EUDI Wallet & Anonymous Credentials

Status and Open Challenges

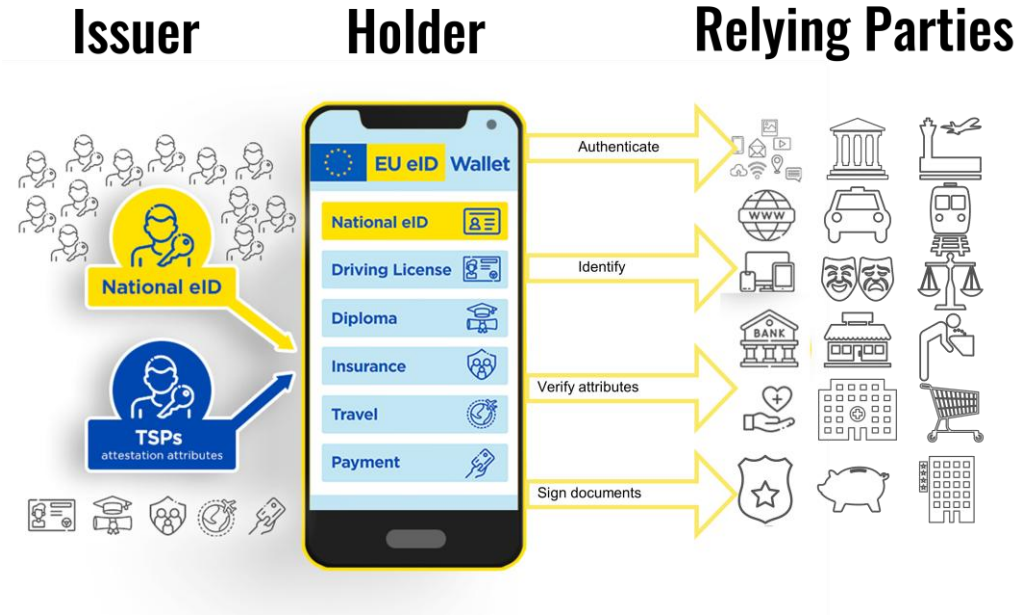
PrivCrypt Workshop @ Eurocrypt 2026

Anja Lehmann

HPI Hasso Plattner Institut
Digital Engineering · Universität Potsdam

European Digital Identity Wallet (EUDI)

- All EU member states must provide a EUDI Wallet by end of 2026



"fully mobile, secure and user-friendly"

https://ec.europa.eu/digital-building-blocks/sites/display/EUDIGITALIDENTITYWALLET/EU+Digital+Identity+Wallet+Home

EUDI | Privacy Requirements in eIDAS

- EU regulation eIDAS 2.0 mandates strong privacy guarantees

> "securely [..] authenticate to relying parties [..] while ensuring **selective disclosure of data** [..]
>
> enable privacy-preserving techniques which ensure **unlinkability** [..]
>
> possibility of users to access services through the use of **pseudonyms** [..]
>
> providers should ensure **unobservability** by not collecting data and not having insight into the transactions of the users [..]

https://eur-lex.europa.eu/eli/reg/2024/1183/oj
https://eidas-open-letter.org/

EUDI | Privacy Requirements in eIDAS

- EU regulation eIDAS 2.0 mandates strong privacy guarantees

"securely [..] authenticate to relying parties [..] while ensuring **selective disclosure of data** [..]

enable privacy-preserving techniques which ensure **unlinkability** [..]

possibility of users to access services through the use of **pseudonyms** [..]

providers should ensure **unobservability** by not collecting data and not having insight into the transactions of the users [..]

§ 16. The technical framework of the European Digital Identity Wallet shall:

(a) **not allow providers** of electronic attestations of attributes or any other party, after the issuance of the attestation of attributes, **to obtain data that allows transactions or user behaviour** to be **tracked, linked or correlated**, or knowledge of transactions or user behaviour to be otherwise obtained, unless explicitly authorised by the user;

(b) enable privacy preserving techniques which ensure unlikeability

https://eur-lex.europa.eu/eli/reg/2024/1183/oj

https://eidas-open-letter.org/

EUDI | Privacy Requirements in eIDAS

- EU regulation eIDAS 2.0 mandates strong privacy guarantees

"securely [..] authenticate to relying parties [..] while ensuring **selective disclosure of data** [..]

enable privacy-preserving techniques which ensure **unlinkability** [..]

possibility of users to access services through the use of **pseudonyms** [..]

providers should ensure **unobservability** by not collecting data and not having insight into the transactions of the users [..]

§ 16. The technical framework of the European Digital Identity Wallet shall:

(a) **not allow providers** of electronic attestations of attributes or any other party, after the issuance of the attestation of attributes, **to obtain data that allows transactions or user behaviour** to be **tracked, linked or correlated**, or knowledge of transactions or user behaviour to be otherwise obtained, unless explicitly authorised by the user;

(b) enable privacy preserving techniques which ensure ~~unlikeability~~

Corrigendum 9.4.25: unlinkability!

https://eur-lex.europa.eu/eli/reg/2024/1183/oj
https://eidas-open-letter.org/

EUDI | From Law to Technology ...

- Architecture Reference Framework (ARF) & Implementing Acts: https://eudi.dev/

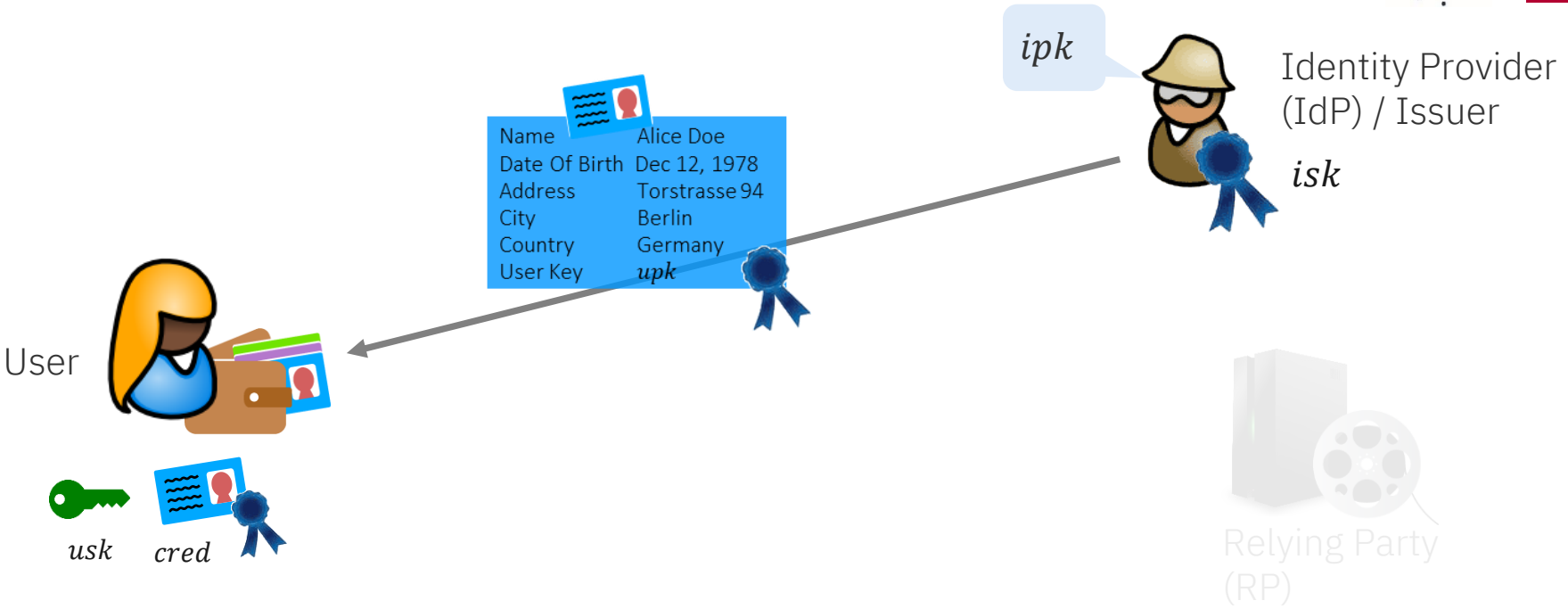
EUDI | From Law to Technology ...

- Architecture Reference Framework (ARF) & Implementing Acts: https://eudi.dev/

ECDSA

... maybe they did mean “unlikeability” ...

EUDI Wallet | Selective Disclosure with ECDSA

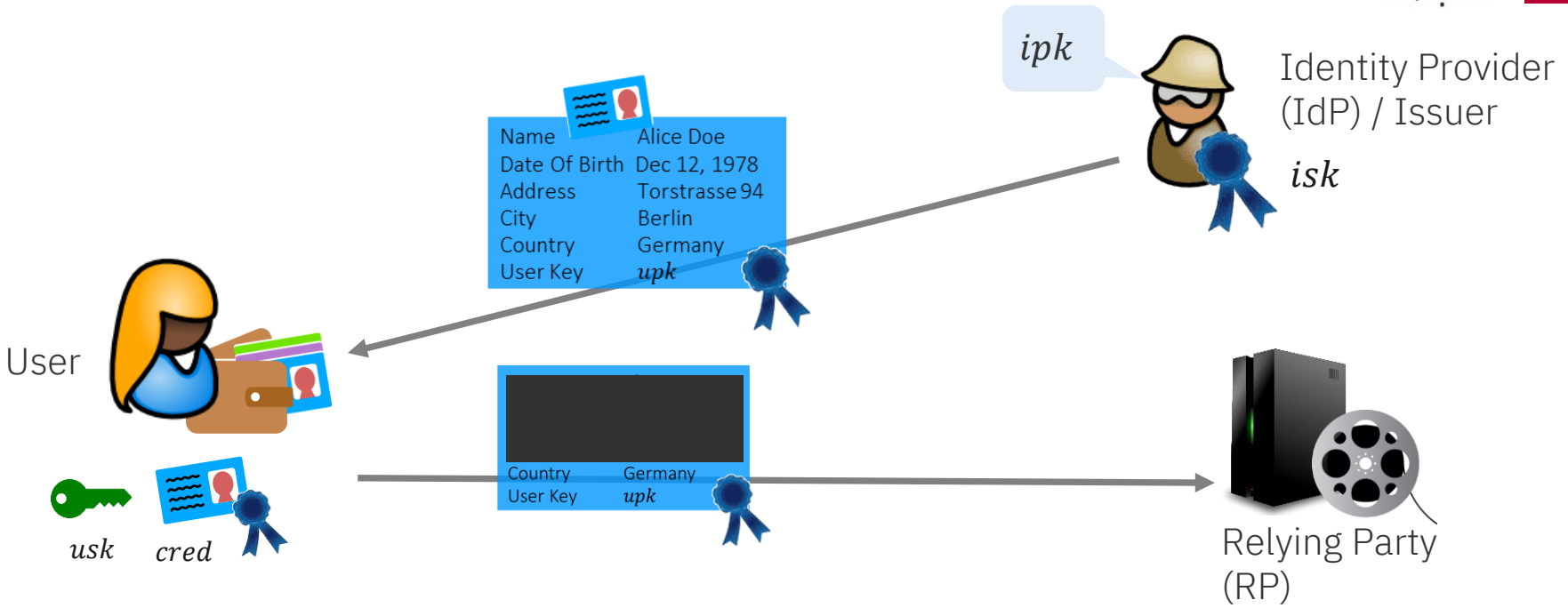


Random salt $s_1, s_2, \ldots, s_\ell$

Attributes $a_1, a_2, \ldots, a_\ell$

$$cred = Sign(isk, (h_1, h_2, \ldots, h_\ell), upk)$$

EUDI Wallet | Selective Disclosure with ECDSA

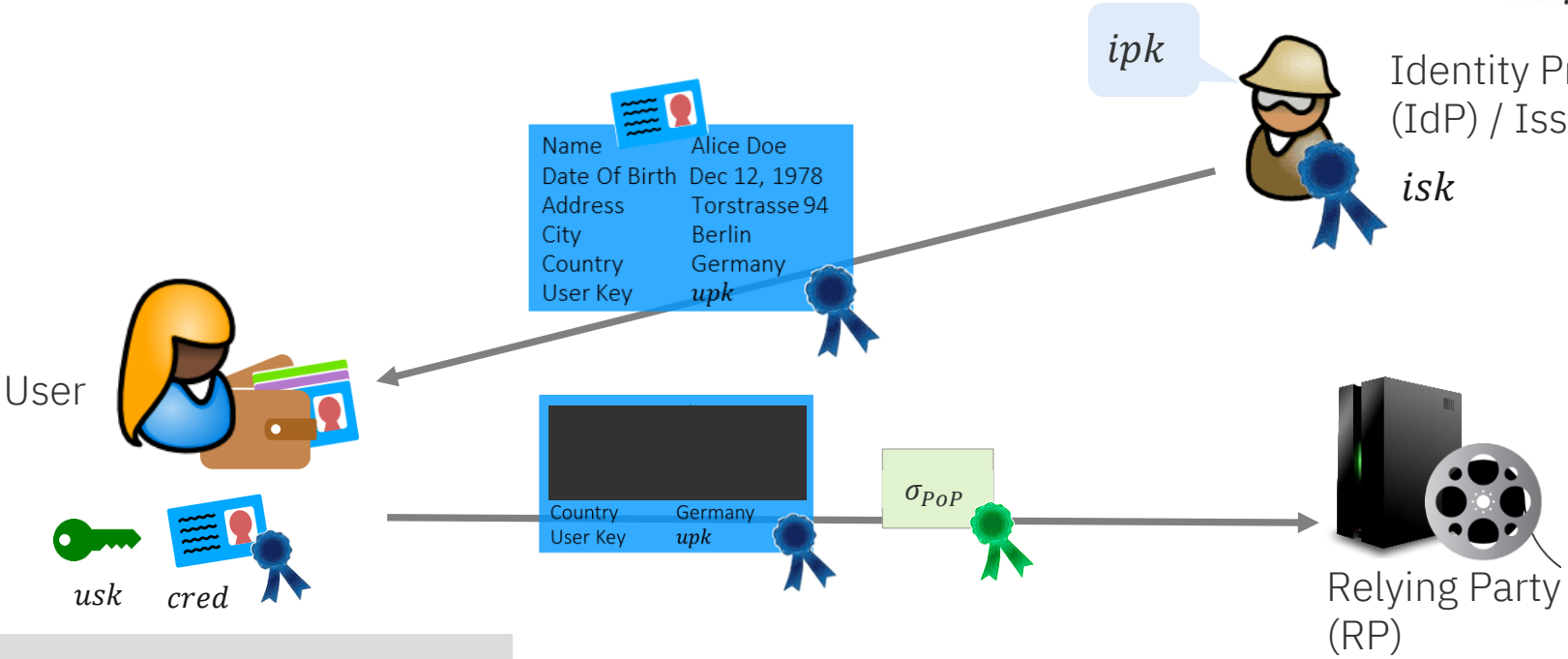


Random salt [redacted], $s_2, \dots,$ [redacted]

Attributes [redacted], $a_2, \dots,$ [redacted]

$$cred = Sign(isk, (h_1, h_2, \dots, h_\ell), upk)$$

EUDI Wallet | Selective Disclosure with ECDSA



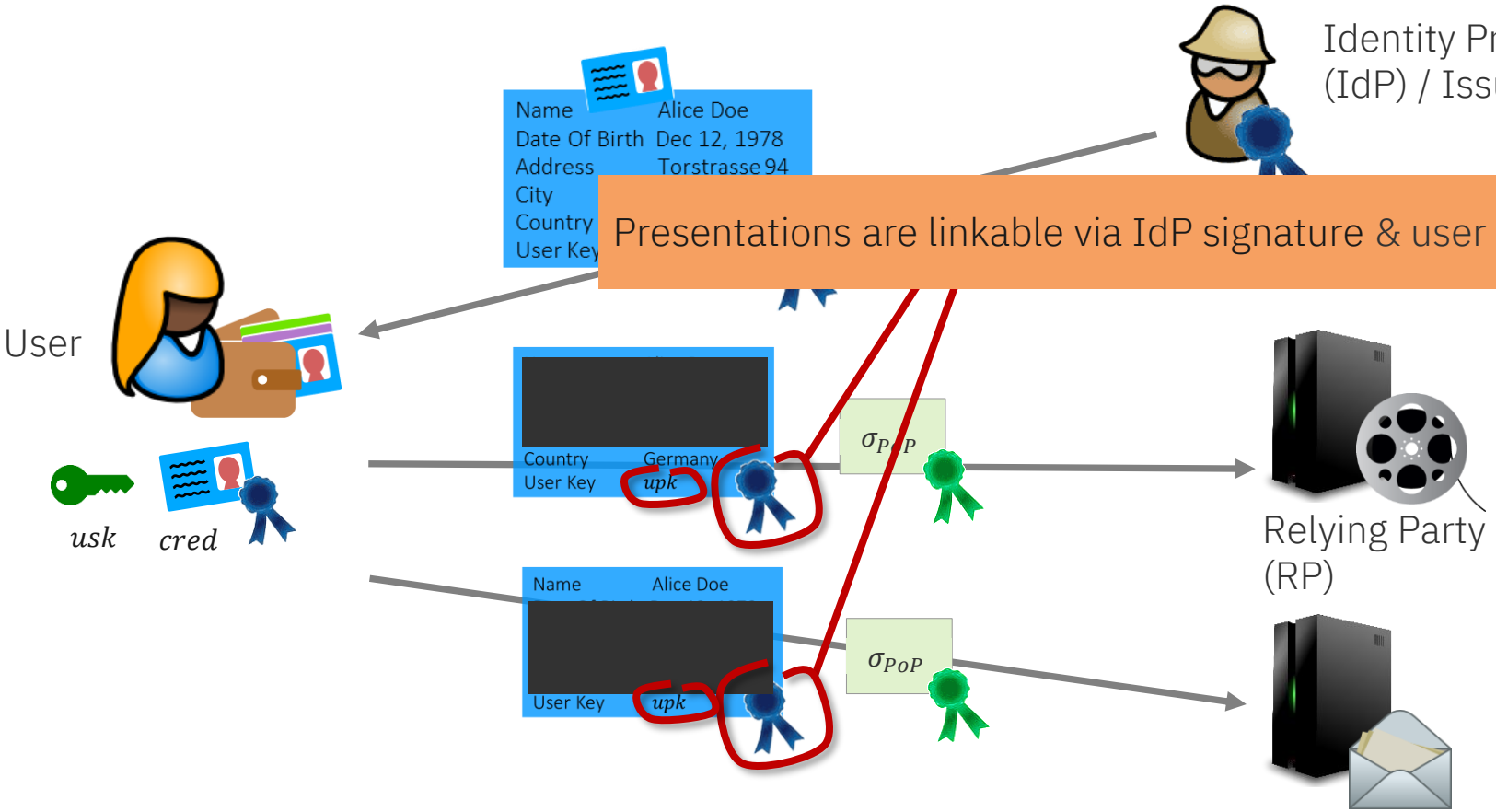
$$\sigma_{PoP} := Sign(usk, nonce)$$

Random salt [redacted], s_2, …, [redacted]

Attributes [redacted], a_2, …, [redacted]

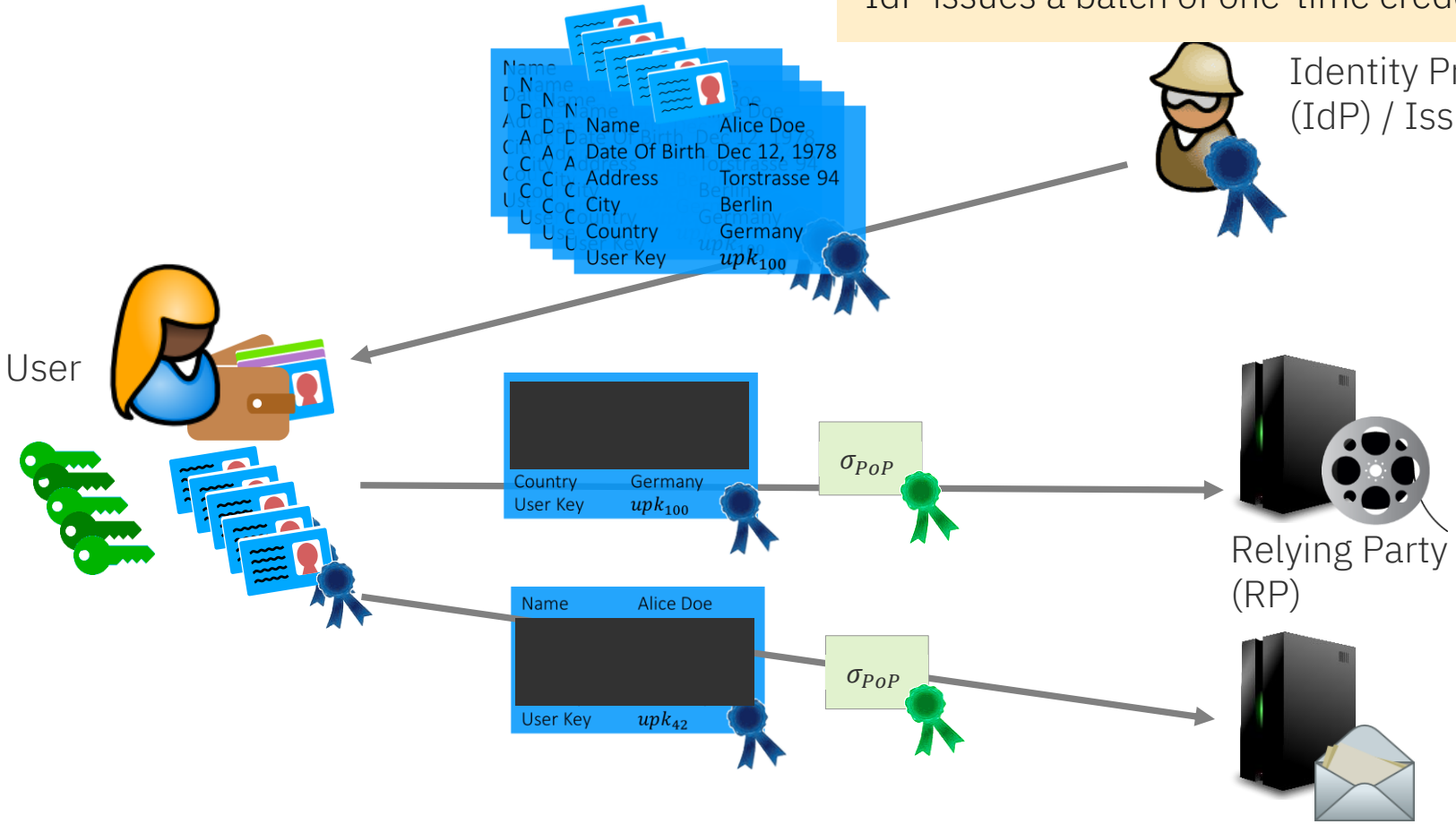
$$cred = Sign(isk, (h_1, h_2, \ldots, h_\ell), upk)$$

EUDI Wallet | Linkability of Presentations



EUDI Wallet | Batch Issuance

IdP issues a batch of one-time credentials

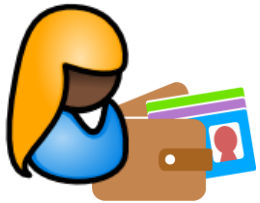


EUDI Wallet | Batch Issuance

IdP issues a batch of one-time credentials

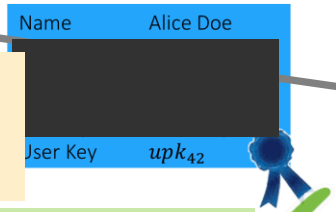
Identity Provider (IdP) / Issuer

User



σ_{PoP}

Relying Party (RP)



σ_{PoP}

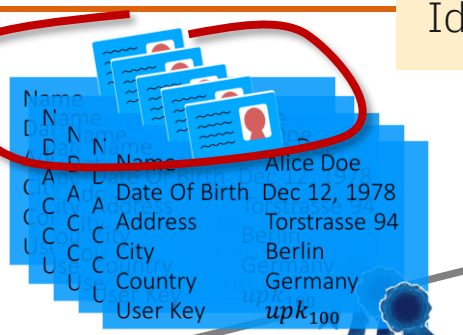
Uses different key & credential with every authentication

RP-RP Unlinkability (with trade-offs)

EUDI Wallet | Batch Issuance

IdP issues a batch of one-time credentials

Expensive & cumbersome

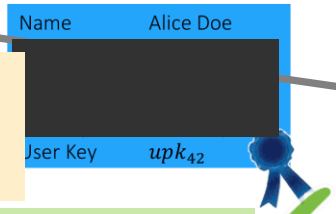


Identity Provider (IdP) / Issuer

User

σ_{PoP}

Relying Party (RP)



σ_{PoP}

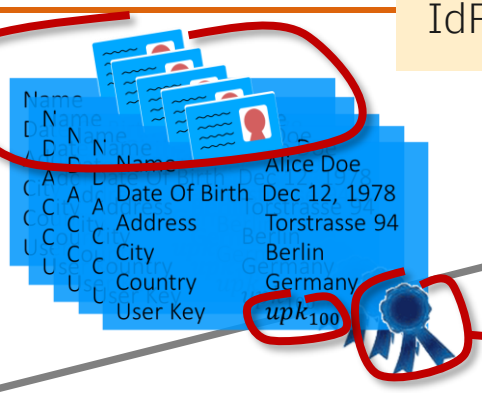
Uses different key & credential with every authentication

RP-RP Unlinkability (with trade-offs)

EUDI Wallet | Batch Issuance

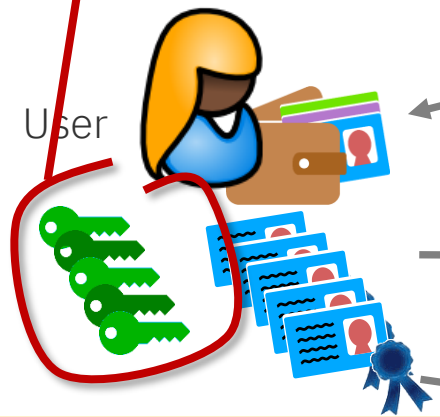
IdP issues a batch of one-time credentials

Expensive & cumbersome



Identity Provider (IdP) / Issuer

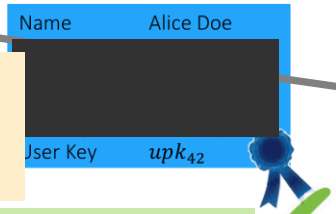
No IdP-RP Unlinkability



User

σ_{PoP}

Relying Party (RP)



σ_{PoP}

Uses different key & credential with every authentication

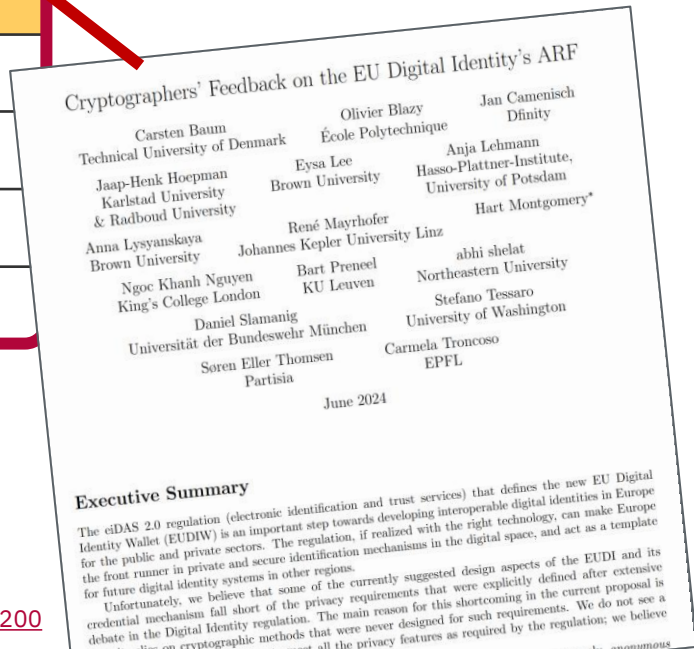
RP-RP Unlinkability (with trade-offs)

Cryptographers‘ Feedback on the EUDI ARF

- Current solution does not satisfy unlinkability requirement mandated by eIDAS

... But we have solutions with build-in privacy & unlinkability

Properties	ECDSA + Batch Issuance	Anonymous Credentials
Unobservability	✓	✓
Selective Disclosure	Salted hashes ✓	✓
RP ↔ RP Unlinkability	Batch issuance ✓	✓
IdP ↔ RP Unlinkability	Impossible ✗	✓



Cryptographers' Feedback on the EU Digital Identity's ARF

Carsten Baum
Technical University of Denmark
Olivier Blazy
École Polytechnique
Jan Camenisch
Dfinity
Jaap-Henk Hoepman
Karlstad University
& Radboud University
Eysa Lee
Brown University
Anja Lehmann
Hasso-Plattner-Institute,
University of Potsdam
Anna Lysyanskaya
Brown University
René Mayrhofer
Johannes Kepler University Linz
Hart Montgomery*
Ngoc Khanh Nguyen
King's College London
Bart Preneel
KU Leuven
abhi shelat
Northeastern University
Daniel Slamanig
Universität der Bundeswehr München
Stefano Tessaro
University of Washington
Søren Eller Thomsen
Partisia
Carmela Troncoso
EPFL

June 2024

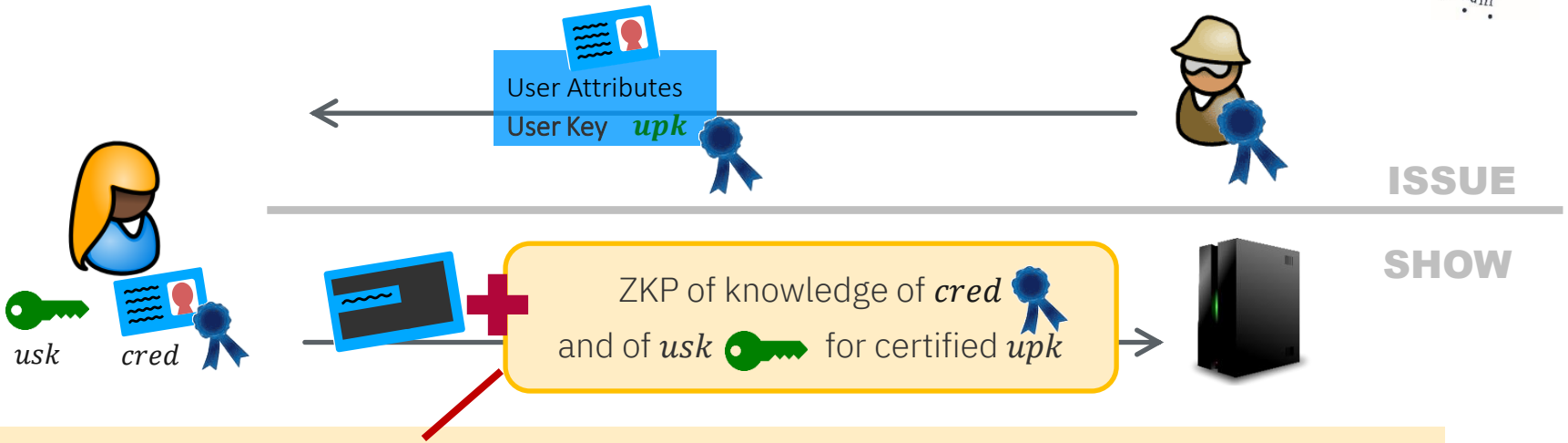
Executive Summary

The eIDAS 2.0 regulation (electronic identification and trust services) that defines the new EU Digital Identity Wallet (EUDIW) is an important step towards developing interoperable digital identities in Europe for the public and private sectors. The regulation, if realized with the right technology, can make Europe the front runner in private and secure identification mechanisms in the digital space, and act as a template for future digital identity systems in other regions.

Unfortunately, we believe that some of the currently suggested design aspects of the EUDI and its credential mechanism fall short of the privacy requirements that were explicitly defined after extensive debate in the Digital Identity regulation. The main reason for this shortcoming in the current proposal is

https://github.com/eu-digital-identity-wallet/eudi-doc-architecture-and-reference-framework/issues/200

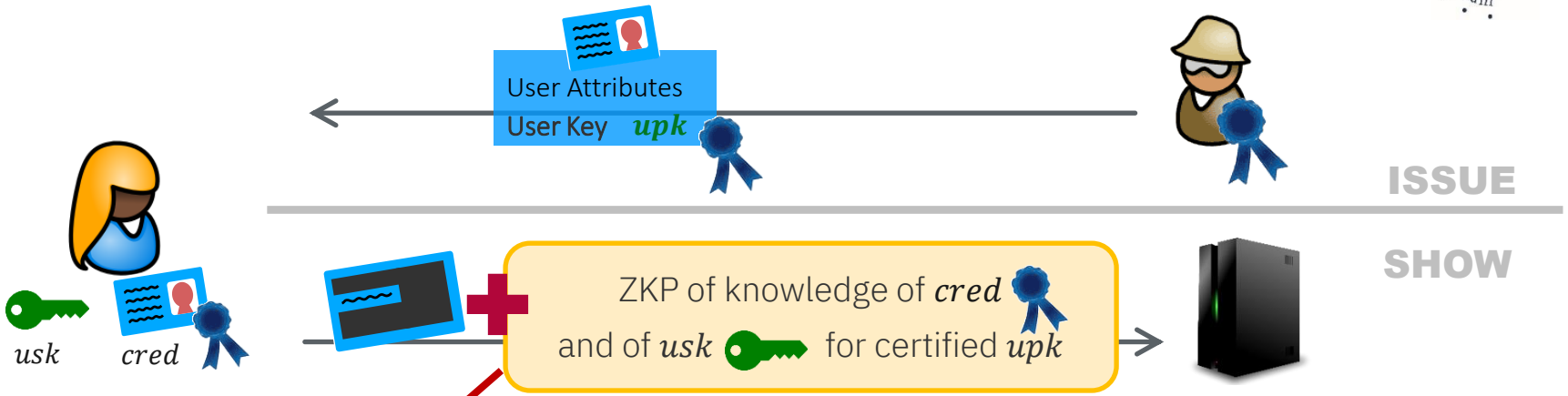
Anonymous Credentials | Privacy through ZKPs



Zero-Knowledge Proof (ZKP)

Proof of a statement that reveals *nothing* beyond validity

Anonymous Credentials | Privacy through ZKPs



Zero-Knowledge Proof (ZKP)

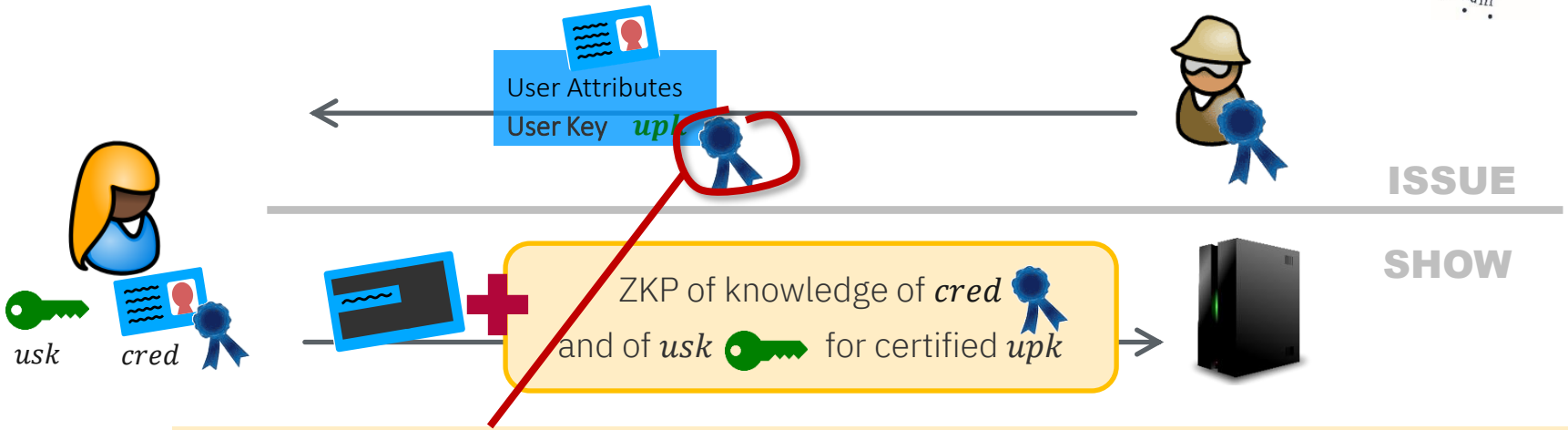
Proof of a statement that reveals *nothing* beyond validity

Here: user proves she owns $cred$ from IdP on the revealed attributes & knows usk but reveals nothing about IdP‘s signature, or upk (!)

Multi-Show RP-RP & IdP-RP Unlinkability ✓✓

Selective disclosure ✓

Anonymous Credentials | Signatures with ZKPs



Needs **signature scheme** (for IdP) **that allows for efficient ZKP of a signature**

Option 1 | Dedicated signature scheme with„build-in“ ZKP-capabilities

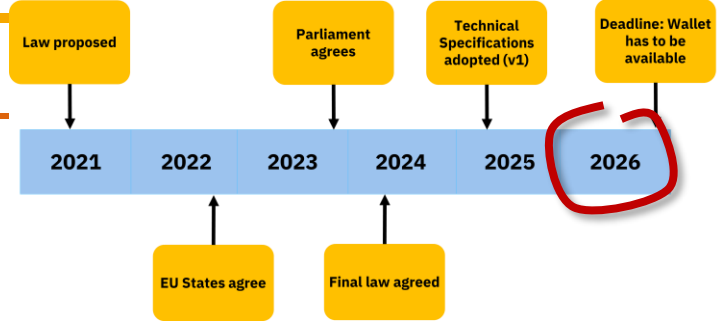
E.g., CL/**BBS**/PS-signatures → ZKP is then a Schnorr-type proof

Option 2 | Use any signature scheme (e.g., ECDSA) & generic (circuit-based) ZKP

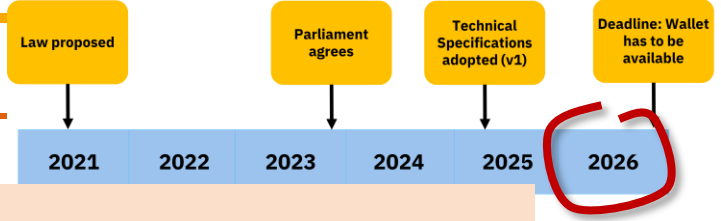
Legacy-compatible, but less efficient & more complex

[BL25] Bormann, Lehmann. SoK: Anonymous Credentials for Digital Identity Wallets. SSR25

Why does EUDI not use ACs (yet) ?



Why does EUDI not use ACs (yet) ?



1 | All protocols and schemes must be **standardized** → interoperability (& sign of maturity)

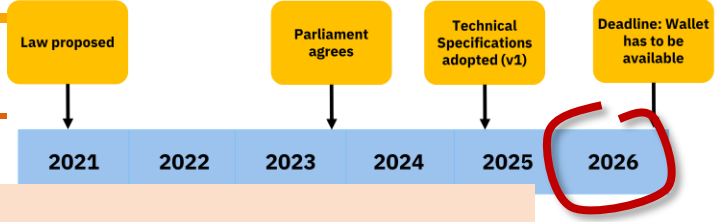
Regulated use cases (e.g. eID): all crypto must be in "Agreed Cryptographic Mechanisms" by ENISA

(previously SOG-IS catalogue)

Primitive	Scheme	R/L	Notes
RSA	PSS (PKCS#1v2.1) [RFC8017, PKCS1, ISO9796-2]	R	
FF-DLOG	KCDSA [ISO14888-3]	R	
	Schnorr [ISO14888-3]	R	41-DSARandom
	DSA [FIPS186-4, ISO14888-3]	R	
EC-DLOG	EC-KCDSA [ISO14888-3]	R	
	EC-DSA [FIPS186-4, ISO14888-3]	R	41-DSARandom
	EC-GDSA [TR-03111]	R	
	EC-Schnorr [ISO14888-3]	R	
RSA	PKCS#1v1.5 [RFC8017, PKCS1, ISO9796-2]	L	40-PKCSFormatCheck

List of „approved" crypto when EUDI ARF was designed (now it also includes PQC)

Why does EUDI not use ACs (yet) ?



1 | All protocols and schemes must be **standardized** → interoperability (& sign of maturity)

Regulated use cases (e.g. eID): all crypto must be in "Agreed Cryptographic Mechanisms" by ENISA

(previously SOG-IS catalogue)

Primitive	Scheme	R/L	Notes
RSA	PSS (PKCS#1v2.1) [RFC8017, PKCS1, ISO9796-2]	R	
FF-DLOG	KCDSA [ISO14888-3]	R	
	Schnorr [ISO14888-3]	R	41-DSARandom
	DSA [FIPS186-4, ISO14888-3]	R	
EC-DLOG	EC-KCDSA [ISO14888-3]	R	
	EC-DSA [FIPS186-4, ISO14888-3]	R	41-DSARandom
	EC-GDSA [TR-03111]	R	
	EC-Schnorr [ISO14888-3]	R	
RSA	PKCS#1v1.5 [RFC8017, PKCS1, ISO9796-2]	L	40-PKCSFormatCheck

List of „approved" crypto when EUDI ARF was designed (now it also includes PQC)

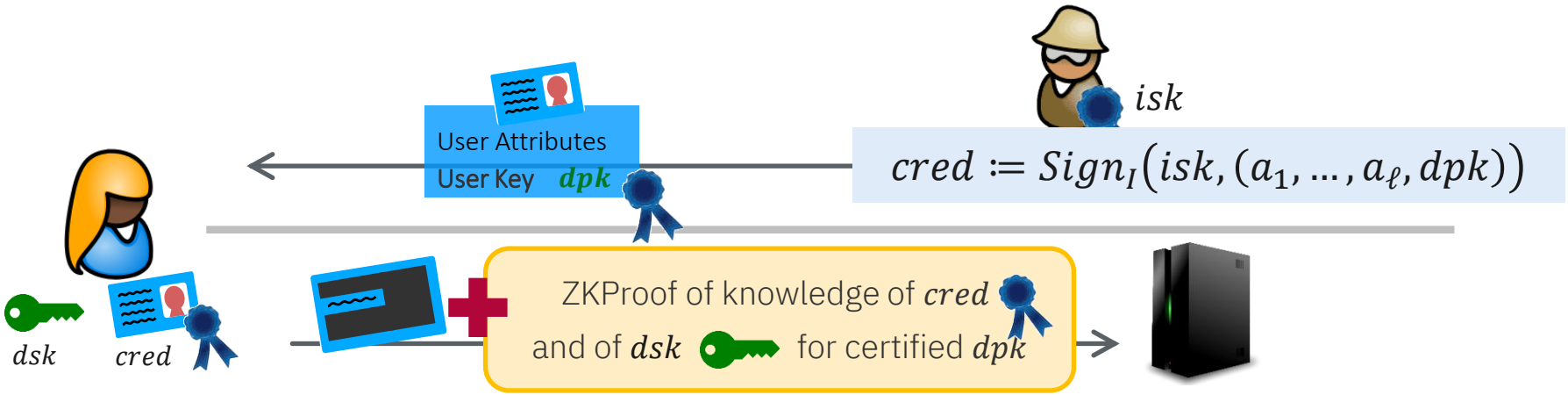
2 | Credential must be bound to **hardware**-protected device key

EUDI Wallet requires Level-of-Assurance (LoA) High

Secure Elements support only ECDSA (and curve P256)

Standards & Hardware Support for Anonymous Credentials

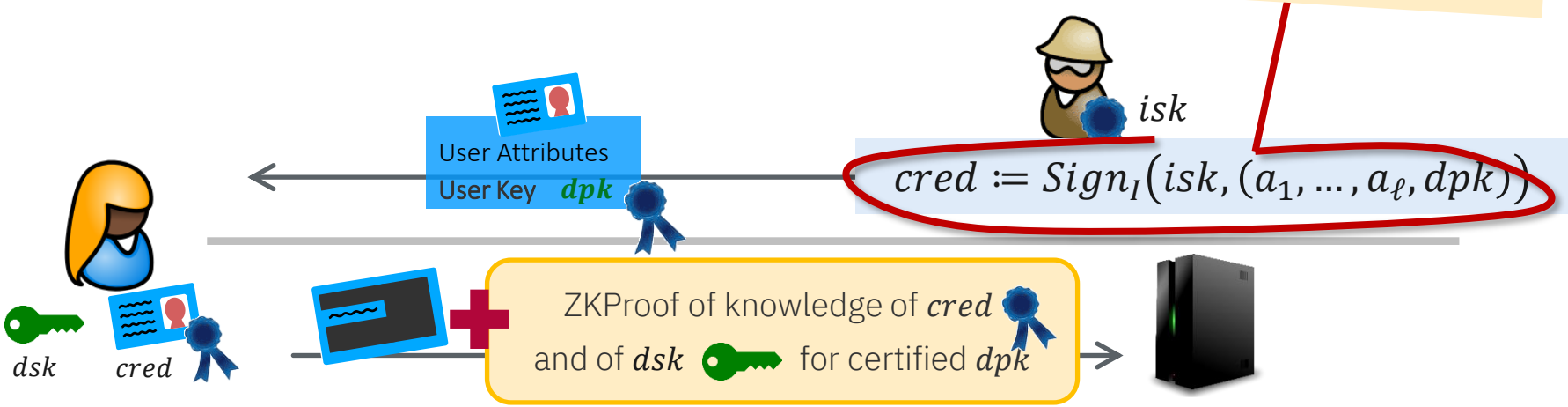
- Signature scheme for Issuer & $cred$
- ZKP for presentation incl. device-binding



Standards & Hardware Support for Anonymous Credentials

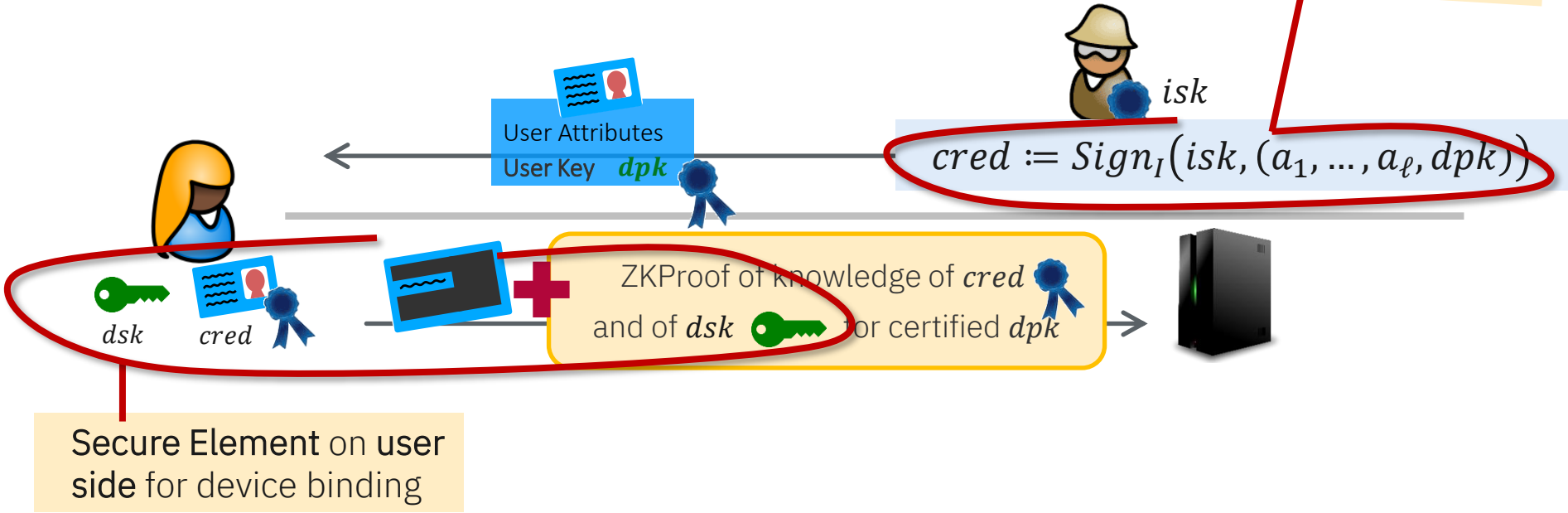
- Signature scheme for Issuer & $cred$
- ZKP for presentation incl. device-binding

Signature for credentials
HSM for increased protection



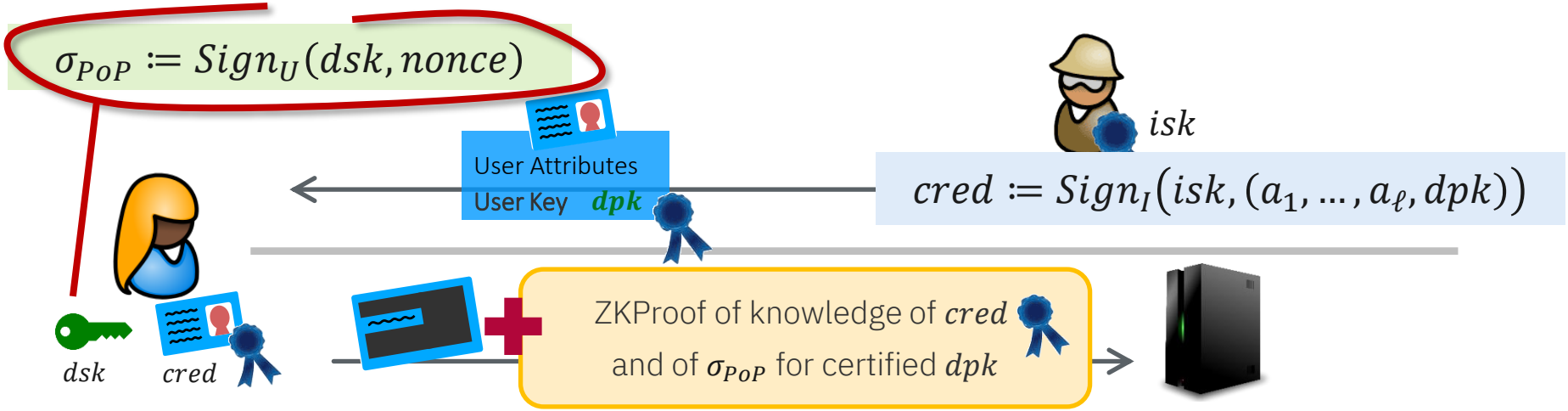
Standards & Hardware Support for Anonymous Credentials

- Signature scheme for Issuer & $cred$
- ZKP for presentation incl. device-binding



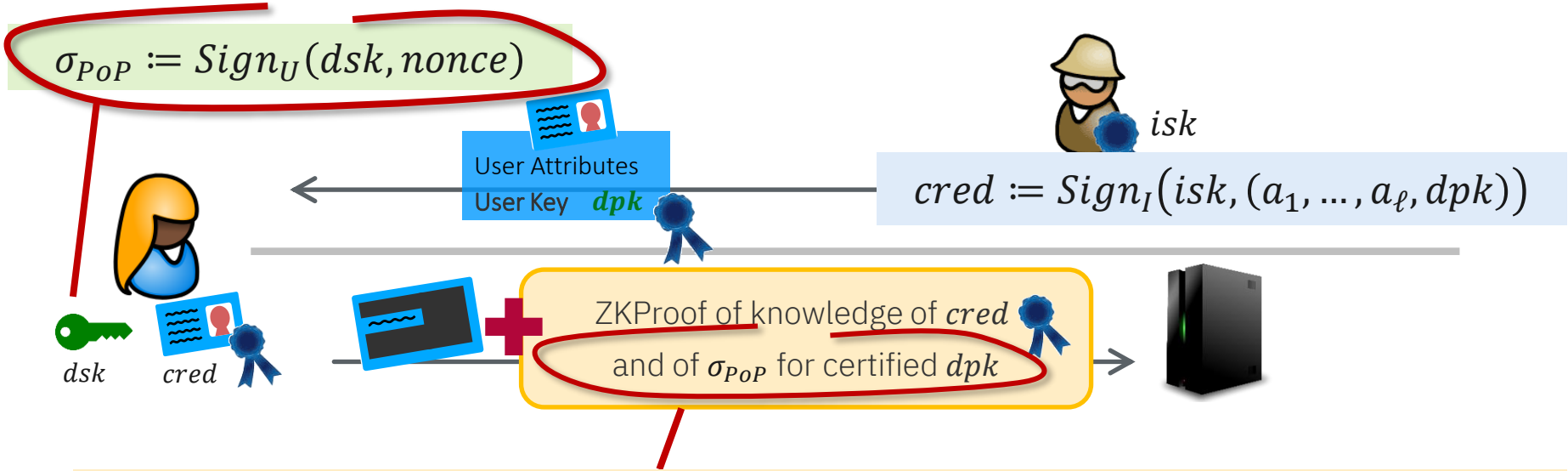
Standards & Hardware Support for Anonymous Credentials

- Secure Element only computes signature



Standards & Hardware Support for Anonymous Credentials

- Secure Element only computes signature

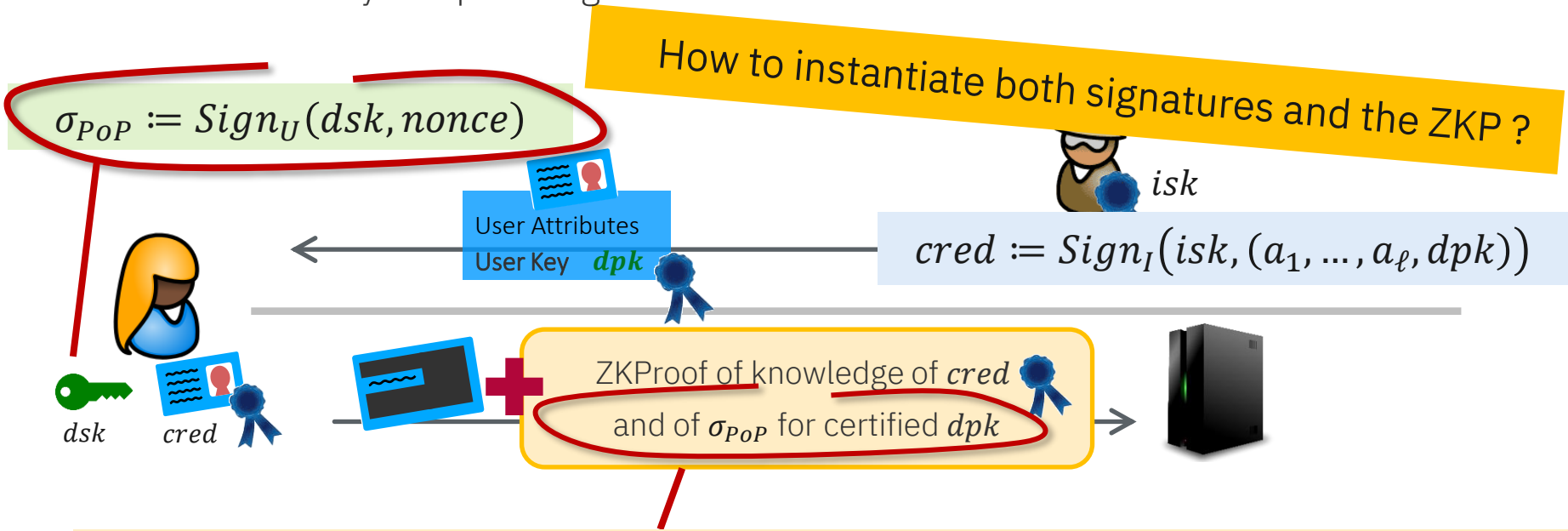


ZKP runs entirely on “public” values

→ ZKP not needed in secure hardware, only standardized (and “certified” for LoA high)

Standards & Hardware Support for Anonymous Credentials

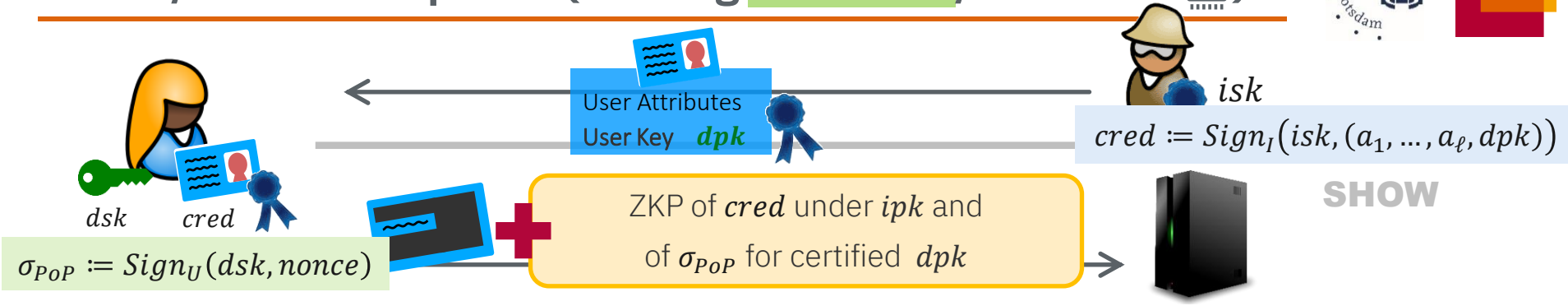
- Secure Element only computes signature



ZKP runs entirely on “public” values

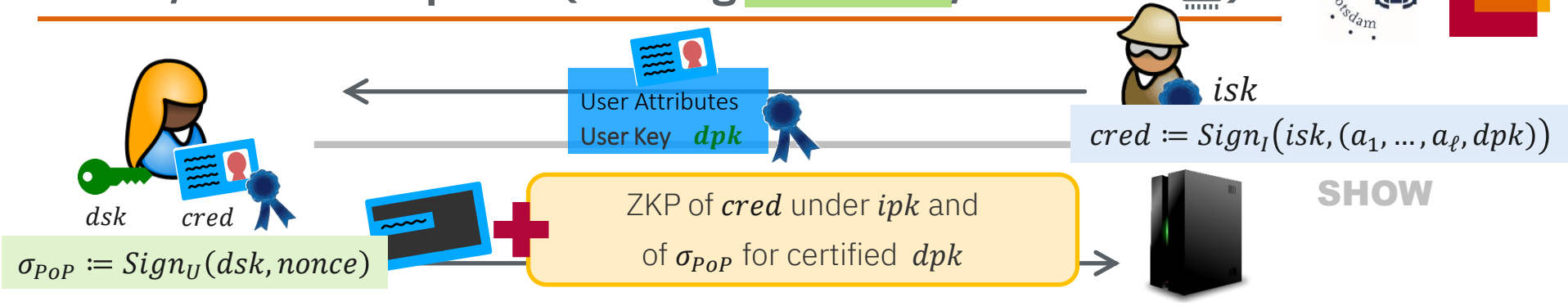
→ ZKP not needed in secure hardware, only standardized (and “certified” for LoA high)

Short/Mid-Term Options (Existing Standards/Hardware)



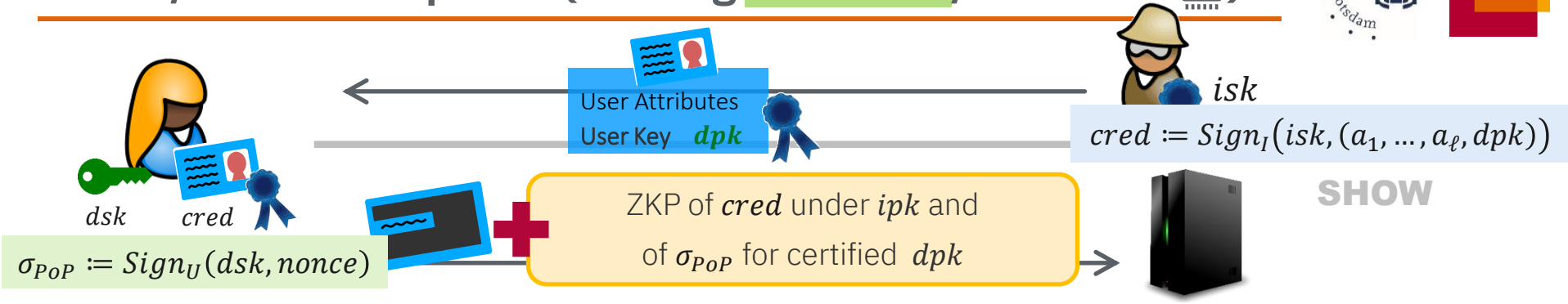
Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Notes
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Legacy compliant But very complex

Short/Mid-Term Options (Existing Standards/Hardware)



Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Notes
BBS Pairing-friendly curve, e.g. BLS12-381	Schnorr(ish)	Schnorr-type	~10 ms, 1kB	~DAA (ISO 2013),TPM2.0 Dedicated SE API
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Legacy compliant But very complex

Short/Mid-Term Options (Existing Standards/Hardware)



Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Notes
BBS Pairing-friendly curve, e.g. BLS12-381	Schnorr(ish)	Schnorr-type	~10 ms, 1kB	~DAA (ISO 2013),TPM2.0 Dedicated SE API
BBS Pairing-friendly curve, e.g. BLS12-381	BLS or Schnorr	Schnorr-type	~10 ms, 1kB	Simple SE API, Better privacy, (Cloud HSM) But: not on hardware yet
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Legacy compliant But very complex

Short/Mid-Term Options (Existing Standards/Hardware)

User Attributes
User Key dpk

$cred \coloneqq Sign_I(isk, (a_1, \dots, a_\ell, dpk))$

$\sigma_{PoP} \coloneqq Sign_U(dsk, nonce)$

ZKP of $cred$ under ipk and
of σ_{PoP} for certified dpk

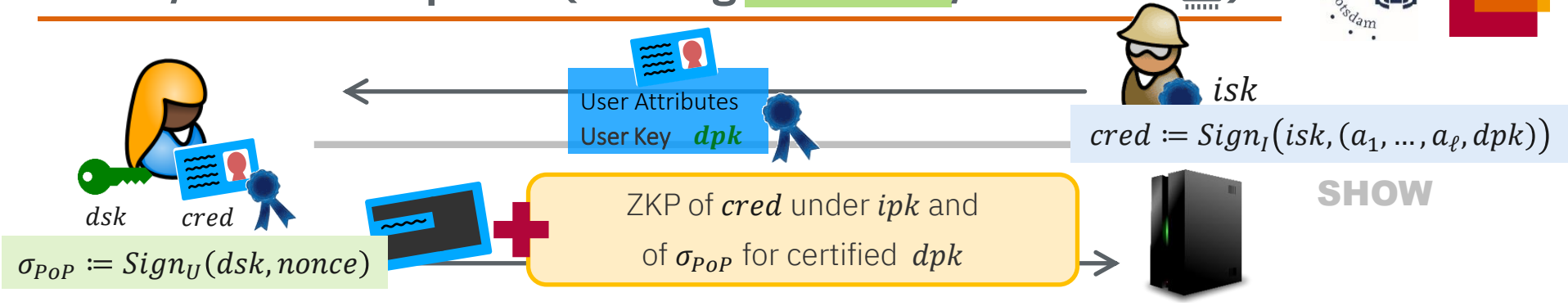
SHOW

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Notes
BBS Pairing-friendly curve, e.g. BLS12-381	Schnorr(ish)	Schnorr-type	~10 ms, 1kB	~DAA (ISO 2013),TPM2.0 Dedicated SE API
BBS Pairing-friendly curve, e.g. BLS12-381	BLS or Schnorr	Schnorr-type	~10 ms, 1kB	Simple SE API, Better privacy, (Cloud HSM) But: not on hardware yet
ECDS…			…0kB	Legacy compliant But very complex

Uses randomisable signatures/keys [FHLL26]
Want to know more? Karla's talk Tue 11:05 @ Eurocrypt

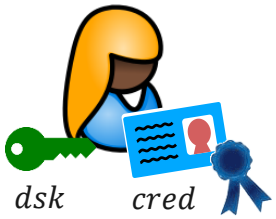
[FHLL26] Friedrichs, Harding, Lehmann, Lysyanskaya. Device-Bound Anonymous Credentials With(out) Trusted Hardware. Eurocrypt 2026

Short/Mid-Term Options (Existing Standards/Hardware)



Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Notes
BBS Pairing-friendly curve, e.g. BLS12-381	Schnorr(ish)	Schnorr-type	~10 ms, 1kB	~DAA (ISO 2013),TPM2.0 Dedicated SE API
BBS Pairing-friendly curve, e.g. BLS12-381	BLS or Schnorr	Schnorr-type	~10 ms, 1kB	Simple SE API, Better privacy, (Cloud HSM) But: not on hardware yet
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Legacy compliant But very complex

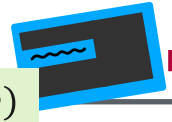
Short/Mid-Term Options (Existing Standards/Hardware)



$cred := Sign_I(isk, (a_1, \dots, a_\ell, dpk))$

SHOW

$\sigma_{PoP} := Sign_U(dsk, nonce)$



ZKP of $cred$ under ipk and of σ_{PoP} for certified dpk

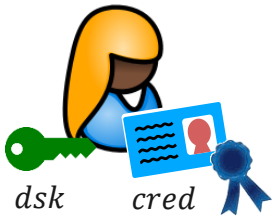
Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Notes
BBS Pairing-friendly curve, e.g. BLS12-381	BLS or Schnorr	Schnorr-type	~10 ms, 1kB	~DAA (ISO 2013),TPM2.0 Dedicated SE API
BBS Pairing-friendly curve, e.g. BLS12-381	BLS or Schnorr	Schnorr-type		Simple SE API, Better SM) vare yet
ECDSA	ECDSA	Circuit-based		

Not available in hardware

Very complex – challenge for security, standardisation, certification and extensibility

https://blog.trailofbits.com/2026/04/17/we-beat-googles-zero-knowledge-proof-of-quantum-cryptanalysis/

Device vs. Issuer Signature: Important Differences



isk

User Attributes
User Key dpk

$cred := Sign_I(isk, (a_1, \dots, a_\ell, dpk))$

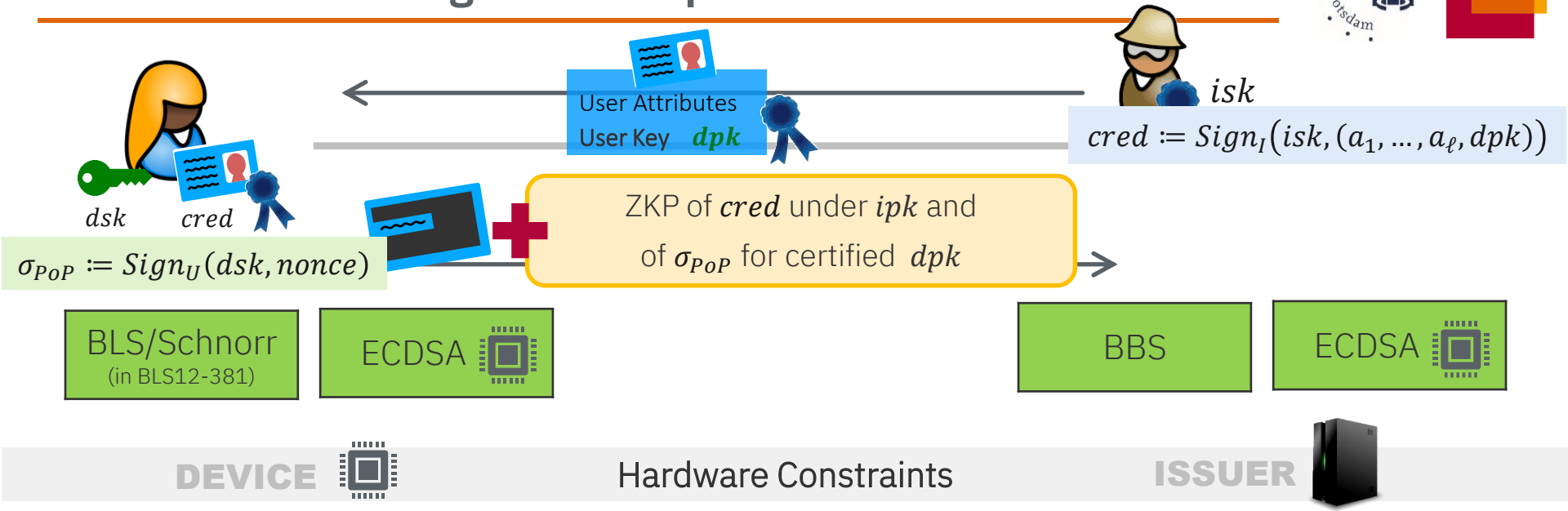
$\sigma_{PoP} := Sign_U(dsk, nonce)$

ZKP of $cred$ under ipk and
of σ_{PoP} for certified dpk

BLS/Schnorr (in BLS12-381)	ECDSA	BBS	ECDSA

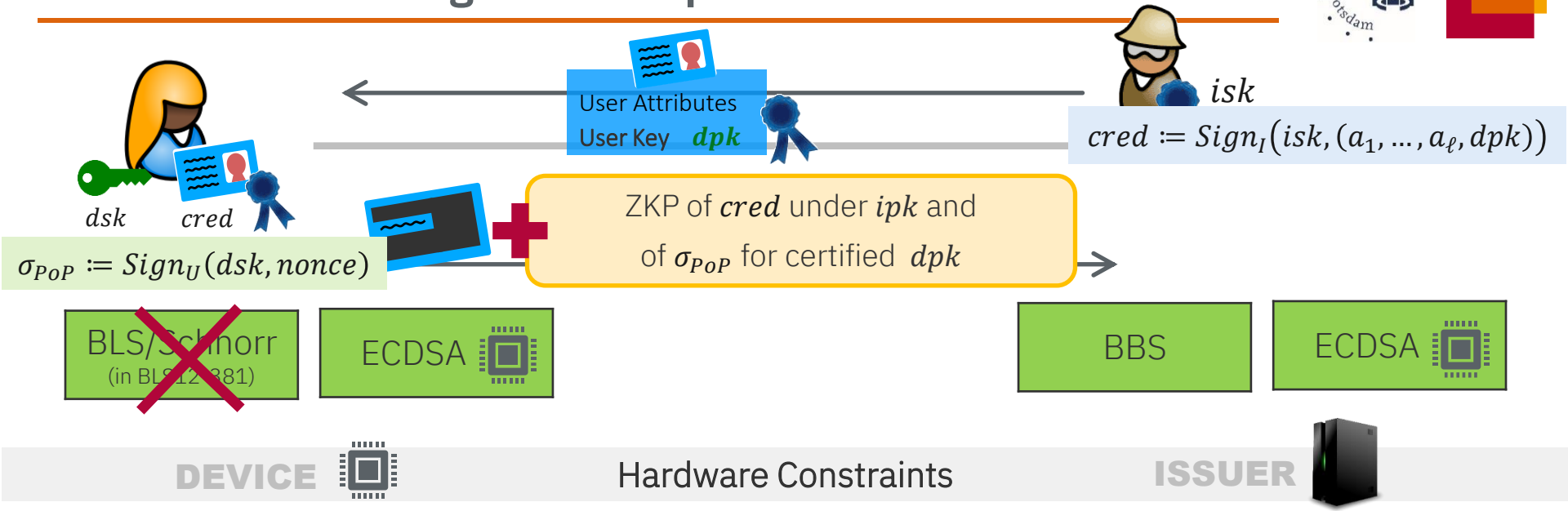
DEVICE — Hardware Constraints — ISSUER

Device vs. Issuer Signature: Important Differences



- Very hard to change/update
- Single/versatile API/scheme (small chip)
- Essential for non-transferability

Device vs. Issuer Signature: Important Differences



- Very hard to change/update
- Single/versatile API/scheme (small chip)
- Essential for non-transferability

→ ECDSA unavoidable for short-term

Device vs. Issuer Signature: Important Differences

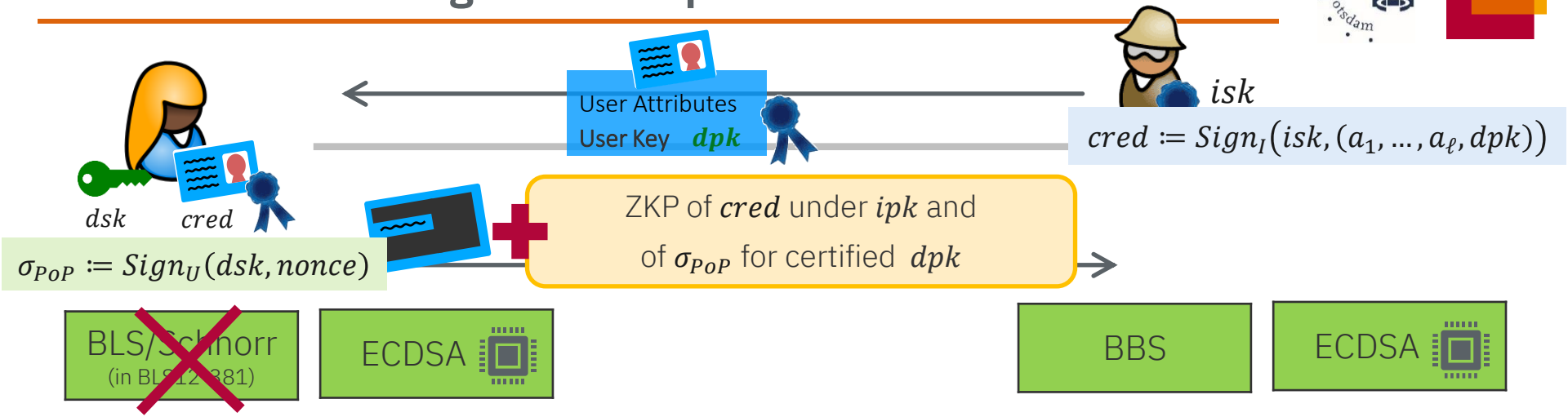
DEVICE — Hardware Constraints — ISSUER

- Very hard to change/update
- Single/versatile API/scheme (small chip)
- Essential for non-transferability

→ ECDSA unavoidable for short-term

- Easier to change (HSM ~ secure server)
- Can support multiple crypto schemes
- LoA high not always needed (e.g. age proofs)
- Threshold solutions provide similar guarantees

Device vs. Issuer Signature: Important Differences



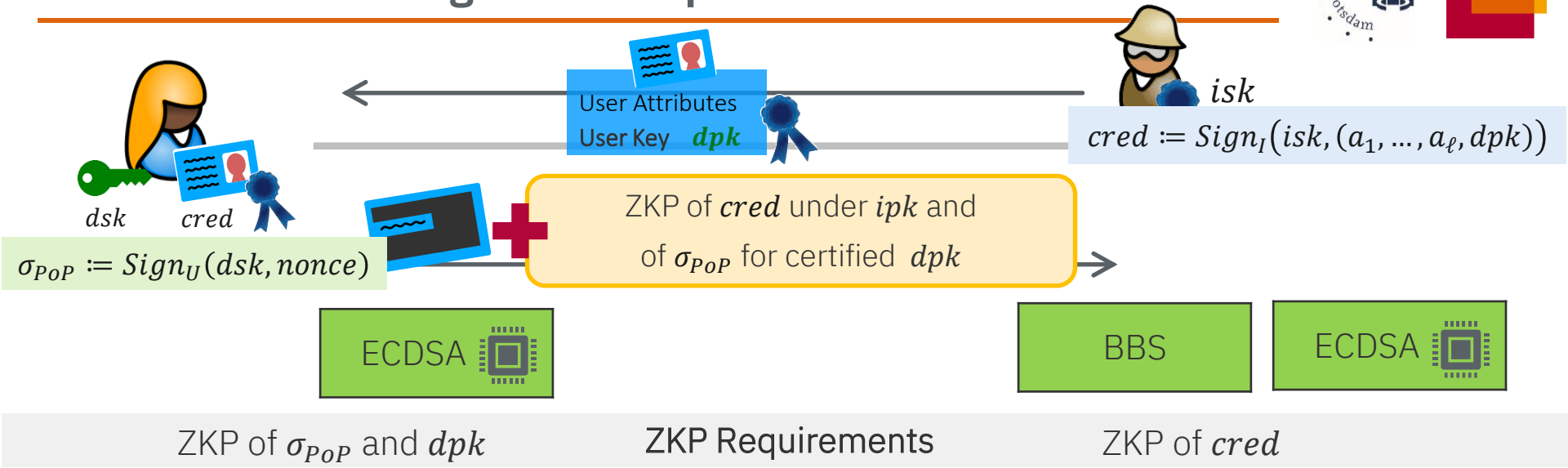
- Very hard to change/update
- Single/versatile API/scheme (small chip)
- Essential for non-transferability

→ ECDSA unavoidable for short-term

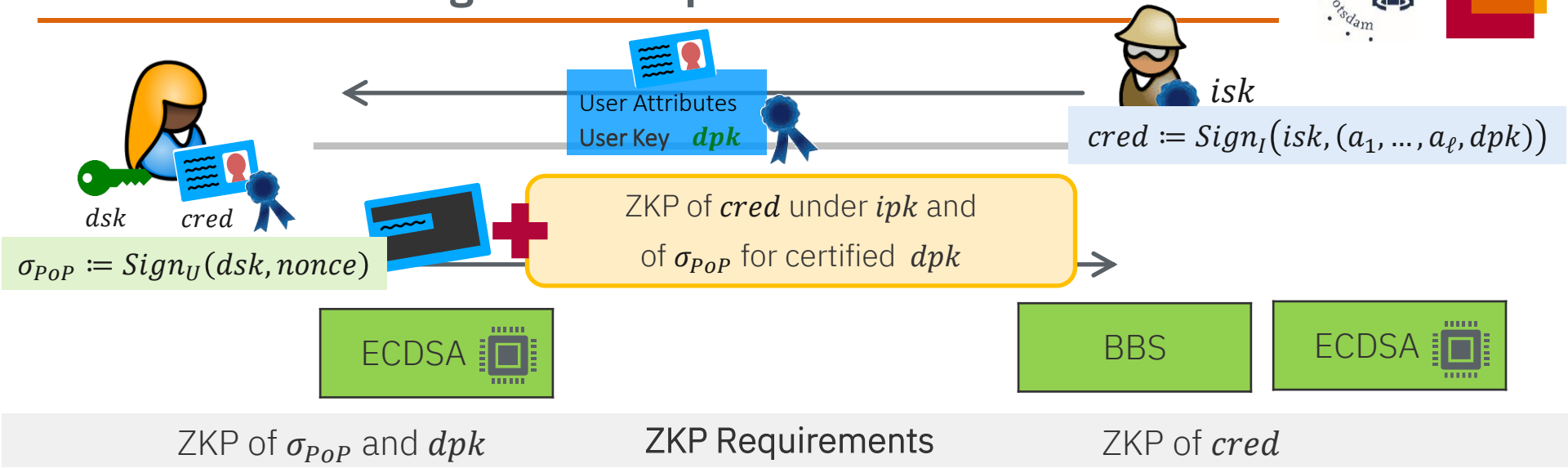
- Easier to change (HSM ~ secure server)
- Can support multiple crypto schemes
- LoA high not always needed (e.g. age proofs)
- Threshold solutions provide similar guarantees

→ ECDSA is avoidable

Device vs. Issuer Signature: Important Differences

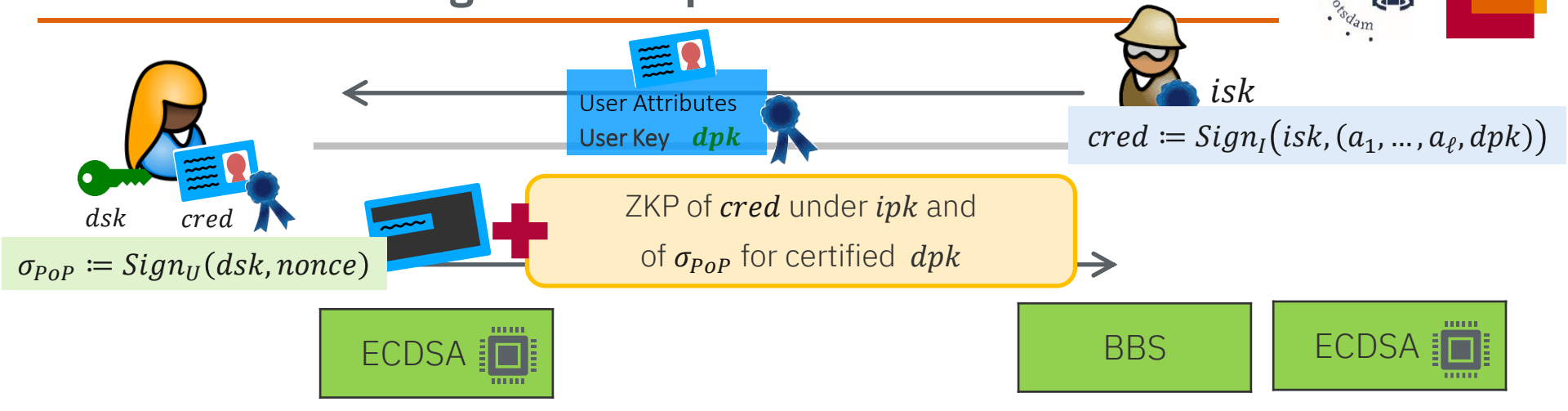


Device vs. Issuer Signature: Important Differences



- Message (nonce) is public
  → Nothing to prove about hash!
- Can reveal part of σ_{PoP} → fresh & used once
- Must hide dpk
- Statement is fix

Device vs. Issuer Signature: Important Differences

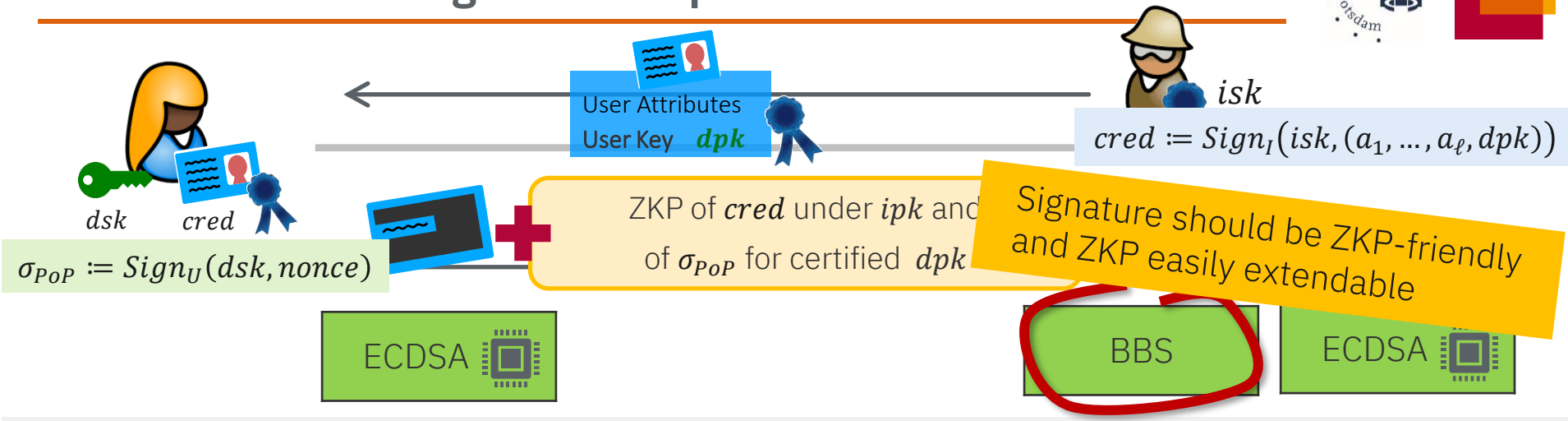


ZKP of σ_{PoP} and dpk | **ZKP Requirements** | ZKP of $cred$

- Message (nonce) is public
  - → Nothing to prove about hash!
- Can reveal part of σ_{PoP} → fresh & used once
- Must hide dpk
- Statement is fix

- Must hide messages
  - → Ideally, avoid hashing of $attr, dpk$
- Must hide $cred$ → **multi-show** unlinkability
- Typically reveals ipk
- Should support many statements & be easy to extend (nym, range proofs, …)

Device vs. Issuer Signature: Important Differences



ZKP of σ_{PoP} and dpk

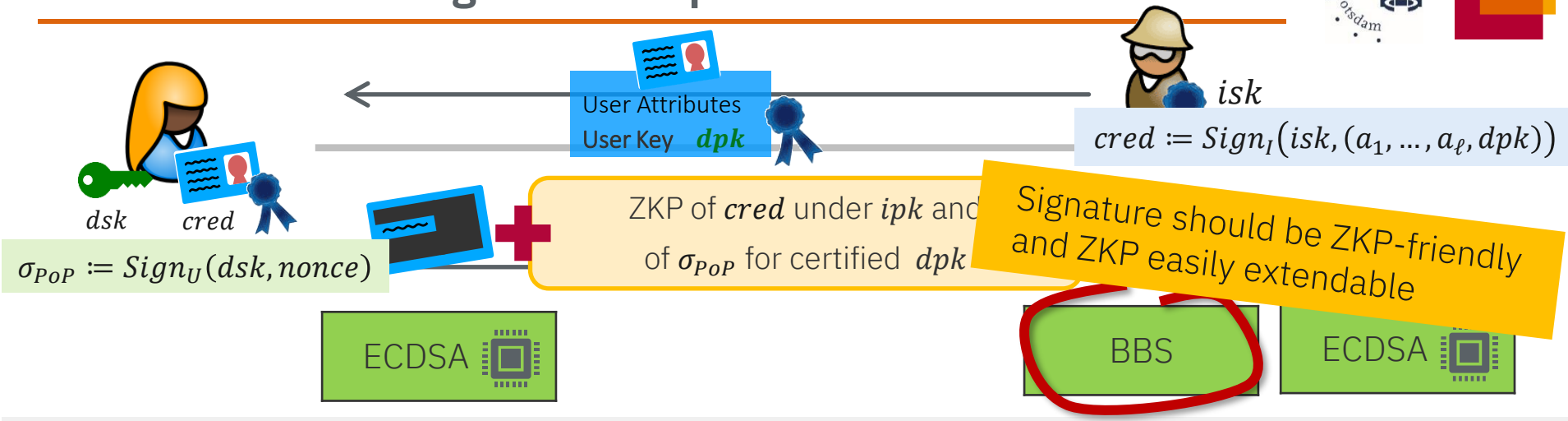
ZKP Requirements

ZKP of $cred$

- Message (nonce) is public
  - → Nothing to prove about hash!
- Can reveal part of σ_{PoP} → fresh & used once
- Must hide dpk
- Statement is fix

- Must hide messages
  - → Ideally, avoid hashing of $attr, dpk$
- Must hide $cred$ → **multi-show** unlinkability
- Typically reveals ipk
- Should support many statements & be easy to extend (nym, range proofs, …)

Device vs. Issuer Signature: Important Differences



ZKP of σ_{PoP} and dpk | **ZKP Requirements** | ZKP of $cred$

- Message (nonce) is public
  - → Nothing to prove about hash!
- Can reveal part of σ_{PoP} → fresh & used once
- Must hide dpk
- Statement is fix

This does not need full power of ZKP & can be „hardcoded“

- Must hide messages
  - → Ideally, avoid hashing of $attr, dpk$
- Must hide $cred$ → **multi-show** unlinkability
- Typically reveals ipk
- Should support many statements & be easy to extend (nym, range proofs, …)

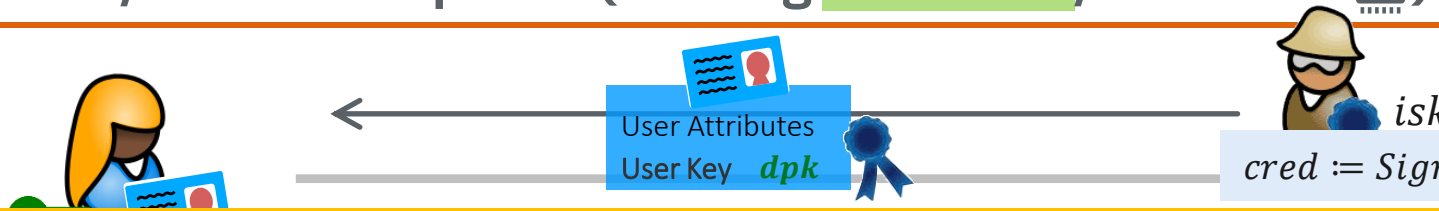
Short/Mid-Term Options (Existing Standards/Hardware)

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Maturity & Simplicity
BBS	BLS/Schnorr	Schnorr-type	~10 ms, 1kB	Simple, 20y of research
BBS BLS12-381	ECDSA P-256			
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Very complex, ongoing research

Short/Mid-Term Options (Existing Standards/Hardware)

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Maturity & Simplicity
BBS	BLS/Schnorr	Schnorr-type	~10 ms, 1kB	Simple, 20y of research
BBS BLS12-381	ECDSA P-256	Schnorr-type (a lot for ECDSA)	~400 ms, ~175kB	A lot of simple(ish) proofs
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Very complex, ongoing research

Short/Mid-Term Options (Existing Standards/Hardware)



BBS-ECDSA proposed by Ubique (in EUDI innovation challenge)

https://github.com/heidiverse/zkattest-rs (formalized in [LSZ25], analyzed in [LZ26])

- Based on zkAttest/CDLS [FLM21,CLR24]
- ECDSA dpk (on P-256) gets encoded as attribute in BBS ($\mathbb{Z}_q$ in BLS12-381)

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP	Efficiency (rough guestimates)	Maturity & Simplicity
BBS	BLS/Schnorr	Schnorr-type	~10 ms, 1kB	Simple, 20y of research
BBS BLS12-381	**ECDSA** P-256	**Schnorr-type** (a lot for ECDSA)	**~400 ms, ~175kB**	A lot of simple(ish) proofs
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Very complex, ongoing research

[FLM21] Faz-Hernández, Ladd, Maram. ZKAttest: Ring and Group Signatures for Existing ECDSA Keys. SAC 2021
[CLR24] Celi, Levin, Rowell. CDLS: Proving Knowledge of Committed Discrete Logarithms with Soundness. AFRICACRYPT 2024
[LSZ25] Lehmann, Sidorenko, Zacharakis. Vision: A Modular Framework for Anonymous Credential Systems. SSR25

Short/Mid-Term Options (Existing Standards/Hardware)

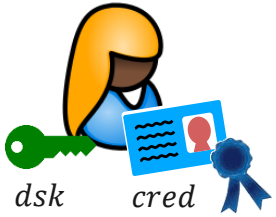
Both considered for EUDI Wallet „v2“ [TS13& TS14]

Currently undergoing standardization in ETSI

Issuer (Si				ity & licity
BBS	BLS/Schnorr	Schnorr-type	~10 ms, 1kB	Simple, 20y of research
BBS BLS12-381	ECDSA P-256	Schnorr-type (a lot for ECDSA)	~400 ms, ~175kB	A lot of simple(ish) proofs
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Very complex, ongoing research

https://github.com/eu-digital-identity-wallet/eudi-doc-standards-and-technical-specifications/blob/main/docs/technical-specifications/ts14-zkps-from-mms.md

Short/Mid-Term Options (Existing Standards/Hardware)



$\sigma_{PoP} \coloneqq Sign_U(dsk, nonce)$

User Attributes
User Key dpk

isk

$cred \coloneqq Sign_I(isk, (a_1, \ldots, a_\ell, dpk))$

ZKP of $cred$ under ipk and
of σ_{PoP} for certified dpk

Both considered for EUDI Wallet „v2“ [TS13& TS14]
Currently undergoing standardization in ETSI → main challenge: ZKP

Issuer (*Si*… …ity & …licity

BBS	BLS/Schnorr	Schnorr-type	~10 ms, 1kB	Simple, 20y of research
BBS BLS12-381	ECDSA P-256	Schnorr-type (a lot for ECDSA)	~400 ms, ~175kB	A lot of simple(ish) proofs
ECDSA	ECDSA	Circuit-based	~400 ms, ~300kB	Very complex, ongoing research

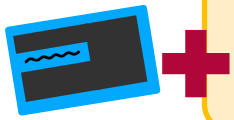
https://github.com/eu-digital-identity-wallet/eudi-doc-standards-and-technical-specifications/blob/main/docs/technical-specifications/ts14-zkps-from-mms.md

How to standardize the ZKP?

$cred \coloneqq Sign_I\big(isk, (a_1, \ldots, a_\ell, dpk)\big)$

$\sigma_{PoP} \coloneqq Sign_U(dsk, nonce)$

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP
BBS	ECDSA	Schnorr-type
ECDSA	ECDSA	Circuit-based



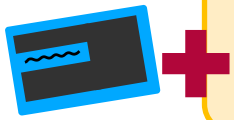
ZKP of $cred$ s.t. $Vf_I(ipk, cred, (attr, dpk)) = 1$ and
σ_{PoP}, dpk s.t. $Vf_U(dpk, \sigma_{PoP}, nonce) = 1$

How to standardize the ZKP?

$cred \coloneqq Sign_I(isk, (a_1, \ldots, a_\ell, dpk))$

$\sigma_{PoP} \coloneqq Sign_U(dsk, nonce)$

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP
BBS	ECDSA	Schnorr-type
ECDSA	ECDSA	Circuit-based



ZKP of $cred$ s.t. $Vf_I(ipk, cred, (attr, dpk)) = 1$ and σ_{PoP}, dpk s.t. $Vf_U(dpk, \sigma_{PoP}, nonce) = 1$

- This just does selective disclosure and device binding, but AC can do so much more:
  - Pseudonyms, e.g., $nym = PRF(pns, scope)$ with pns being secret attribute in $cred$
  - Range Proofs, e.g., $age > 18$, or $expiration > today$
  - Deniability, cross-credential proofs, issuer-hiding(!) [KS25,FFL26,FFK+26,KKLT26], ...

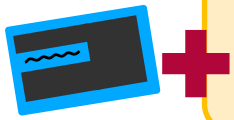
[KS25] Katz, Sefranek. Issuer Hiding for BBS-Based Anonymous Credentials
[FFL26] Flamini, Friedrichs, Lehmann. Issuer-Hiding for BBS Anonymous Credentials via Randomizable Keys
[FFK+26] Flamini et al. Issuer-Hiding BBS-Based Anonymous Credentials without Policy Keys
[KKLZ26] Kaaniche et al. Improved Issuer Hiding for BBS-based Anonymous Credentials

How to standardize the ZKP?

$cred \coloneqq Sign_I(isk, (a_1, \ldots, a_\ell, dpk))$

$\sigma_{PoP} \coloneqq Sign_U(dsk, nonce)$

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP
BBS	ECDSA	Schnorr-type
ECDSA	ECDSA	Circuit-based



ZKP of $cred$ s.t. $Vf_I(ipk, cred, (attr, dpk)) = 1$ and σ_{PoP}, dpk s.t. $Vf_U(dpk, \sigma_{PoP}, nonce) = 1$

- This just does selective disclosure and device binding, but AC can do so much more:
  - Pseudonyms, e.g., $nym = PRF(pns, scope)$ with pns being secret attribute in $cred$
  - Range Proofs, e.g., $age > 18$, or $expiration > today$
  - Deniability, cross-credential proofs, issuer-hiding(!)

Want to know more?
Andrea‘s talk 4pm @ PrivCrypt

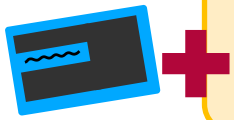
[KS25] Katz, Sefranek. Issuer Hiding for BBS-Based Anonymous Credentials
[FFL26] Flamini, Friedrichs, Lehmann. Issuer-Hiding for BBS Anonymous Credentials via Randomizable Keys
[FFK+26] Flamini et al. Issuer-Hiding BBS-Based Anonymous Credentials without Policy Keys
[KKLZ26] Kaaniche et al. Improved Issuer Hiding for BBS-based Anonymous Credentials

How to standardize the ZKP?

$cred \coloneqq Sign_I(isk, (a_1, \ldots, a_\ell, dpk))$

$\sigma_{PoP} \coloneqq Sign_U(dsk, nonce)$

Issuer ($Sign_I$)	Device ($Sign_U$)	ZKP
BBS	ECDSA	Schnorr-type
ECDSA	ECDSA	Circuit-based



ZKP of $cred$ s.t. $Vf_I(ipk, cred, (attr, dpk)) = 1$ and σ_{PoP}, dpk s.t. $Vf_U(dpk, \sigma_{PoP}, nonce) = 1$

- This just does selective disclosure and device binding, but AC can do so much more:
  - Pseudonyms, e.g., $nym = PRF(pns, scope)$ with pns being secret attribute in $cred$
  - Range Proofs, e.g., $age > 18$, or $expiration > today$
  - Deniability, cross-credential proofs, issuer-hiding(!)

Want to know more?
Andrea‘s talk 4pm @ PrivCrypt

- How to know now **what** will be needed? (ZKP „requirements“ in EUDI (TS4) very vague)
- Does every variant need a **new standard**? Even bigger challenge with monolithic circuits
- How to ensure **interoperability**, re-usability of code and security proofs?

[KS25] Katz, Sefranek. Issuer Hiding for BBS-Based Anonymous Credentials
[FFL26] Flamini, Friedrichs, Lehmann. Issuer-Hiding for BBS Anonymous Credentials via Randomizable Keys
[FFK+26] Flamini et al. Issuer-Hiding BBS-Based Anonymous Credentials without Policy Keys
[KKLZ26] Kaaniche et al. Improved Issuer Hiding for BBS-based Anonymous Credentials

(Some) Standards for BBS/PS-based Credentials

		Scheme	Nyms	Range	Device Bind	Blind Issue
ISO	Anonymous Digital Signatures ISO/IEC 20008-2:2013 / published in 2013(!)	CL/BBS/PS*	X		X	(X)
	Attribute-based Credentials ISO/IEC 24843 / just started	(BBS/PS/?)	?	?	?	?
ETSI	ZKP-EUDI-Wallet (BBS/ECDSA-Variant) ETSI TS 119 476-2 Work plan	BBS	(X)	X	X	(X)

+ cross-credential proofs, issuer-hiding, revocation(?)

*BBS only 1 attribute, old ZKP in Gt. PS signatures were added in 2023 update

(Some) Standards for BBS/PS-based Credentials

		Scheme	Nyms	Range	Device Bind	Blind Issue
ISO	Anonymous Digital Signatures ISO/IEC 20008-2:2013 / published in 2013(!)	CL/BBS/PS*	X		X	(X)
	Attribute-based Credentials ISO/IEC 24843 / just started	(BBS/PS/?)	?	?	?	?
IRTF CFRG draft-irtf-cfrg-pairing-friendly-curves/ also ZKP: draft-irtf-cfrg-sigma-protocols/ draft-irtf-cfrg-fiat-shamir/	BBS-Signatures draft-irtf-cfrg-bbs-signatures/	BBS				
	BBS per Verifier Linkability draft-irtf-cfrg-bbs-per-verifier-linkability/	BBS	X			(X)
	Blind BBS Signatures draft-irtf-cfrg-bbs-blind-signatures/	BBS				X
IETF PrivacyPass	Anonymous Rate-Limited Credentials draft-ietf-privacypass-arc-crypto/ (Apple/Cloudflare)	CMZ (~PS-MAC)	X	X		(X)
	Anonymous Credit Tokens draft-schlesinger-cfrg-act/ (Google)	BBS-MAC	X	(X)		(X)
ETSI	ZKP-EUDI-Wallet (BBS/ECDSA-Variant) ETSI TS 119 476-2 Work plan	BBS	(X)	X	X	(X)

+ cross-credential proofs, issuer-hiding, revocation(?)

*BBS only 1 attribute, old ZKP in Gt. PS signatures were added in 2023 update

(Some) Standards for BBS/PS-based Credentials

Currently: dedicated standard for every use case/feature combination
But: core primitives are the same/very similar → lots of work gets redone

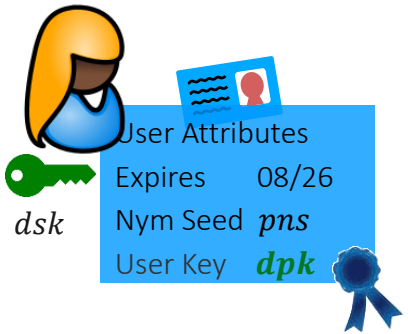
Slow process, e.g., IRTF BBS standardization since 2022!

		Scheme	Nyms	Range	Device Bind	Blind Issue
ISO	Anonymous Digital Signatures ISO/IEC 20008-2:2013 / published in 2013(!)	CL/BBS/PS*	X		X	(X)
	Attribute-based Creden ISO/IEC 24843 / just started	?	?	?	?	?
IRTF CFRG draft-irtf-cfrg-pairing-friendly-curves/ also ZKP: draft-irtf-cfrg-sigma-protocols/ draft-irtf-cfrg-fiat-shamir/	BBS-Signatures draft-irtf-cfrg-bbs-signatures/	BBS				
	BBS per Verifier Linkability draft-irtf-cfrg-bbs-per-verifier-linkability/	BBS	X			(X)
	Blind BBS Signatures draft-irtf-cfrg-bbs-blind-signatures/	BBS				X
IETF PrivacyPass	Anonymous Rate-Limited Credentials draft-ietf-privacypass-arc-crypto/ (Apple/Cloudflare)	CMZ (~PS-MAC)	X	X		(X)
	Anonymous Credit Tokens draft-schlesinger-cfrg-act/ (Google)	BBS-MAC	X	(X)		(X)
ETSI	ZKP-EUDI-Wallet (BBS/ECDSA-Variant) ETSI TS 119 476-2 Work plan	BBS	(X)	X	X	(X)

+ cross-credential proofs, issuer-hiding, revocation(?)

*BBS only 1 attribute, old ZKP in Gt. PS signatures were added in 2023 update

Modular Framework instead of Monolithic Standards

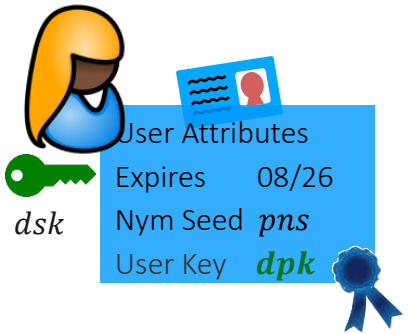


Status quo: monolithic ZKP for every use case

ZKP of $cred$ on hidden dpk, pns, exp
and of σ_{PoP} for certified dpk
and $nym = PRF(pns, scp)$ for certified pns
and valid, i.e., $exp > today$ for certified exp

Ignoring revealed/fully hidden attributes for simplicity here

Modular Framework instead of Monolithic Standards



Status quo: monolithic ZKP for every use case

ZKP of $cred$ on hidden dpk, pns, exp
and of σ_{PoP} for certified dpk
and $nym = PRF(pns, scp)$ for certified pns
and valid, i.e., $exp > today$ for certified exp

Ignoring revealed/fully hidden attributes for simplicity here

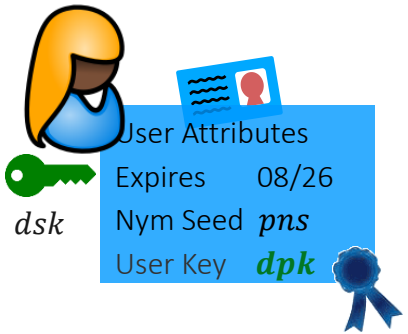
Vision: Modular Framework for Anonymous Credentials [LSZ25]

- Core module for $cred$

ZKP of $cred$ on dpk, pns, exp
committed in $C_{dpk}, C_{pns}, C_{exp}$

Core Module: Multi-Message Signature (or MAC) with **Commited Disclosure**

Modular Framework instead of Monolithic Standards



Status quo: monolithic ZKP for every use case

ZKP of $cred$ on hidden dpk, pns, exp
and of σ_{PoP} for certified dpk
and $nym = PRF(pns, scp)$ for certified pns
and valid, i.e., $exp > today$ for certified exp

Vision: Modular Framework for Anonymous Credentials [LSZ25]

- Core module for $cred$
- Extension Modules working on committed inputs

  Also the **ZKP is per module**!

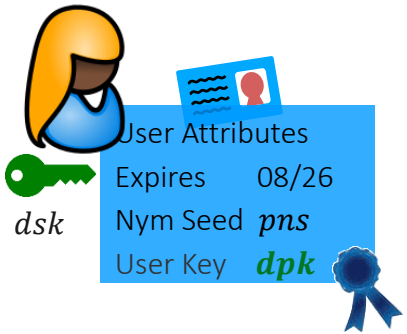
ZKP of $cred$ on dpk, pns, exp
committed in $C_{dpk}, C_{pns}, C_{exp}$

Core Module: Multi-Message Signature (or MAC) with **Commited Disclosure**

ZKP of σ_{PoP} for $dpk \in C_{dpk}$

PoP Module: explicit device binding

Modular Framework instead of Monolithic Standards



Status quo: monolithic ZKP for every use case

ZKP of $cred$ on hidden dpk, pns, exp
and of σ_{PoP} for certified dpk
and $nym = PRF(pns, scp)$ for certified pns
and valid, i.e., $exp > today$ for certified exp

Ignoring revealed/fully hidden attributes for simplicity here

Vision: Modular Framework for Anonymous Credentials [LSZ25]

- Core module for $cred$
- Extension Modules working on committed inputs
  Also the **ZKP is per module**!
- Anonymous Credential system: **plug-and-play composition** of core + needed modules

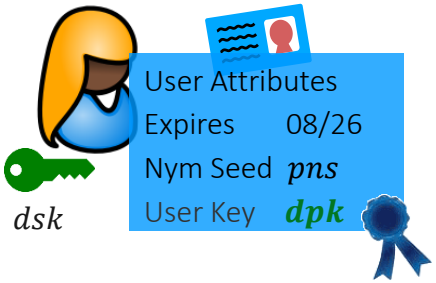
ZKP of $cred$ on dpk, pns, exp *committed in* $C_{dpk}, C_{pns}, C_{exp}$ — **Core Module**: Multi-Message Signature (or MAC) with **Commited Disclosure**

\+ ZKP of σ_{PoP} for $dpk \in C_{dpk}$ — **PoP Module**: explicit device binding

\+ ZKP of $nym = PRF(pns, scp)$ for $pns \in C_{pns}$ — **Pseudonym Module**

\+ ZKP of $exp > today$ for $exp \in C_{exp}$ — **Range Proof Module**

Modular Approach → Helpful for Device Binding too



ZKP of $cred$ on *committed* C_{dpk}

Core Module / Issuer Sig & ZKP

ZKP of σ_{PoP} for $dpk \in C_{dpk}$

PoP Module Device Sig & ZKP

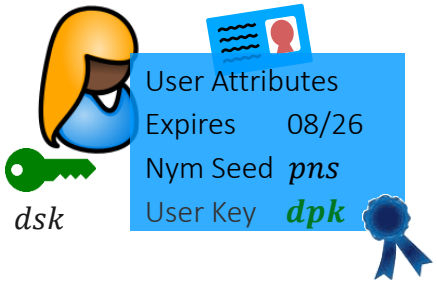
Choose optimal* signature **and proof system** per module

*What is „optimal“ also depends on use case & specific constraints

Issuer	ZKP
BBS	Schnorr

Device	ZKP
ECDSA	Schnorr

Modular Approach → Helpful for Device Binding too



ZKP of $cred$ on *committed* C_{dpk}

ZKP of σ_{PoP} for $dpk \in C_{dpk}$

Core Module / Issuer Sig & ZKP

PoP Module Device Sig & ZKP

Choose optimal* signature **and proof system** per module

*What is „optimal“ also depends on use case & specific constraints

Issuer	ZKP
BBS	Schnorr

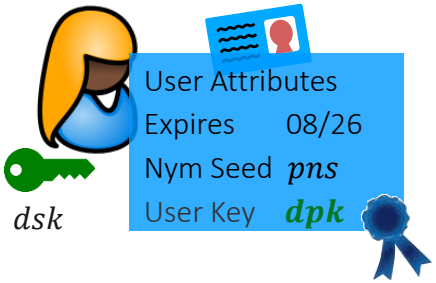
Device	ZKP
ECDSA	Schnorr
ECDSA	Circuit

BBS-ECDSA w. Circuit/Bulletproof
Proof size of 1.5KB! [LZ26]

Want to know more? Alexandros‘ talk today 4pm @ CAW

[LZ26] Lehmann, Zacharakis. Device Binding for Anonymous Credentials on Legacy Phones

Modular Approach → Helpful for Device Binding too



ZKP of $cred$ on *committed* C_{dpk}

Core Module / Issuer Sig & ZKP

ZKP of σ_{PoP} for $dpk \in C_{dpk}$

PoP Module Device Sig & ZKP

Choose optimal* signature **and proof system** per module

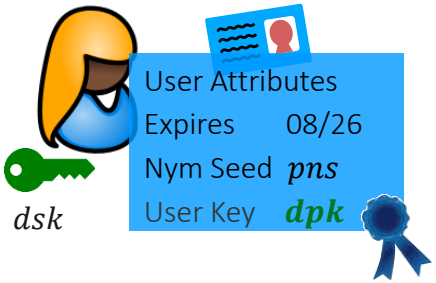
*What is „optimal“ also depends on use case & specific constraints

Issuer	ZKP
BBS	Schnorr
ECDSA	Circuit

Device	ZKP
ECDSA	Schnorr
ECDSA	Circuit

Tame complexity of circuit-based solutions

Modular Approach → Helpful for Device Binding too



ZKP of $cred$ on *committed* C_{dpk}

Core Module / Issuer Sig & ZKP

ZKP of σ_{PoP} for $dpk \in C_{dpk}$

PoP Module Device Sig & ZKP

Choose optimal* signature **and proof system** per module

*What is „optimal“ also depends on use case & specific constraints

Issuer	ZKP
BBS	Schnorr
ECDSA	Circuit
BLNS23	Lattice

Device	ZKP
ECDSA	Schnorr
ECDSA	Circuit
ML-DSA	Circuit

Crypto agility! PQC-Classic → PQC-PQC

Post-Quantum vs. Post-Privacy?

- Does it make sense to deploy new DL-based crypto? PQC!
  PQC less time-critical for authentication than for encryption

- **Main priority: solution should have post-quantum privacy!**
  Anonymous Credentials typically have perfect privacy, but DL-based soundness

Post-Quantum vs. Post-Privacy?

- Does it make sense to deploy new DL-based crypto? PQC!
  PQC less time-critical for authentication than for encryption

- **Main priority: solution should have post-quantum privacy!**
  Anonymous Credentials typically have perfect privacy, but DL-based soundness

- Advantage of pairing-based credential schemes → efficiency & simplicity
  Use cases with strict efficiency requirements and/or low risk
  e.g., age verification; protected value < cost of quantum attack

Post-Quantum vs. Post-Privacy?

- Does it make sense to deploy new DL-based crypto? PQC!
  PQC less time-critical for authentication than for encryption

- **Main priority: solution should have post-quantum privacy!**
  Anonymous Credentials typically have perfect privacy, but DL-based soundness

- Advantage of pairing-based credential schemes → efficiency & simplicity
  Use cases with strict efficiency requirements and/or low risk
  e.g., age verification; protected value < cost of quantum attack

- Identity infrastructure is being built now! Based on „ECDSA mindset“
  If we don’t propose a viable ZKP-based solution now → lack of privacy will manifest

AC-EUDI Roadmap & Open Challenges

Short/Midterm (PQC-privacy, classic soundness)

- Show feasibility & shape sensible (!) requirements
  Currently: same solution & requirements for everything
- ZKP-compatible protocols (OIDC4Vx) & data formats

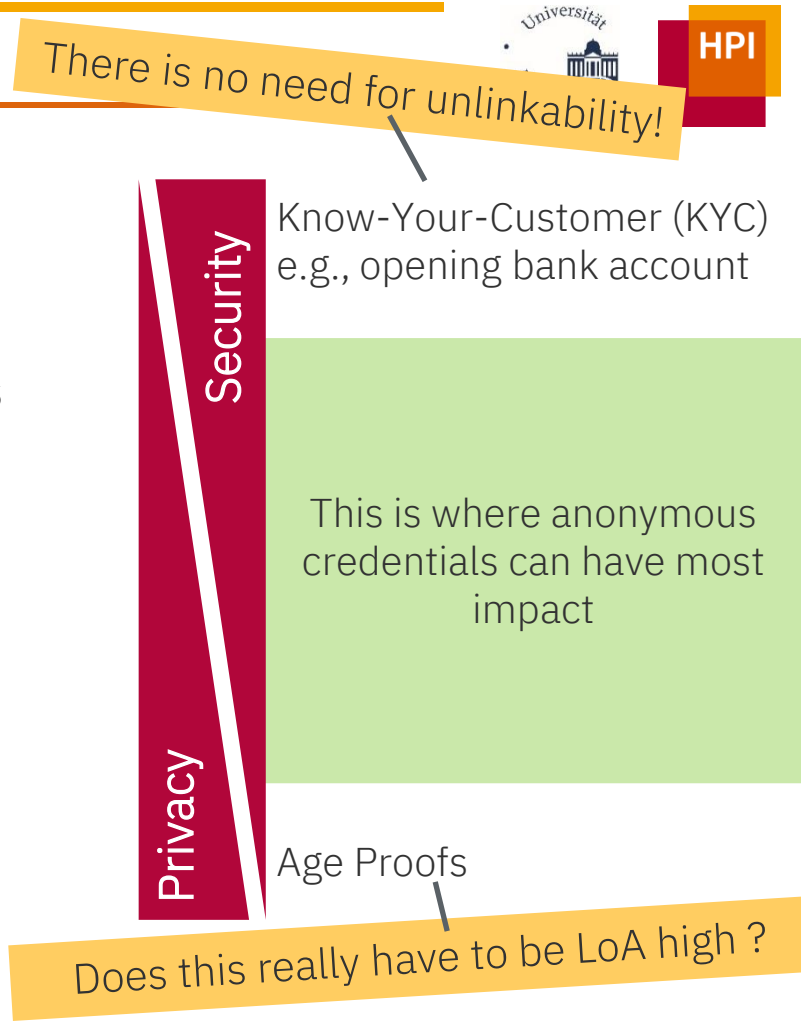
Know-Your-Customer (KYC)
e.g., opening bank account

Age Proofs

AC-EUDI Roadmap & Open Challenges

Short/Midterm (PQC-privacy, classic soundness)

- Show feasibility & shape sensible (!) requirements
  Currently: same solution & requirements for everything
- ZKP-compatible protocols (OIDC4Vx) & data formats



AC-EUDI Roadmap & Open Challenges

There is no need for unlinkability!

Short/Midterm (PQC-privacy, classic soundness)

- Show feasibility & shape sensible (!) requirements
  Currently: same solution & requirements for everything
- ZKP-compatible protocols (OIDC4Vx) & data formats

Longterm (PQC-based)

- Current work provides concrete target for PQC research
  Insights from pre-PQC serve as blueprint
- Shape PQC base standards & hardware APIs now(ish)
  ZKP-friendly version of ML-DSA or FN-DSA?
  E.g. Device-binding trick from [FHLL26] will not work if fixed dpk is hashed

Summary

- Real-world adoption of advanced cryptography requires **standards**
  Get involved in IETF, ETSI, ISO & EUDI Wallet projects
  Shape PQC base standards nowish

Summary

- Real-world adoption of advanced cryptography requires **standards**
  Get involved in IETF, ETSI, ISO & EUDI Wallet projects
  Shape PQC base standards nowish
- Build schemes that are compatible with real-world & **legacy constraints**
- Anonymous credentials with device binding have *two* signature schemes
  → Choose optimal scheme for *each:* their requirements & constraints differ
- **Modular** constructions facilitate analysis, design & standards

Summary

- Real-world adoption of advanced cryptography requires **standards**
  Get involved in IETF, ETSI, ISO & EUDI Wallet projects
  Shape PQC base standards nowish
- Build schemes that are compatible with real-world & **legacy constraints**
- Anonymous credentials with device binding have *two* signature schemes
  → Choose optimal scheme for *each:* their requirements & constraints differ
- **Modular** constructions facilitate analysis, design & standards

The moral character of our work (Phil Rogaway)

Cryptography rearranges power: it configures who can do what, from what.
This makes cryptography an inherently political tool.

Deploying a digital identity system at scale will rearrange a lot of power