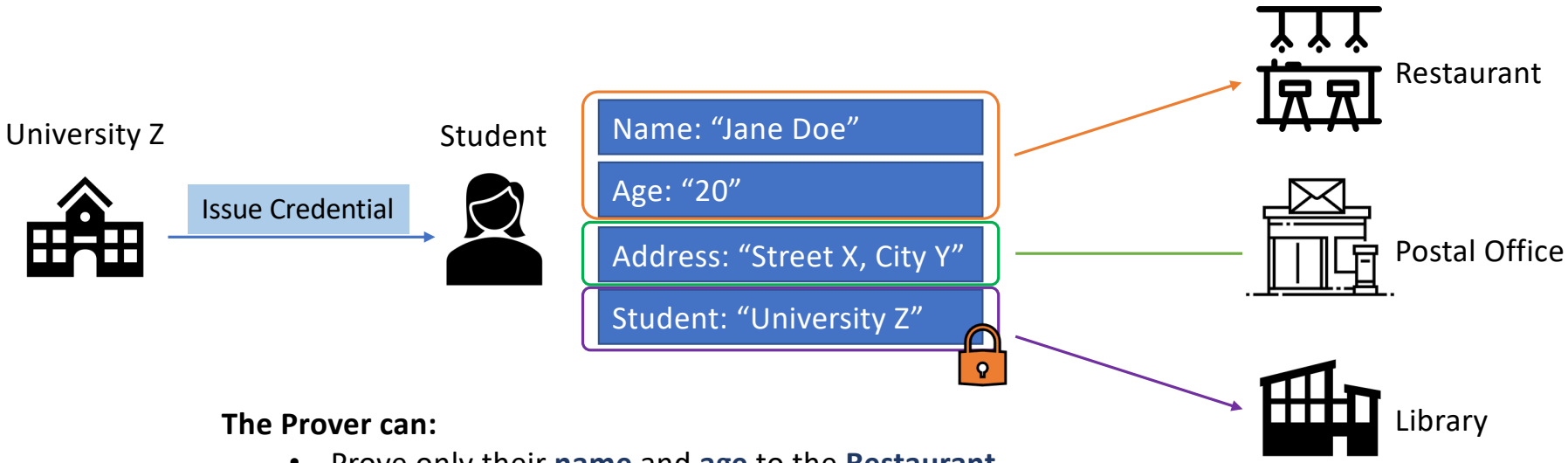
BBS Scheme


2. Send Messages, Header, Signature
4. Send Disclosed Messages, Header, Presentation Header, Proof
header
msg 1
msg 2
msg 3
msg 4
Signature
Pres.header
header
msg 1
msg 4
Proof
Signer
Prover
Verifier
1. Sign Messages and Header
3. Generate Proof bound to Presentation Header
5. Validate Proof

Privacy Preserving Anonymous Credentials



The Prover can:

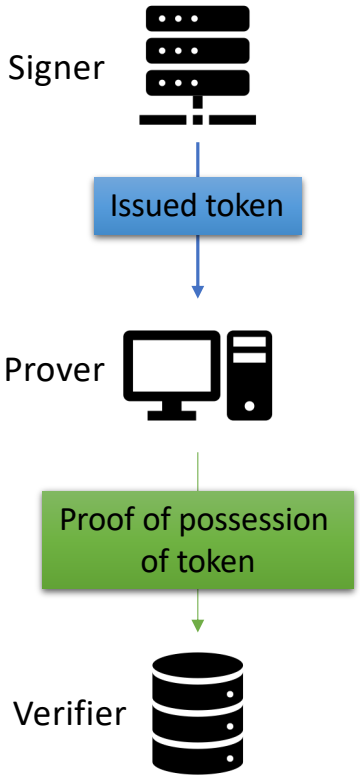
- Prove only their **name** and **age** to the **Restaurant**
- Prove only their **address** to the **Postal Office**.
- Prove only that they **are a student** to the **Library**.

➢ Only send the information that is relative to each Verifier

➢ The Verifiers cannot conspire to discover more information

(each proof is indistinguishable from random)

Proof of Possession enabled Security / Access Tokens



From the signers perspective:

- They can issue a single token that can be used multiple times by the prover.
- Does not require key material supplied by the prover ahead of time to issue.

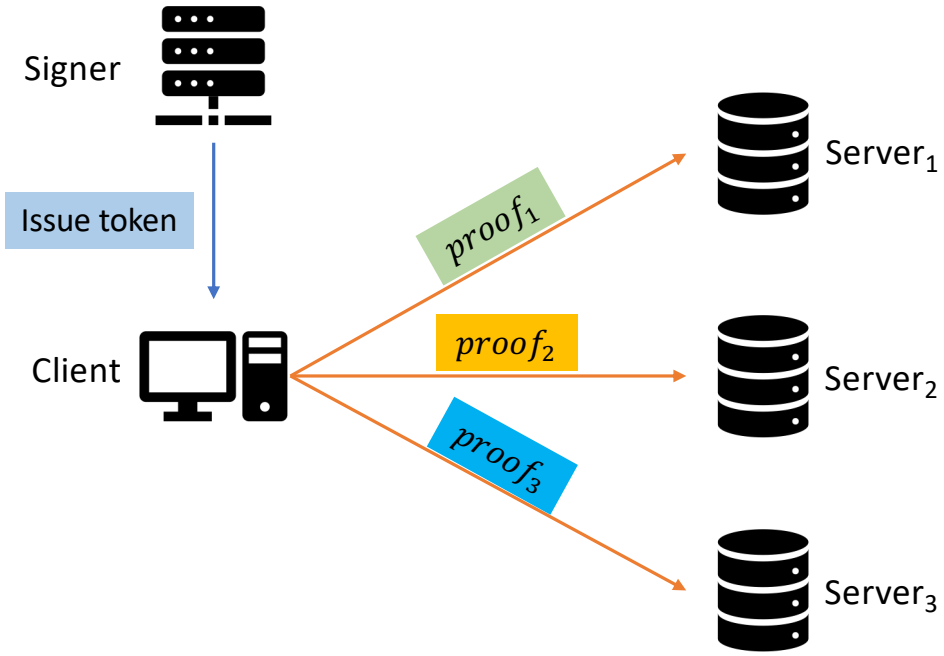
From the provers perspective:

- Can prover possession of the security token multiple times to different parties (verifiers).
- Does not require the prover to manage key material.
- Can scope generated proofs via the presentation header (e.g a generated proof is only valid for a particular verifier or has a TTL etc).

From the verifiers perspective:

- Validates the proof back to the original signer in a way that is inline with existing security tokens (e.g via the signers PK), also provides replay attack detection

Non-Correlating Security Token Proofs



During Proof Presentation:

- Each proof **cannot be correlated** to each other, the token or the client.
- Uncorrelatability holds even against **coalition between RPs** or **RPs and AS**.
- A unique **presentation header is NOT required** for un-correlatability to hold.