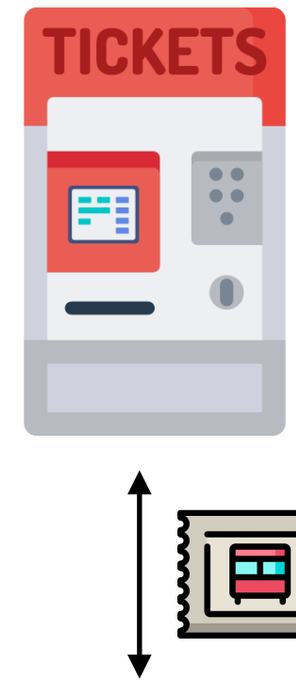
Non-Interactive Keyed-Verification Anonymous Credentials

- Geoffroy Couteau, Michael Reichle

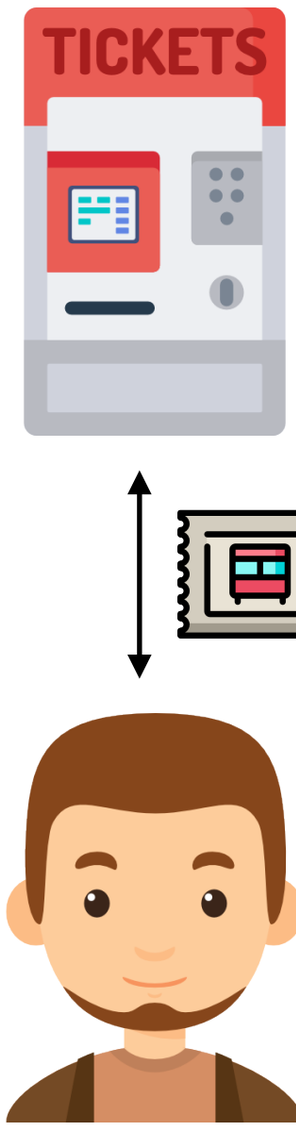
- ia.cr/2019/117

Anonymous Credentials

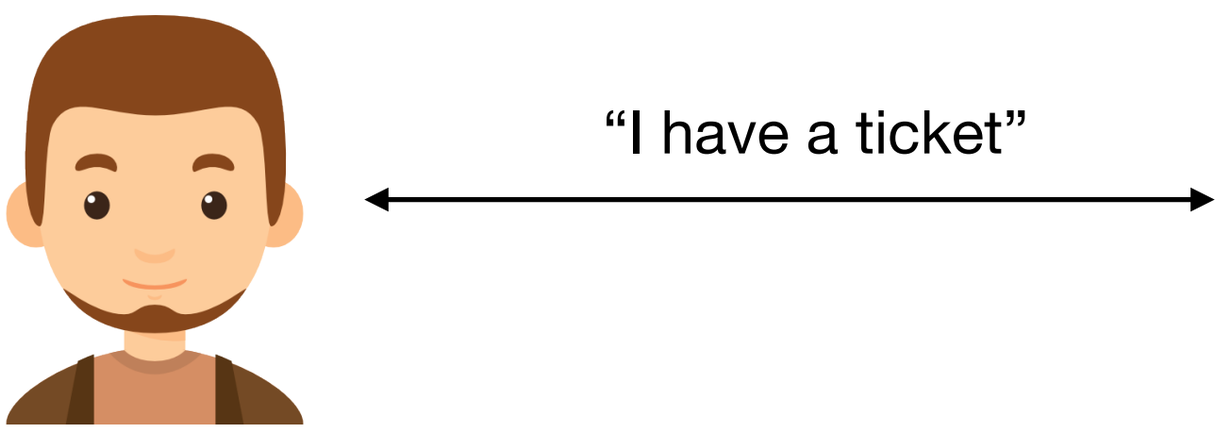


“I have a ticket”

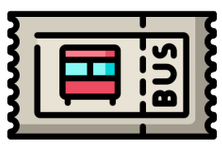
Anonymous Credentials

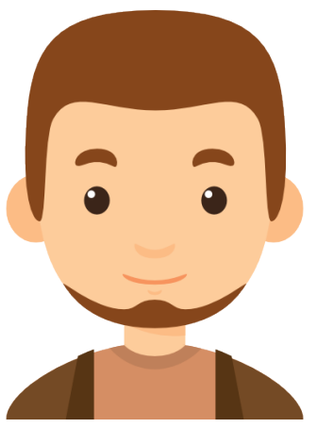


Anonymity:
users that show a credential should stay anonymous



Anonymous Credentials



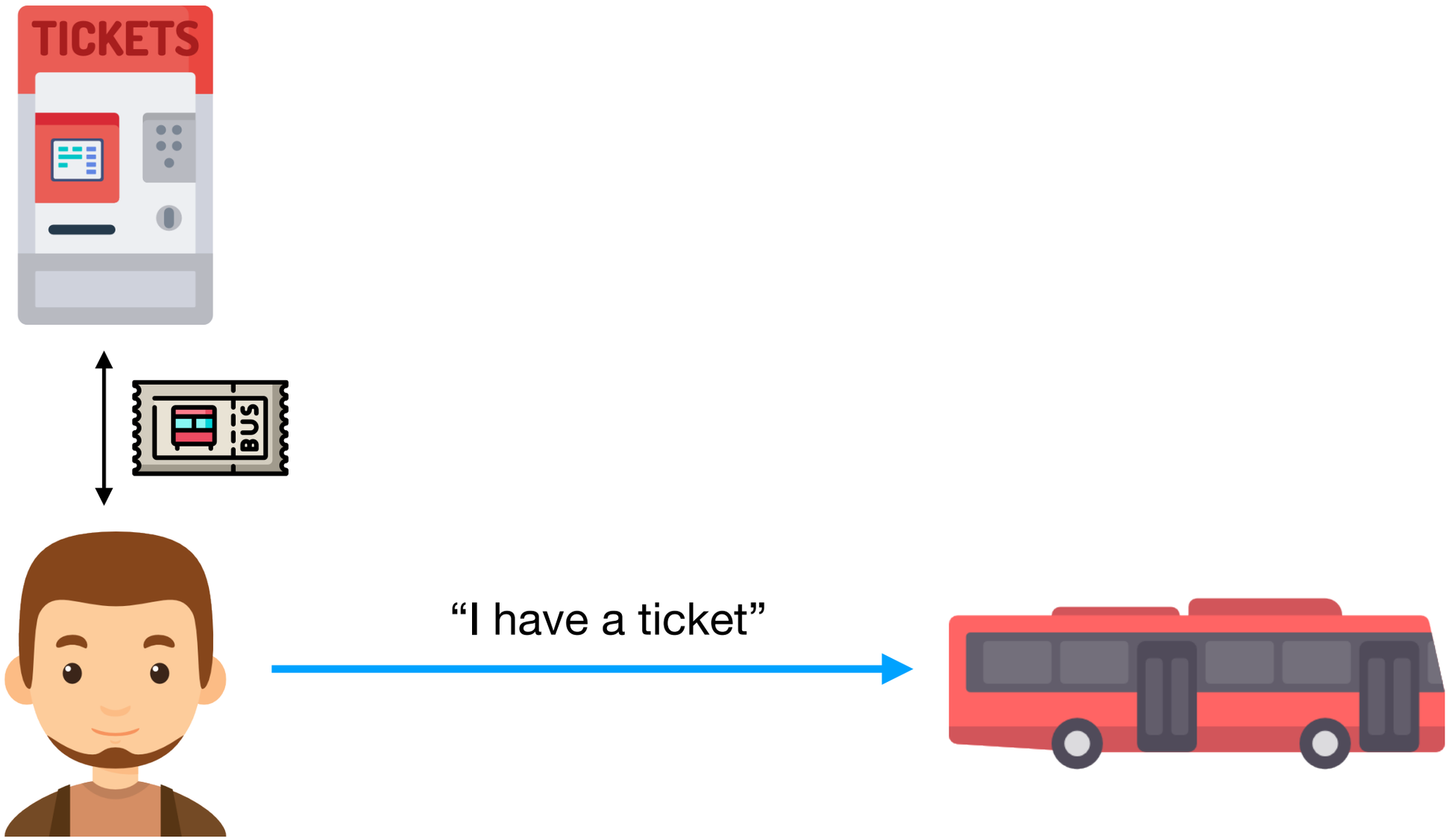


“I have a ticket”

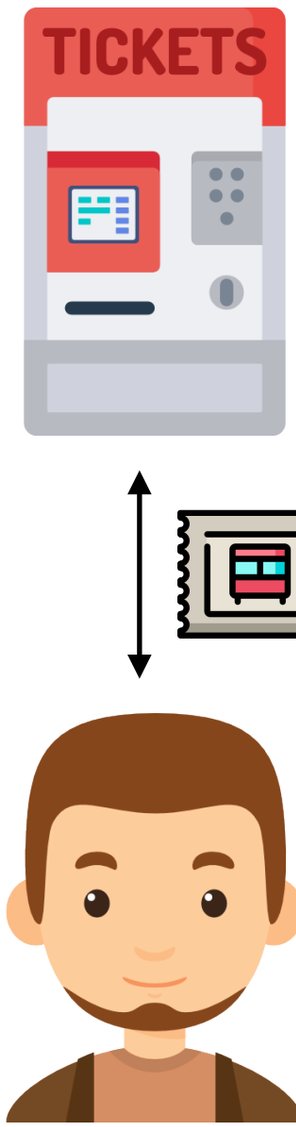
Anonymity:
users that show a credential should stay anonymous

Unforgeability:
users should not be able to forge a credentials

Non-Interactive Anonymous Credentials



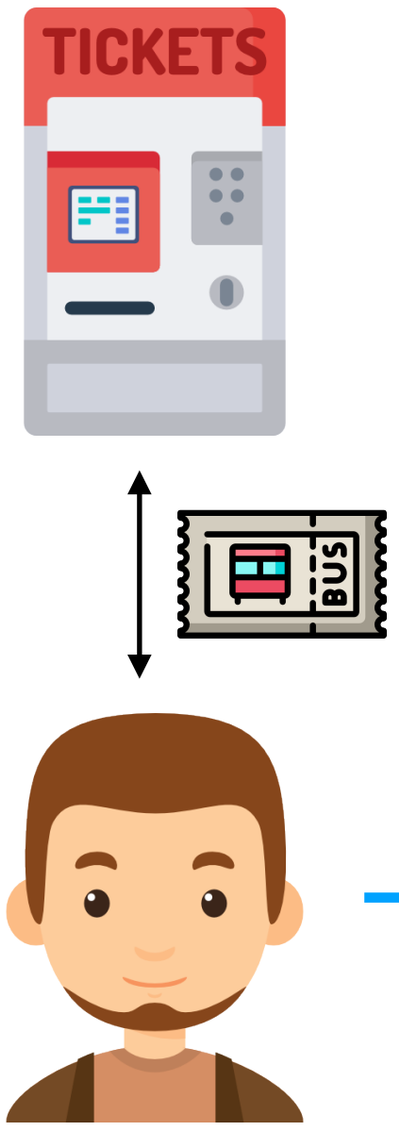
Non-Interactive Anonymous Credentials



Random Oracle Model:
only heuristic security guarantees

“I have a ticket”

Non-Interactive Anonymous Credentials



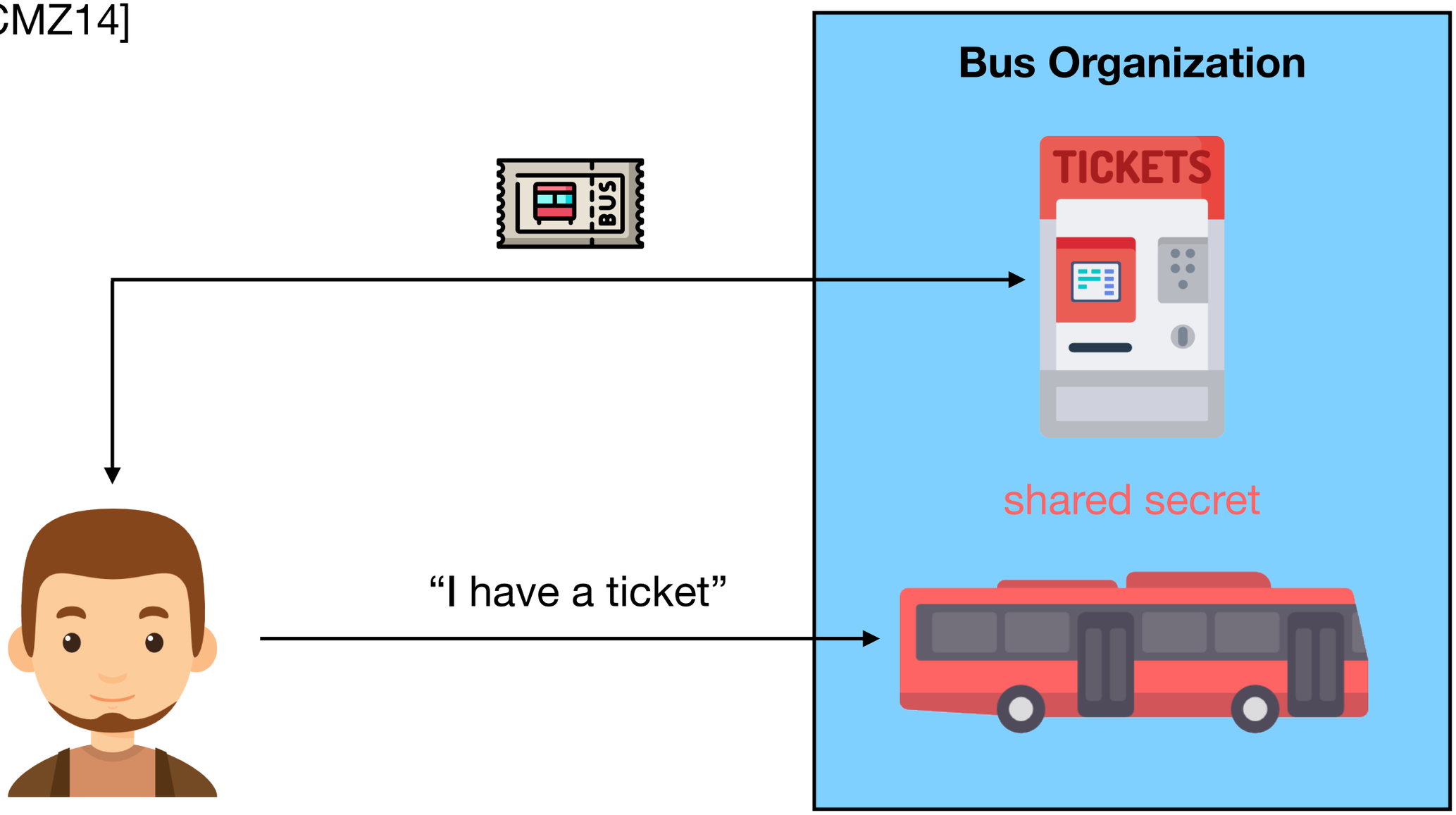
Random Oracle Model:
only heuristic security guarantees

Pairings:
specific assumptions and expensive computation

“I have a ticket”

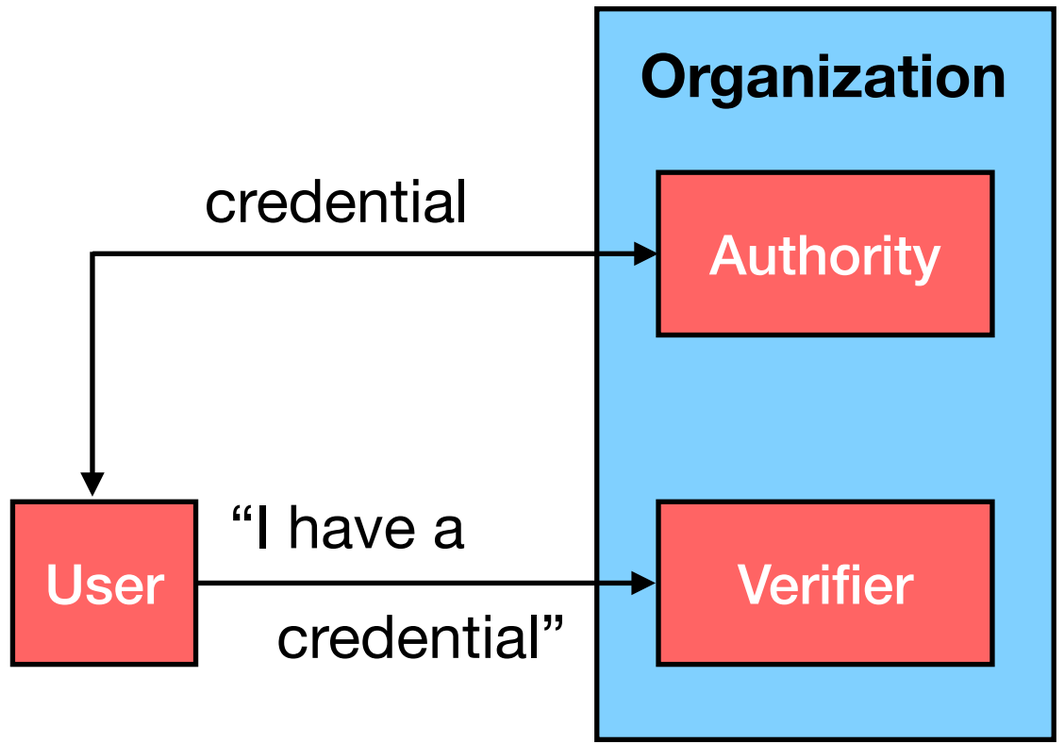
Keyed-Verification Anonymous Credentials

[CMZ14]

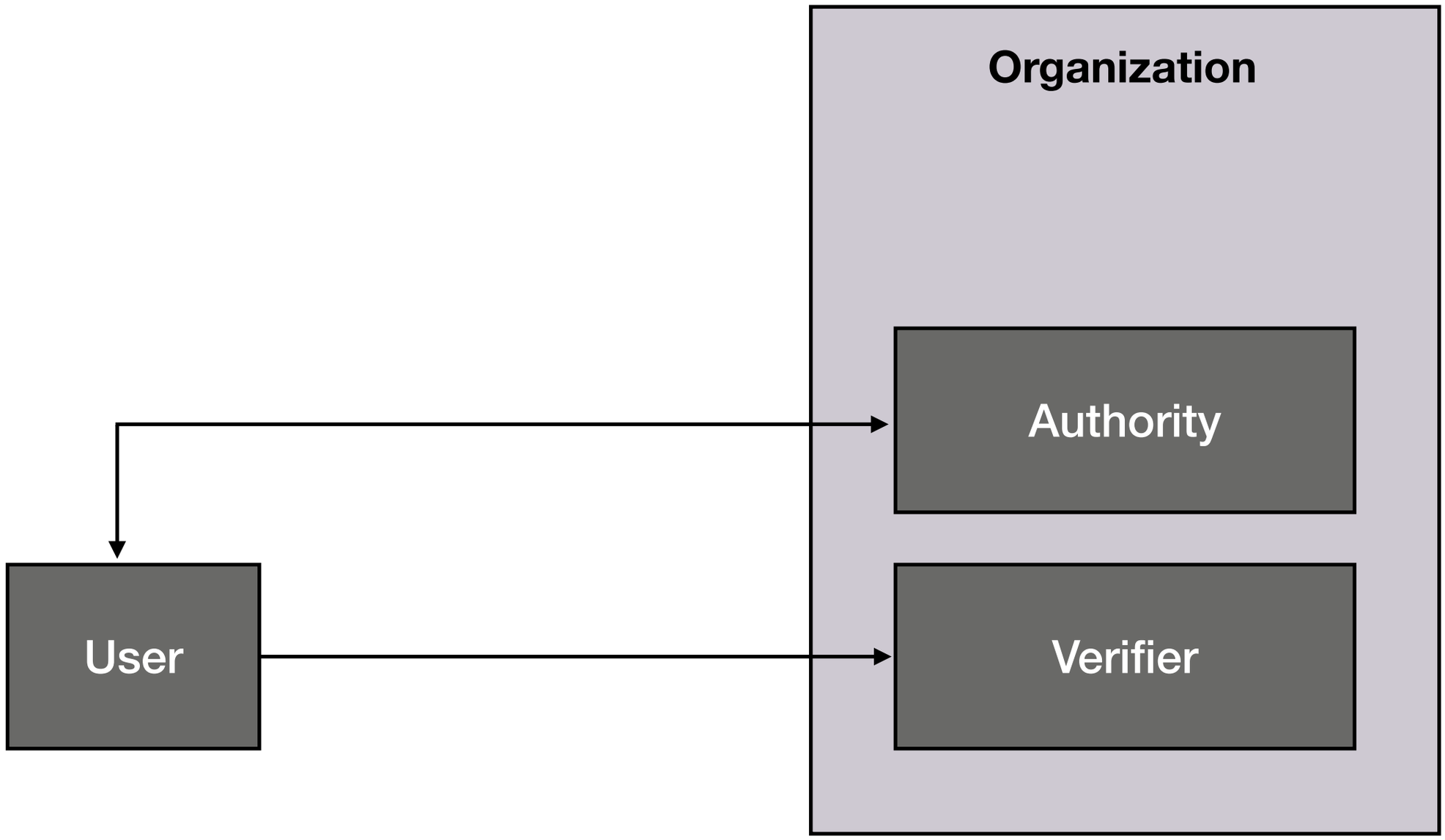


Results

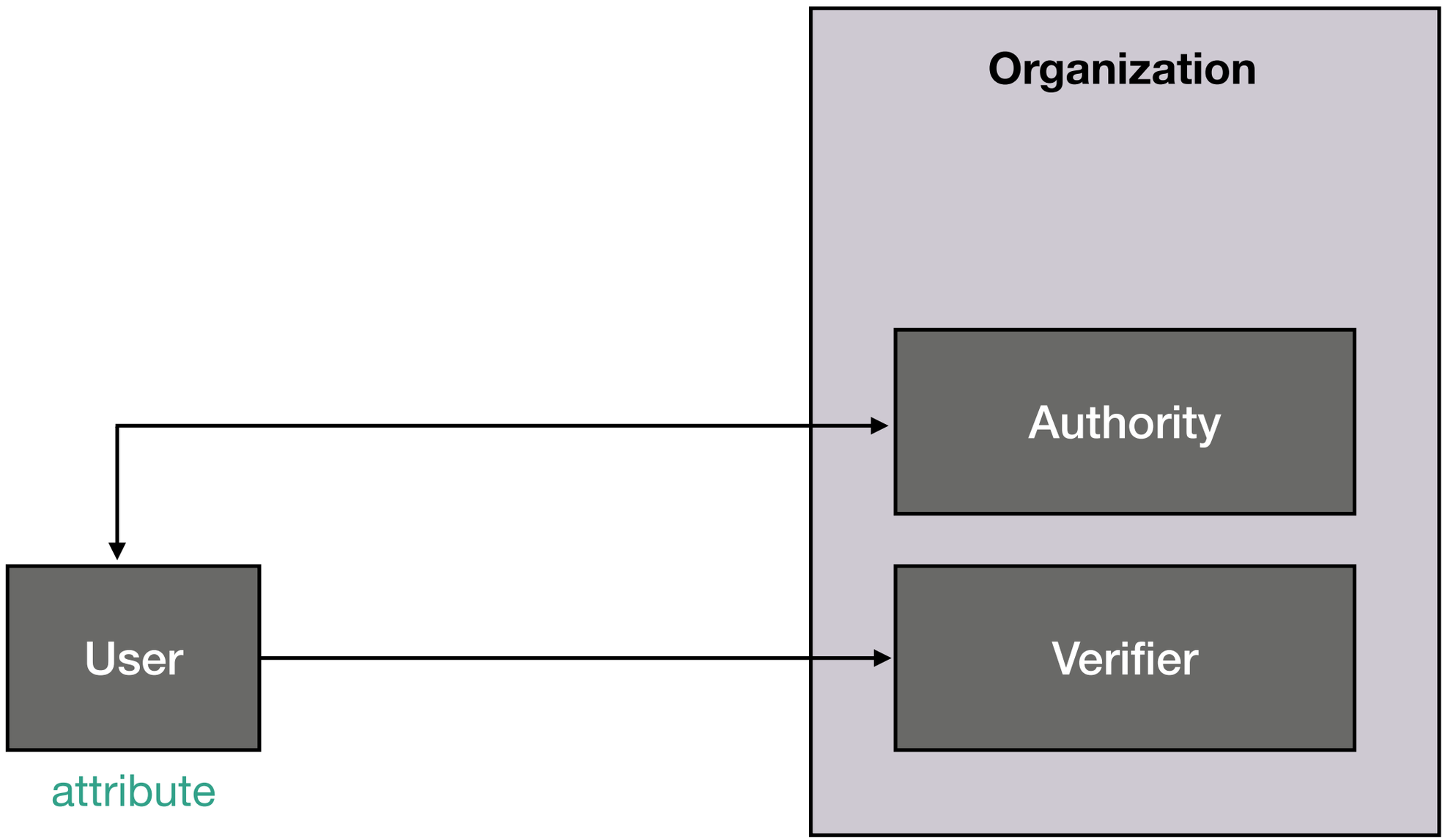
- secure in standard model
- no pairings or ROM
- short proof of possession
- standard properties (and more!)



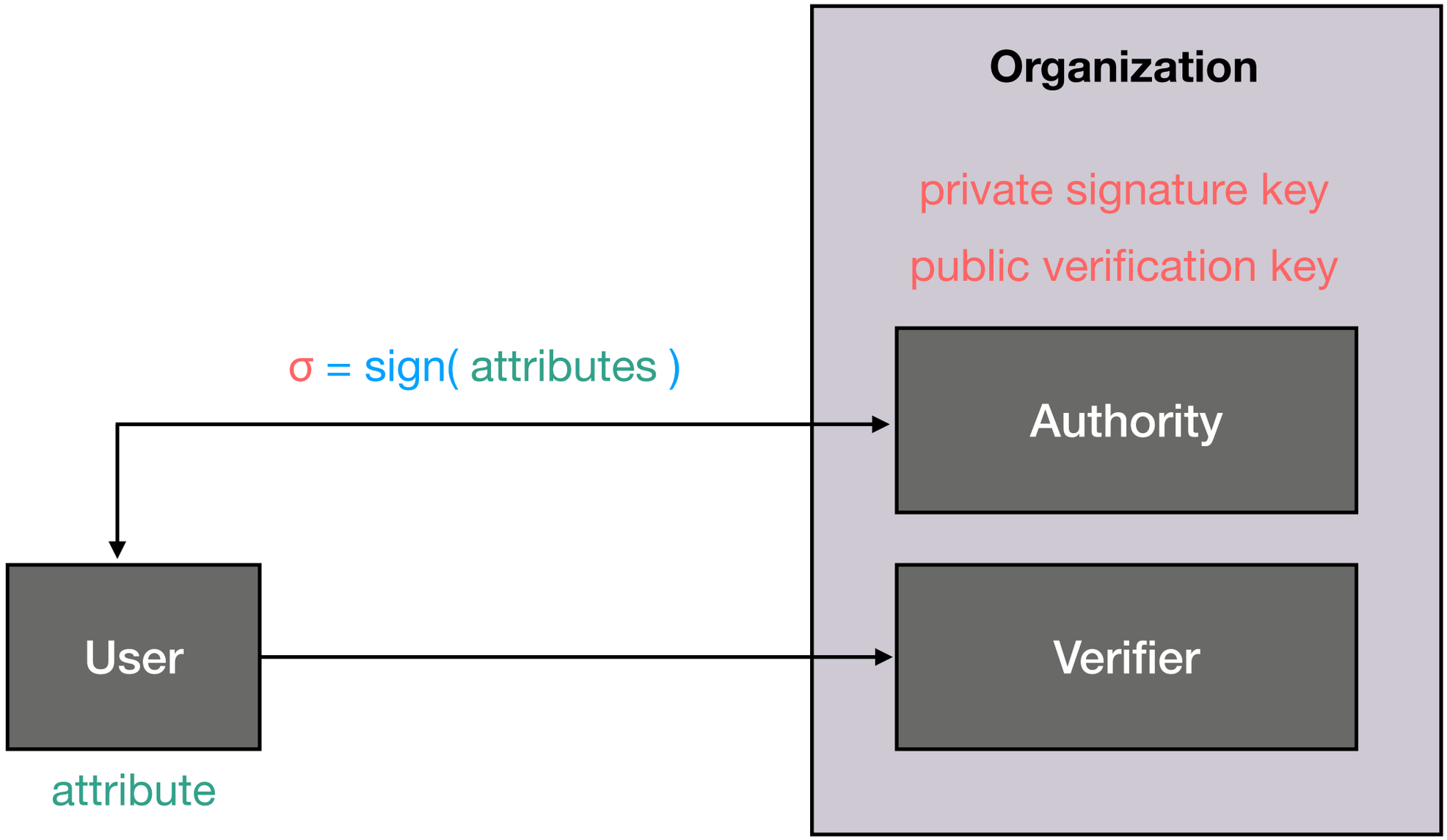
Construction



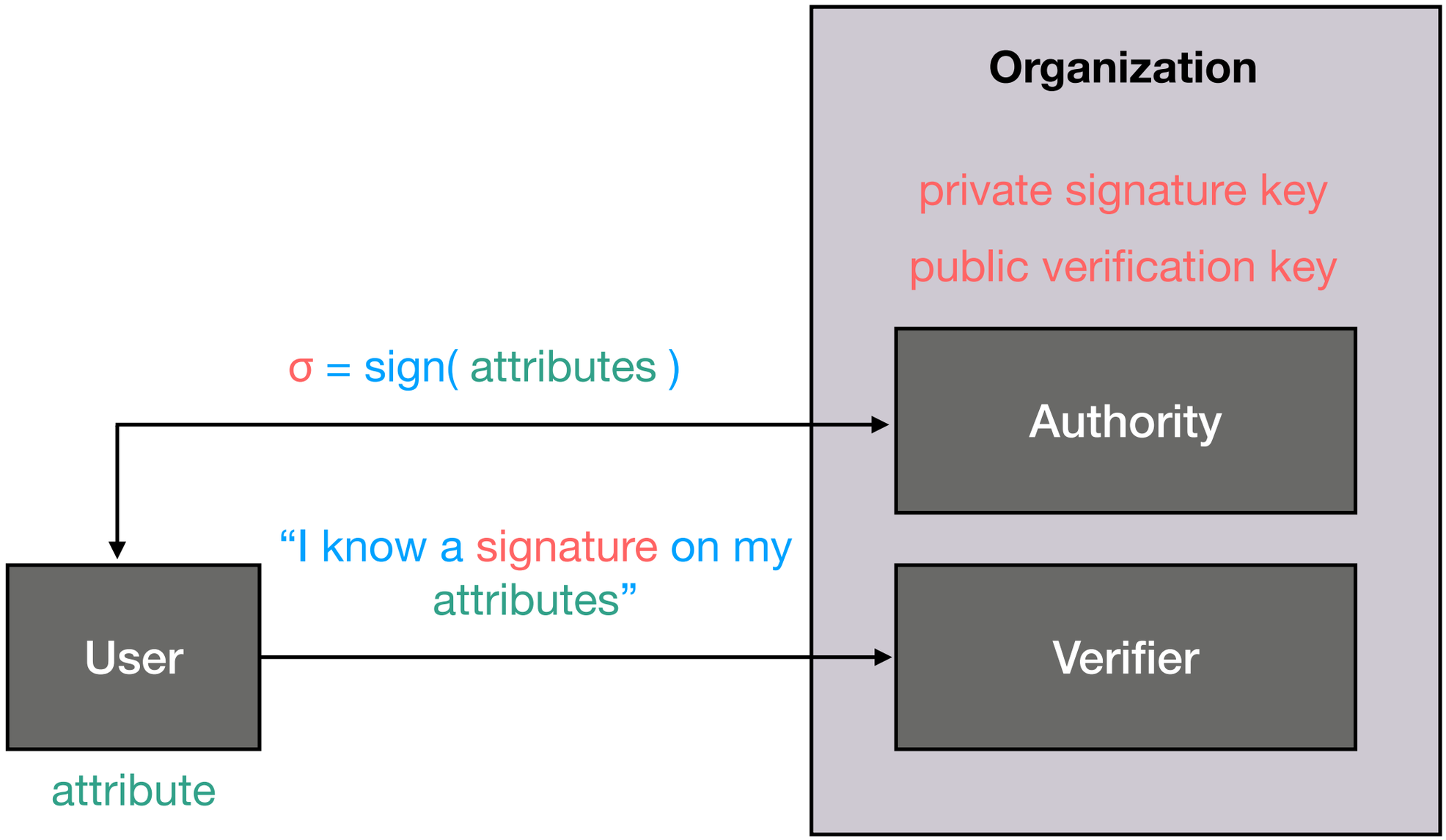
Construction



Construction



Construction



Construction

Zero-Knowledge Proof:

- *attributes stay hidden* *(Zero-Knowledge)*
- *ensures proof is correct* *(Soundness)*

Organization

private signature key

public verification key

σ = sign(attributes)

Authority

“I know a signature on my attributes”

User

attribute

Verifier

Construction

Zero-Knowledge Proof:

- *attributes stay hidden* *(Zero-Knowledge)*
- *ensures proof is correct* *(Soundness)*

Organization

private signature key

public verification key

DVNIZK verification key

σ = sign(attributes)

Authority

User

attribute

[CC18]

Verifier

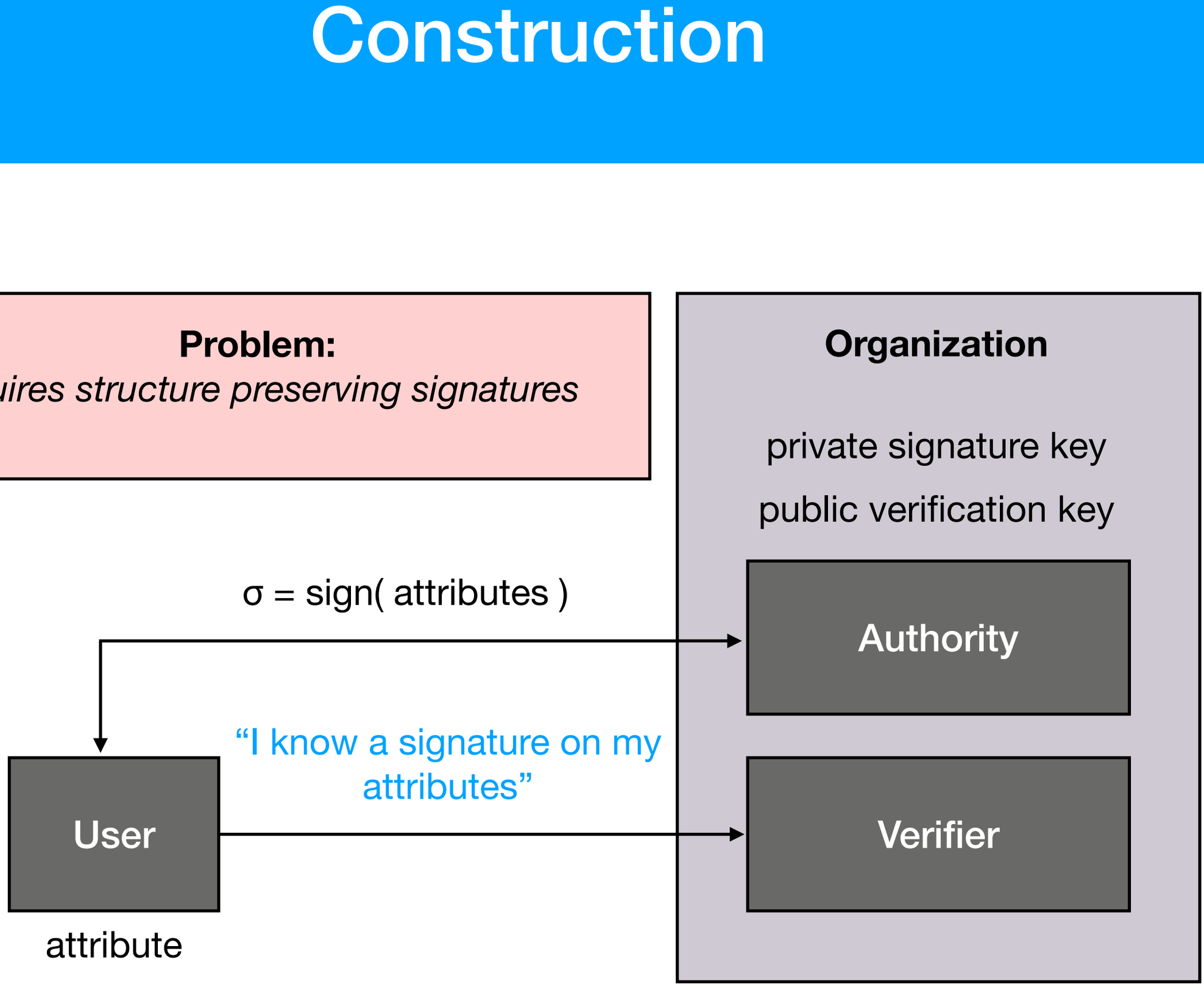
Construction

Problem:
requires structure preserving signatures

Organization

private signature key

public verification key



Construction

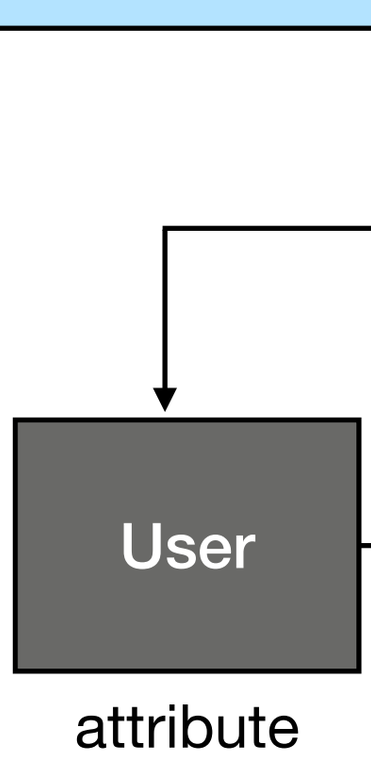
Algebraic MAC

- *message and key are exponents*
- *MAC is a tuple of group elements*

Organization

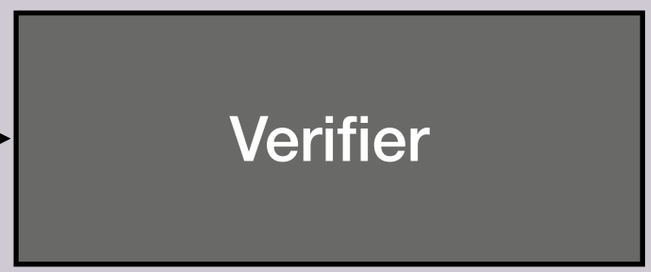
private MAC key

σ = MAC(attributes)



“I know a MAC on my attributes”

attribute

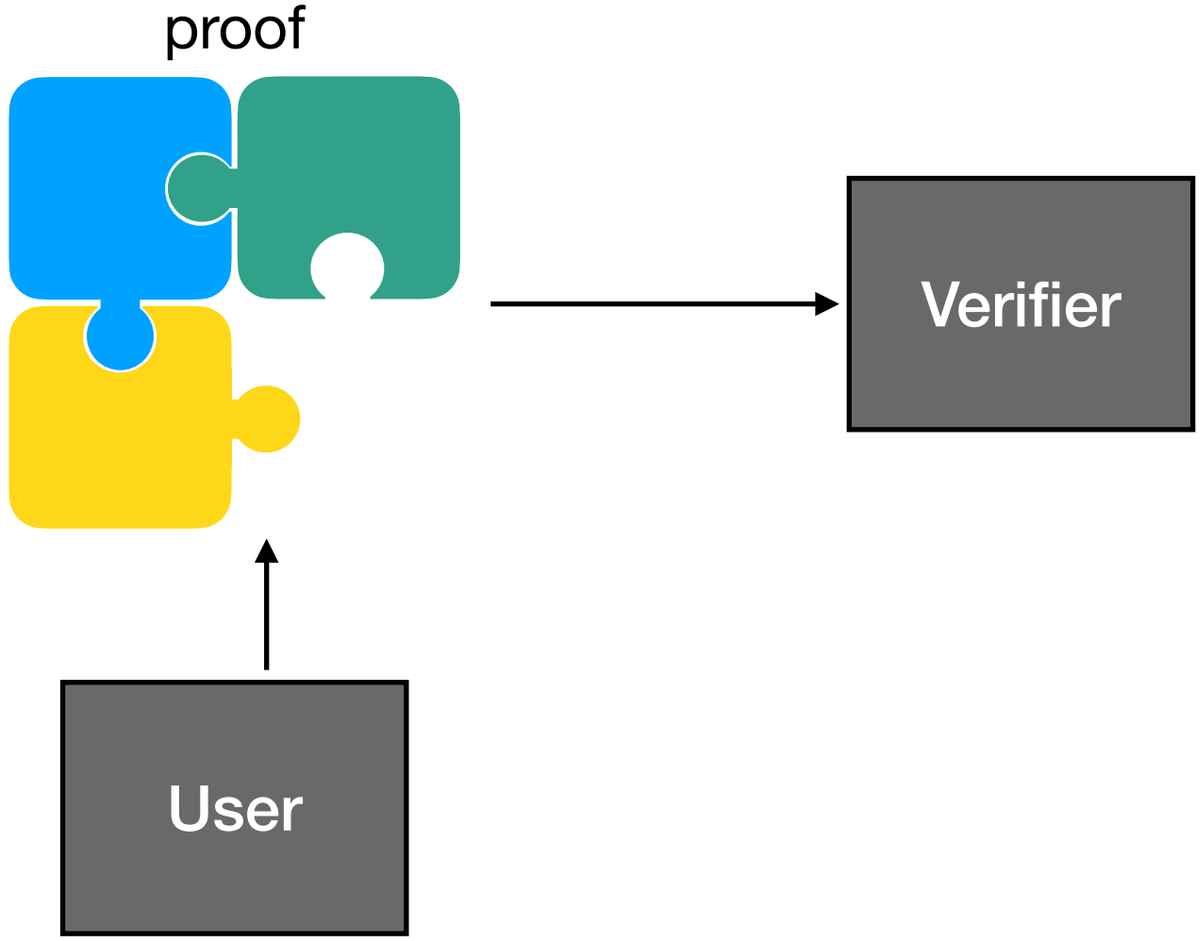


Oblivious Proofs

“I know a MAC on my attributes”

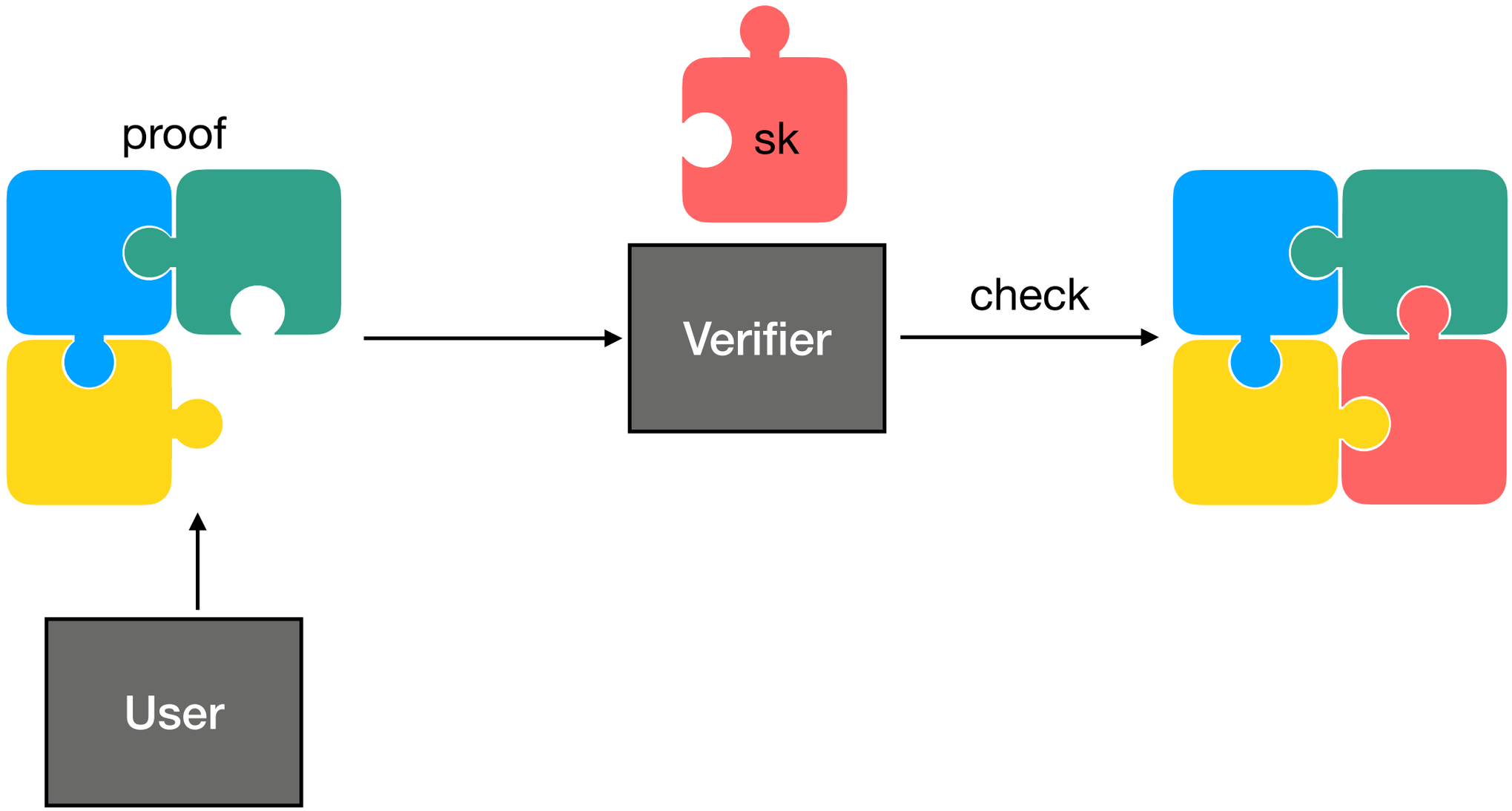
Oblivious Proofs

"I know a MAC on my attributes"



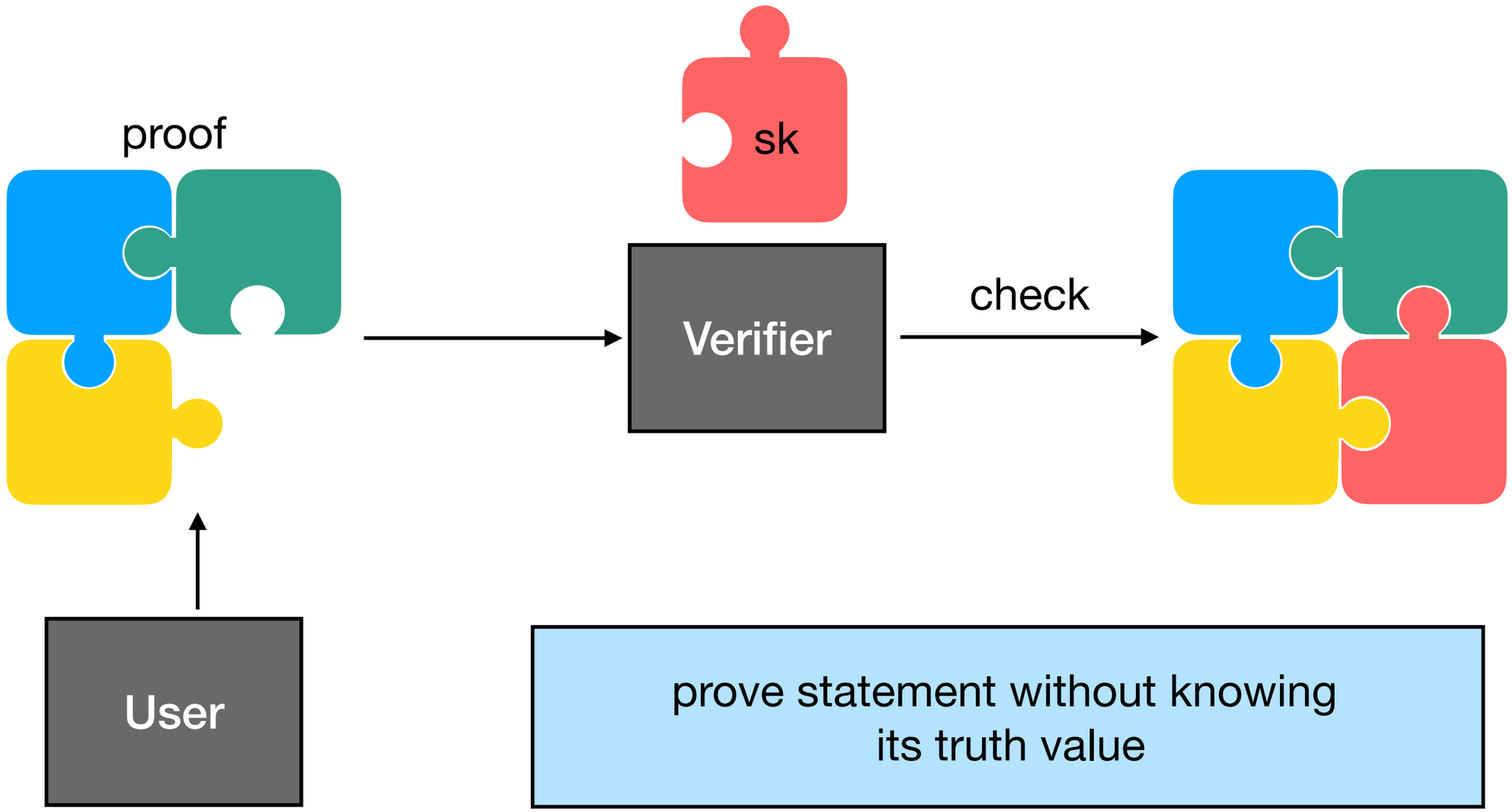
Oblivious Proofs

"I know a MAC on my attributes"



Oblivious Proofs

"I know a MAC on my attributes"



Oblivious Proofs

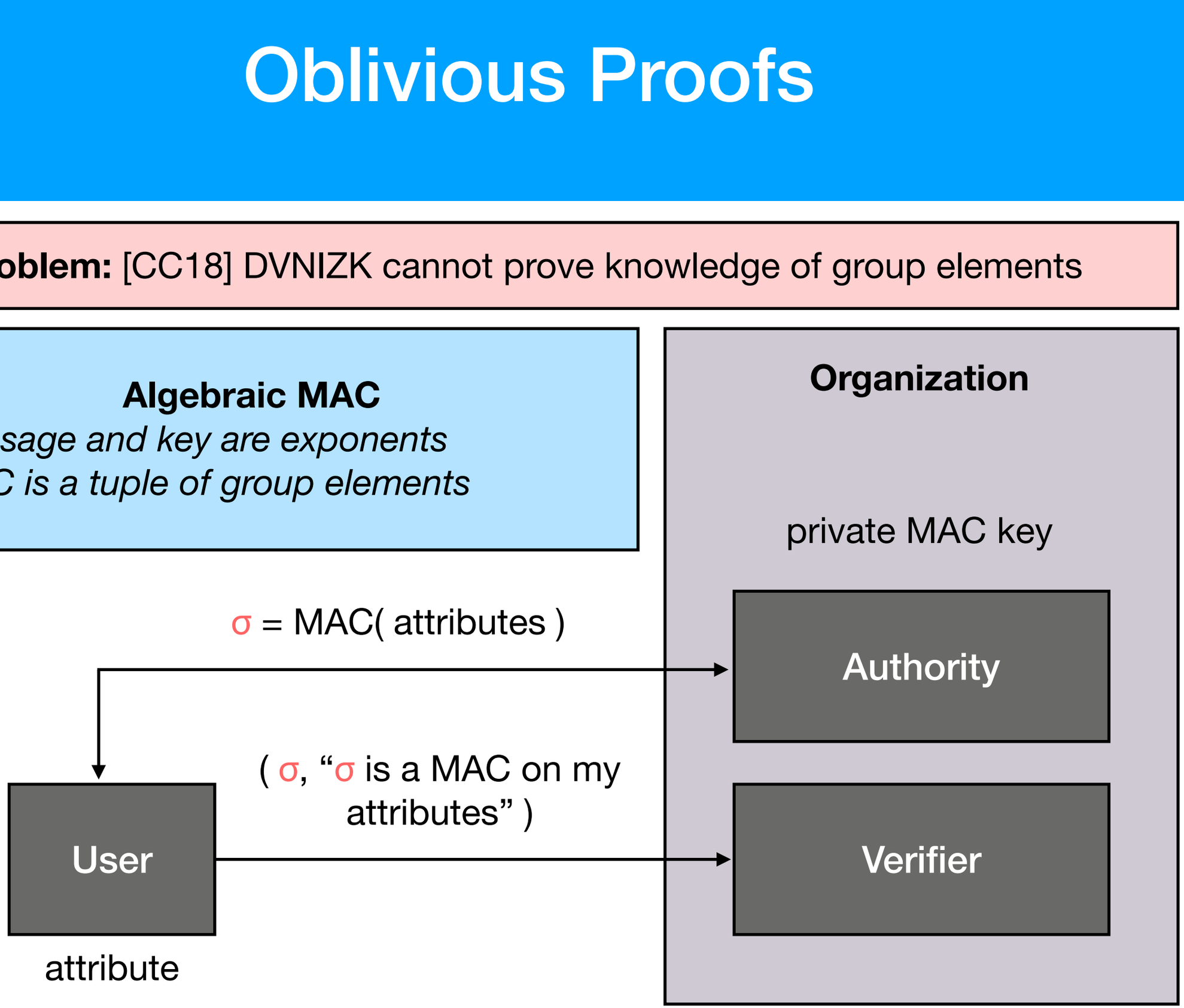
Problem: [CC18] DVNIZK cannot prove knowledge of group elements

Algebraic MAC

- *message and key are exponents*
- *MAC is a tuple of group elements*

Organization

private MAC key



Oblivious Proofs

Last Step:
randomise sent MAC

Organization

private MAC key

Authority

Verifier

σ = MAC(attributes)

(σ*, “σ* is a MAC on my attributes”)

User

attribute

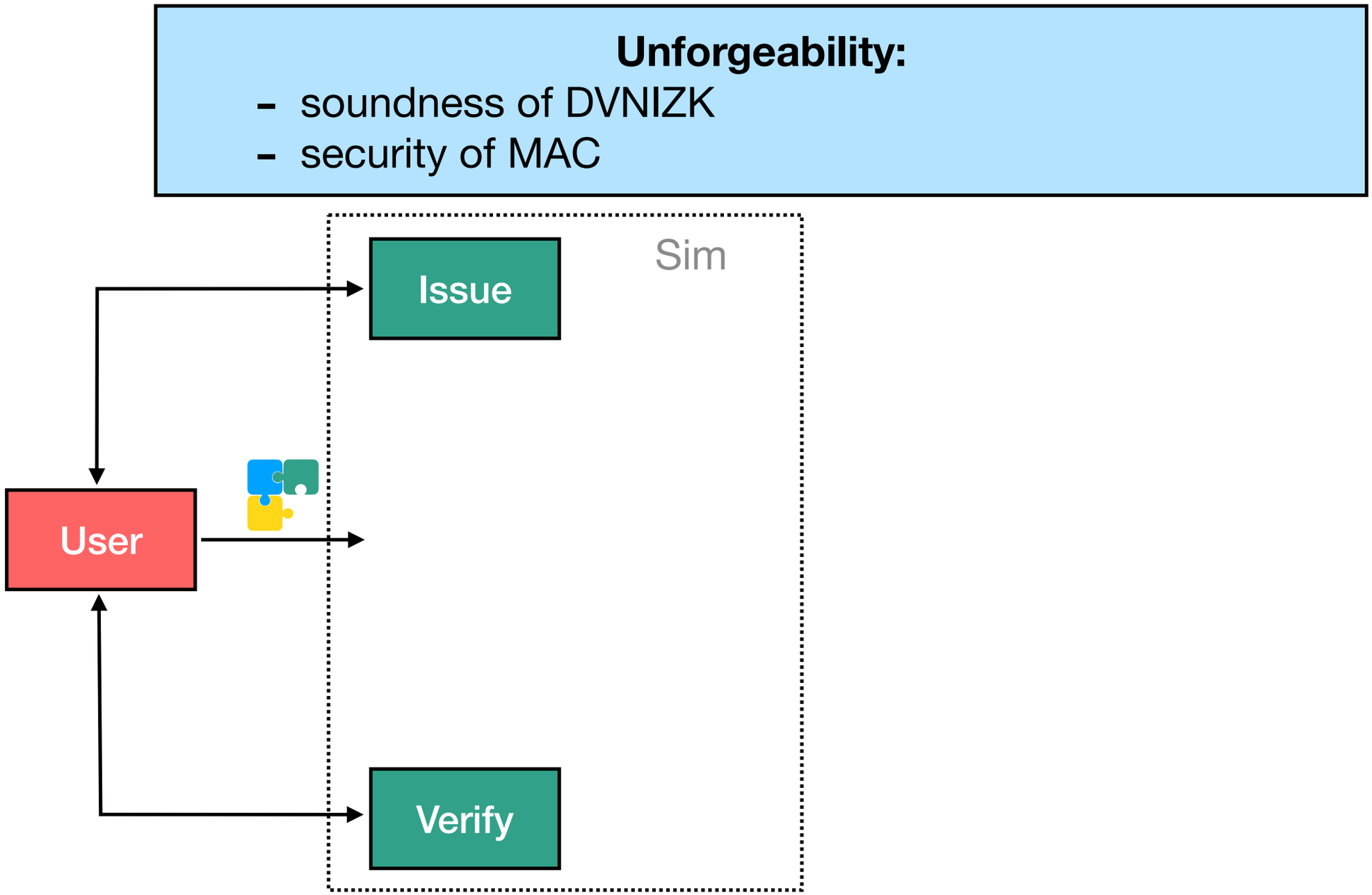
Proving Security

Anonymity:
- randomisation and zero-knowledge of DVNIZK

Proving Security

Unforgeability:

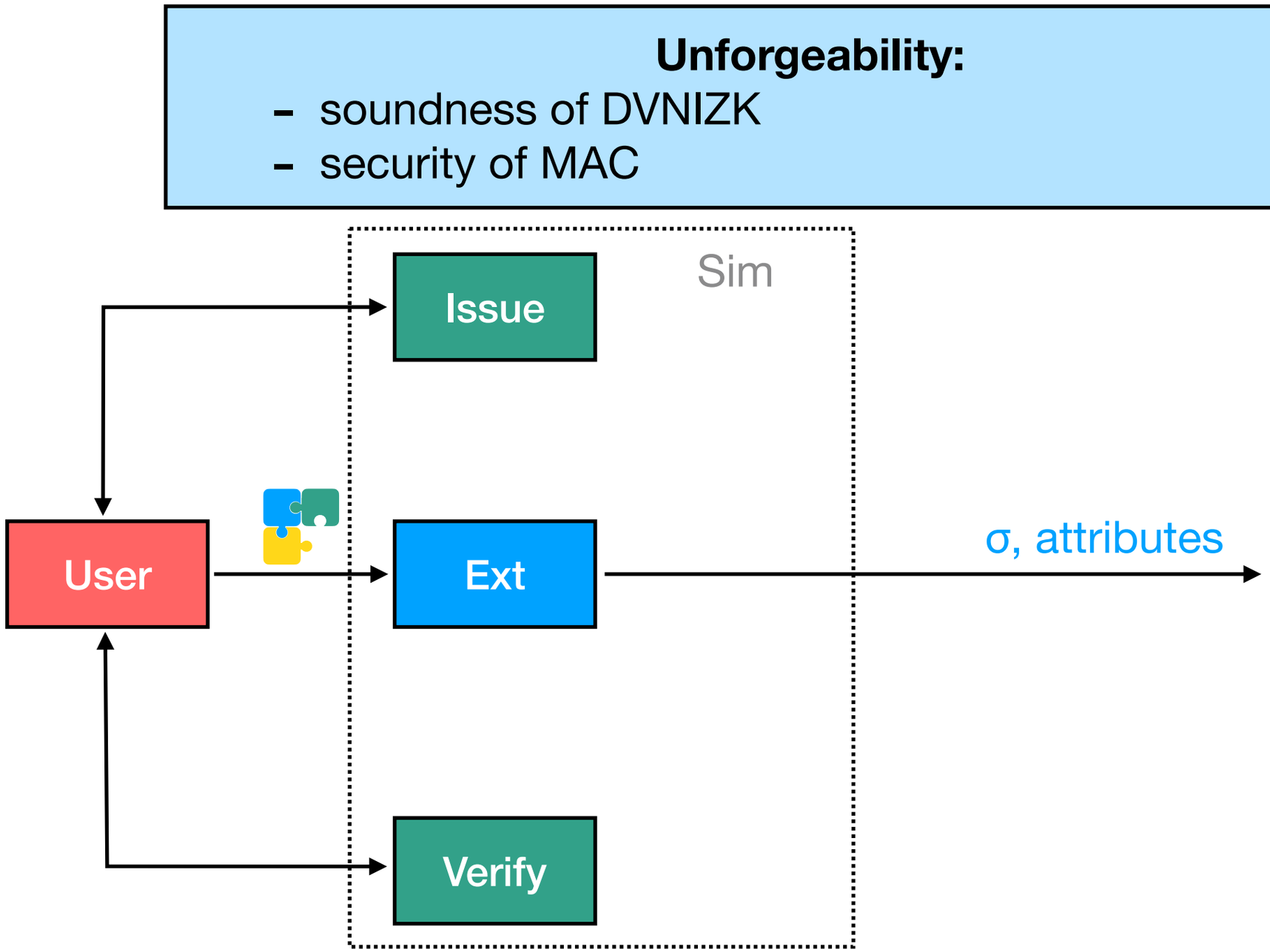
- soundness of DVNIZK
- security of MAC



Proving Security

Unforgeability:

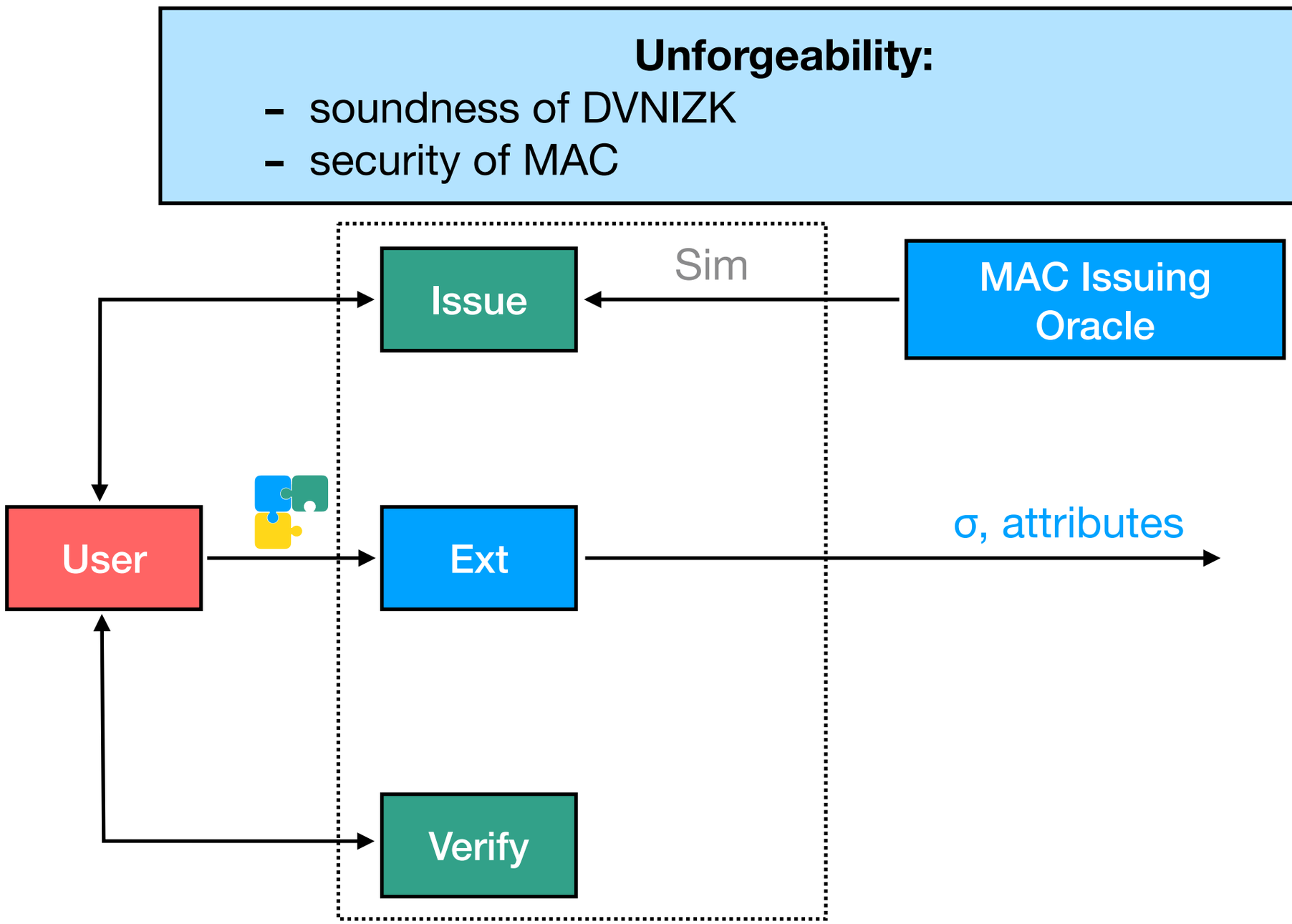
- soundness of DVNIZK
- security of MAC



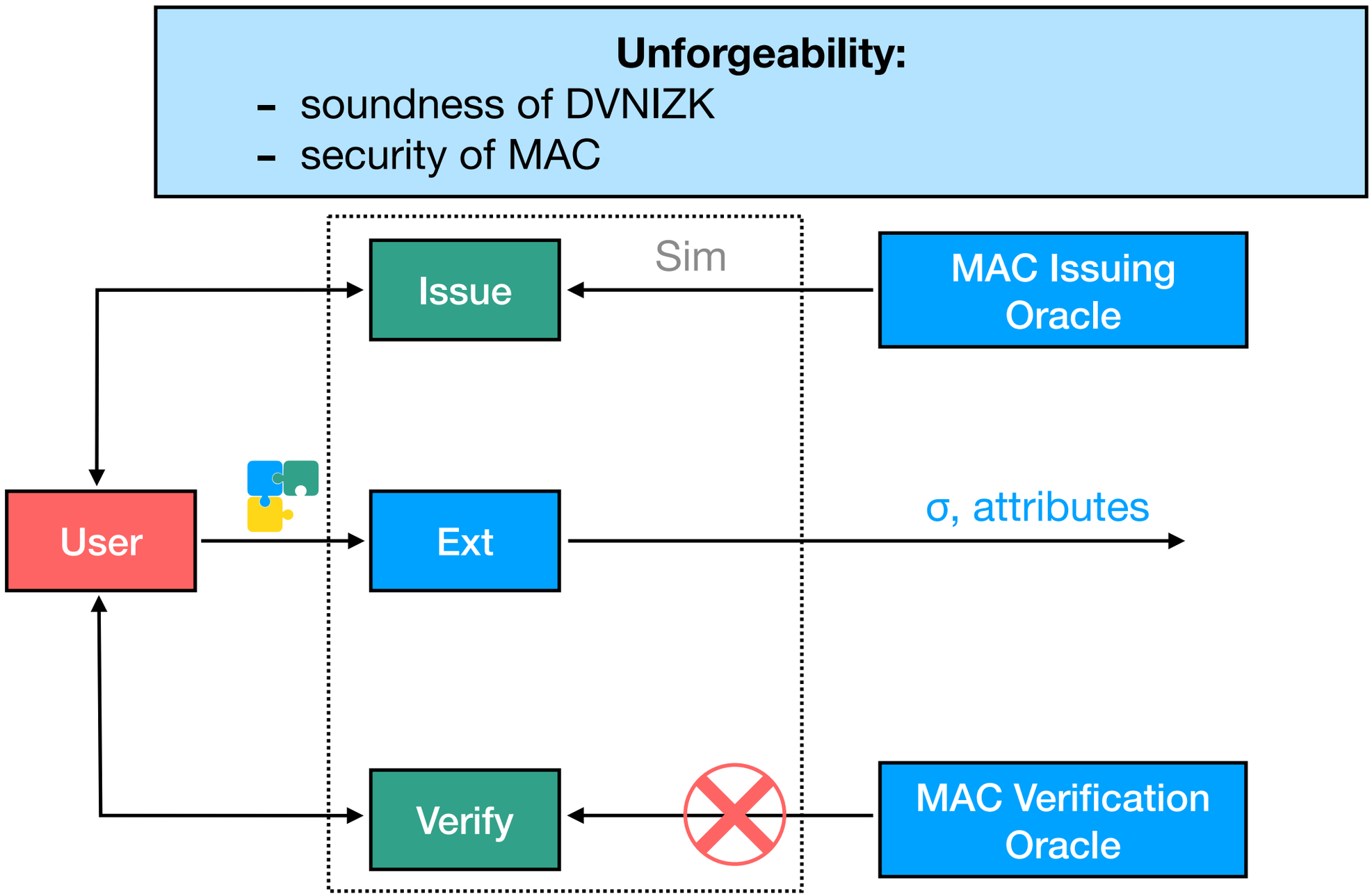
Proving Security

Unforgeability:

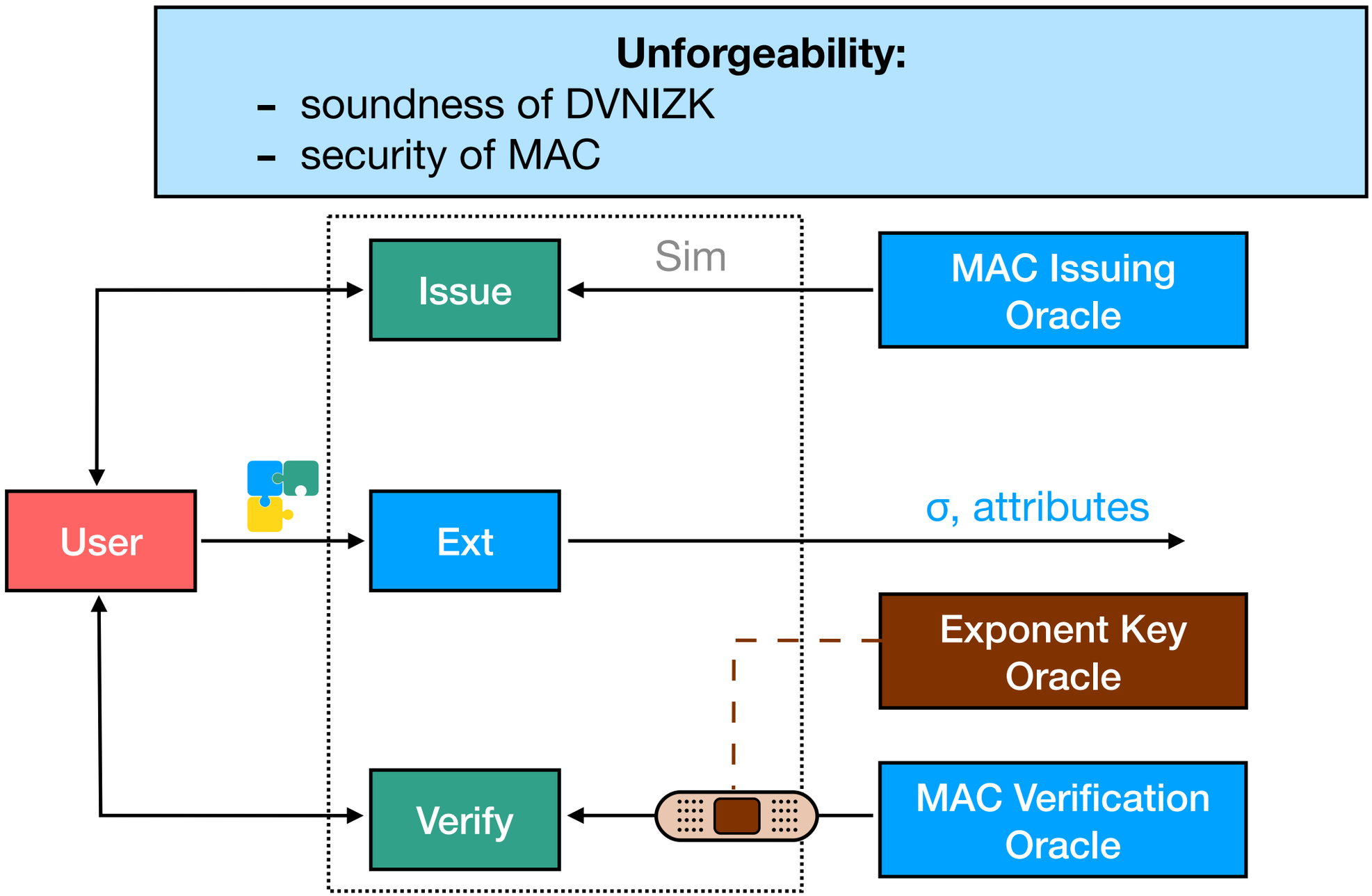
- soundness of DVNIZK
- security of MAC



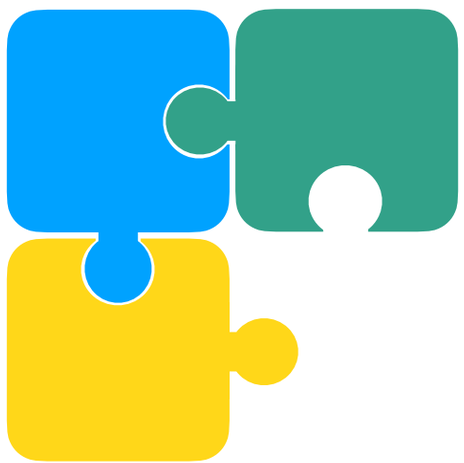
Proving Security



Proving Security



Proving Security

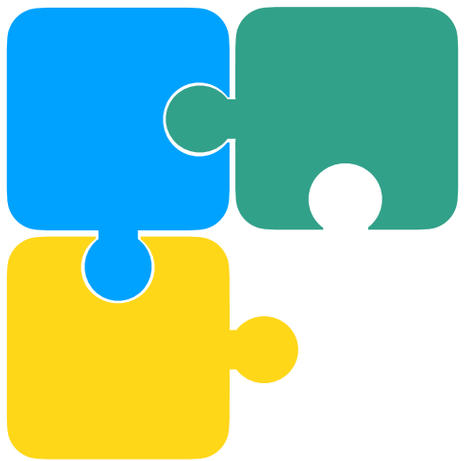


Unforgeability:

- soundness of DVNIZK
- security of MAC
- additional MAC key oracle

→ extract forgery from proof

Proving Security



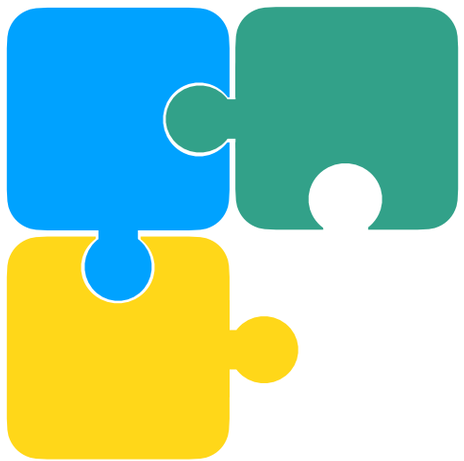
Unforgeability:

- soundness of DVNIZK
- security of MAC
- additional MAC key oracle

} Extended Unforgeability

→ extract forgery from proof

Proving Security



Unforgeability:

- soundness of DVNIZK
- security of MAC
- additional MAC key oracle

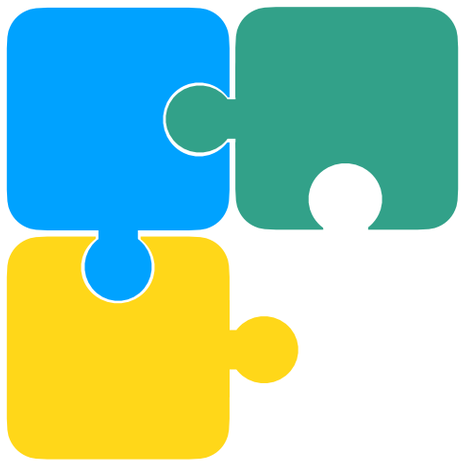
} Extended Unforgeability

→ extract forgery from proof

Good News:

- proof method works with attributes of size 1 without additional oracle

Proving Security



Unforgeability:

- soundness of DVNIZK
- security of MAC
- additional MAC key oracle

} Extended Unforgeability

→ extract forgery from proof

Good News:

- proof method works with attributes of size 1 without additional oracle

MAC_{GGM}:

- prove extended unforgeability for MAC_{GGM} in generic group model
- require additional assumption because of composite order groups

Additional Properties

Proof of Credential

- contains *ExtCom(attributes, random)*

Additional Properties

Proof of Credential
- contains *ExtCom(attributes, random)*

Anonymity Revocation:
secret key allows extracting the attributes

Additional Properties

Proof of Credential

- contains *ExtCom(attributes, random)*

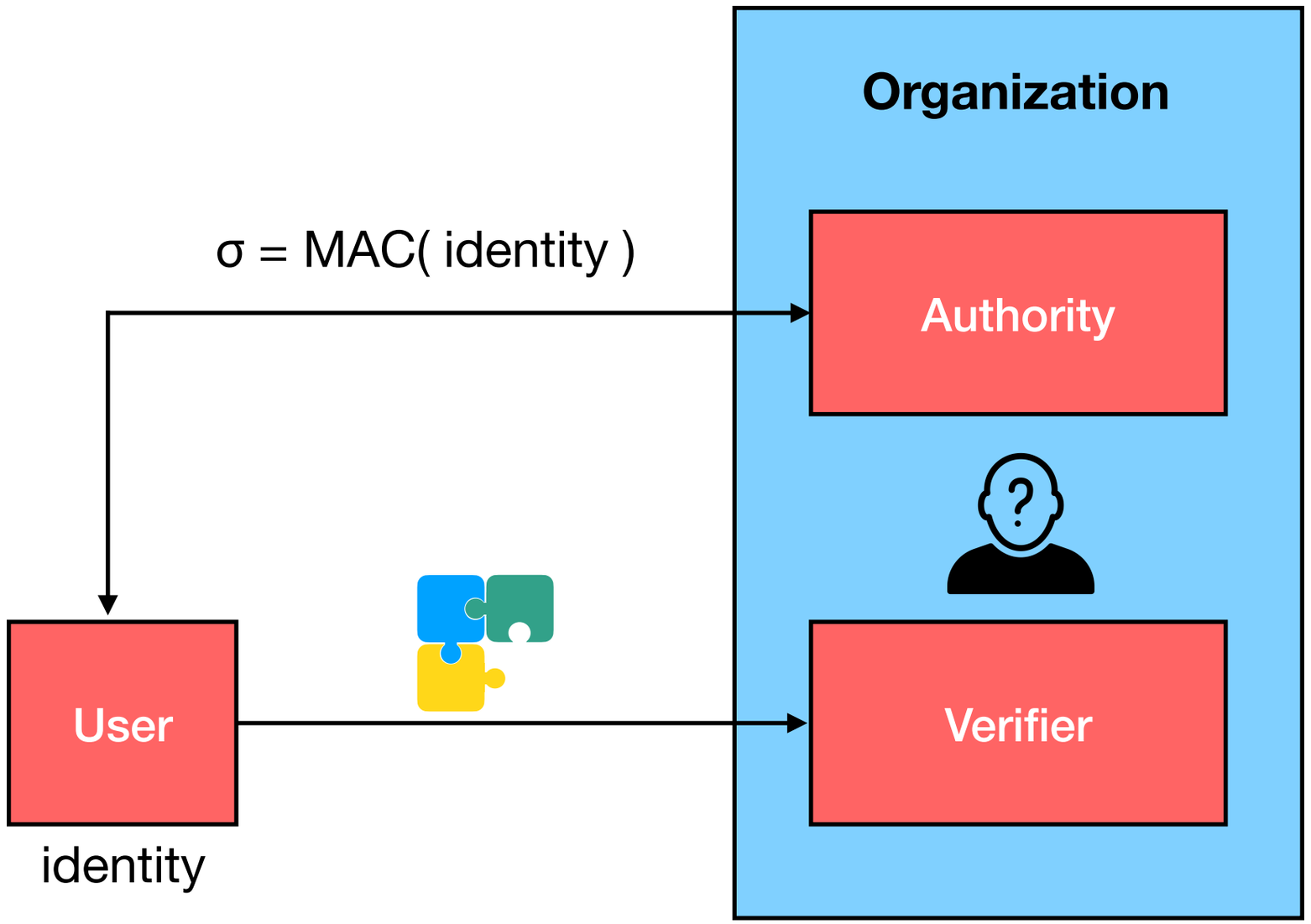
Anonymity Revocation:

secret key allows extracting the attributes

Cheap Pseudonyms:

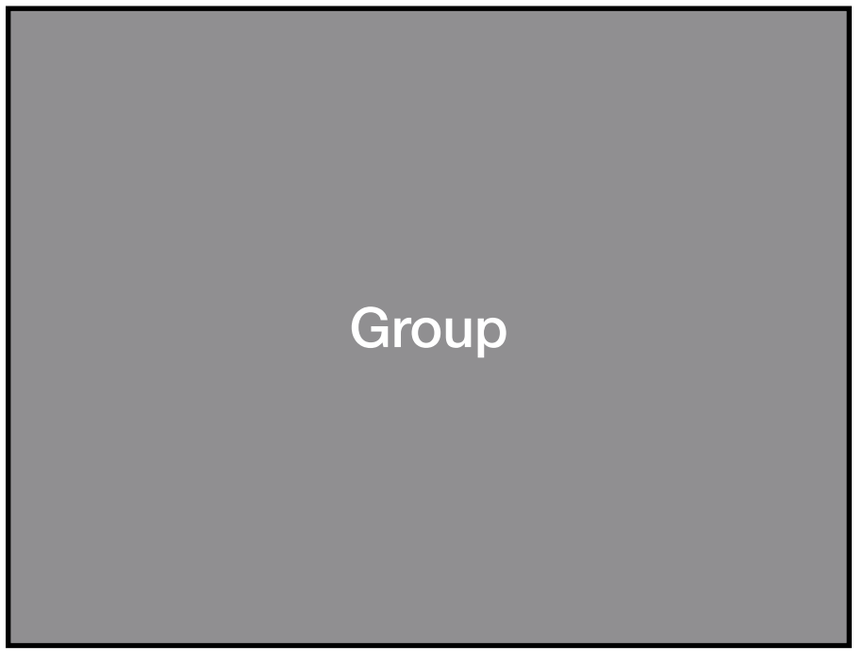
no additional commitments necessary for pseudonyms

Setting for Efficiency Analysis

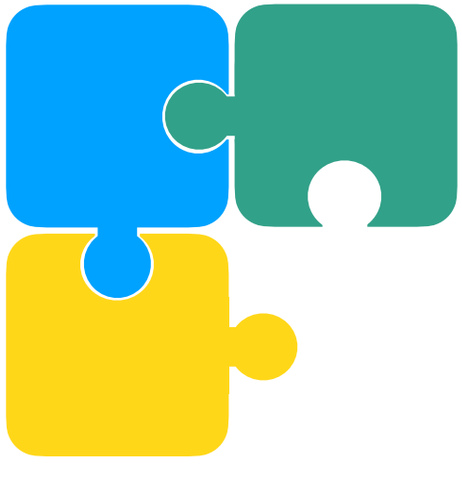


Efficiency

This Work

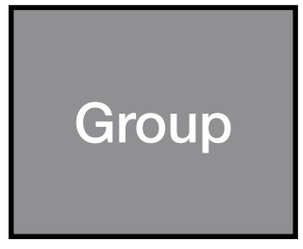


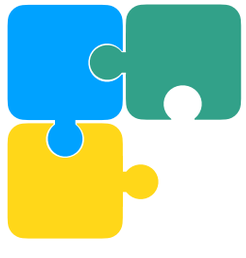
8× larger



3× larger

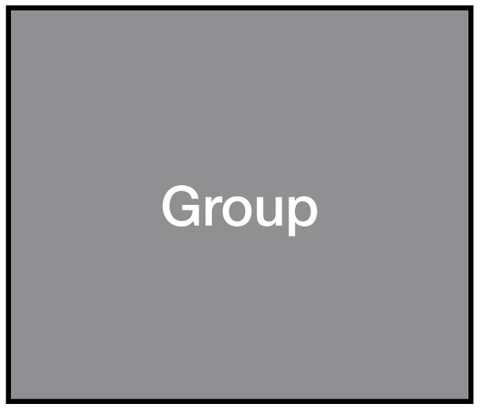
Interactive Construction [CMZ14]





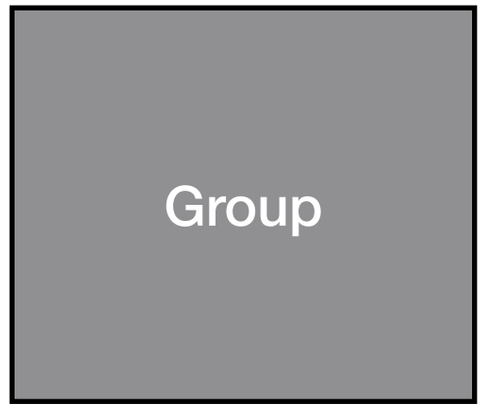
Efficiency

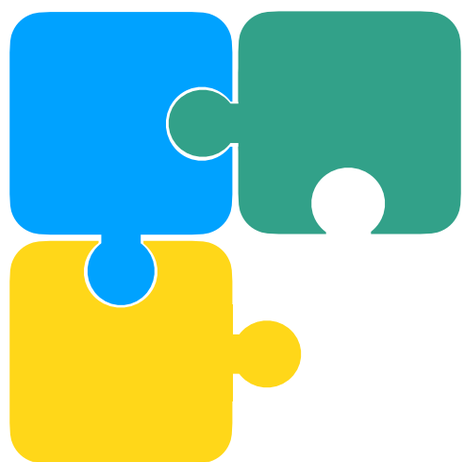
This Work



7× smaller

Naive Construction
[CMZ14 + CC18]





Conclusion

NIKVAC

- *small proofs of possession*
- *several useful properties*
- *secure in standard model*

Link to the Paper

- *ia.cr/2019/117*

Thanks for Listening

- *questions?*