Anonymous Credentials on Legacy Phones

Alexandros Zacharakis

Hasso Plattner Institut and University of Potsdam

May 10, 2026

Based on joint works with Andrea Flamini, Christian Mouchet, Anja Lehmann, Andrey Sidorenko

EUropean Digital Identity Wallet

(Aspiring to be) *privacy centric:*

- selective disclosure
- unlinkability
- pseudonyms

EU regulation eIDAS 2.0

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 [..]

EUropean Digital Identity Wallet

The good,

- We know how!
- Anonymous Credentials
- Zero Knowledge Proofs

EUropean Digital Identity Wallet

The good,

- We know how!
- Anonymous Credentials
- Zero Knowledge Proofs

the bad,

- More *complex* than “conventional” cryptography
- Never* *deployed “en masse”*

EUropean Digital Identity Wallet

The good,

- We know how!
- Anonymous Credentials
- Zero Knowledge Proofs

the bad,

- More *complex* than “conventional” cryptography
- Never* *deployed “en masse”*

... and the ugly

- *Standardization* and the *Standards Zoo*
- Existing *hardware limitations*

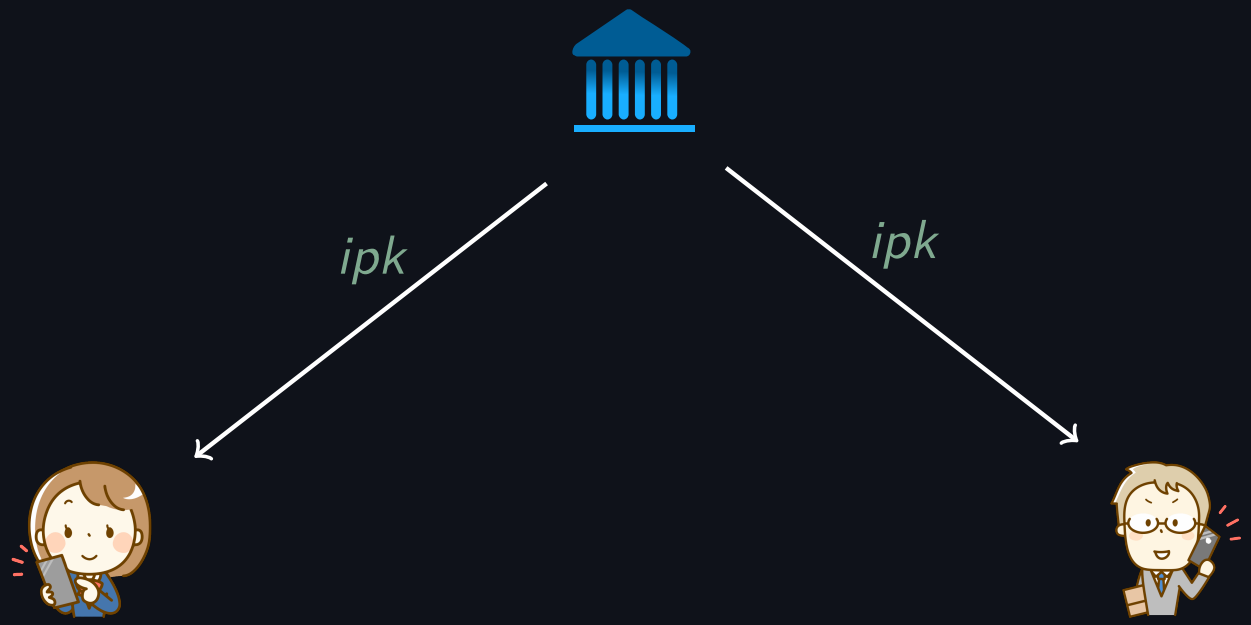
A Practical Barrier: Credential Cloning

- Digital nature $\Rightarrow$ credential cloning / transferability
- Difficult to prevent

A Practical Barrier: Credential Cloning

- Digital nature $\Rightarrow$ credential cloning / transferability
- Difficult to prevent
- *Idea*: credential "bound" to a device $\Rightarrow$ *Proof-of-Possession*
  - *Secure Element*: (inaccessible) dsk
  - dpk in the credential
  - User: *oracle access* to $\mathsf{Sign}(\mathsf{dsk}, \cdot)$
  - During presentation: σ_{PoP} on a challenge *nonce*

A (basic) AC Blueprint: Signature Scheme + ZKP

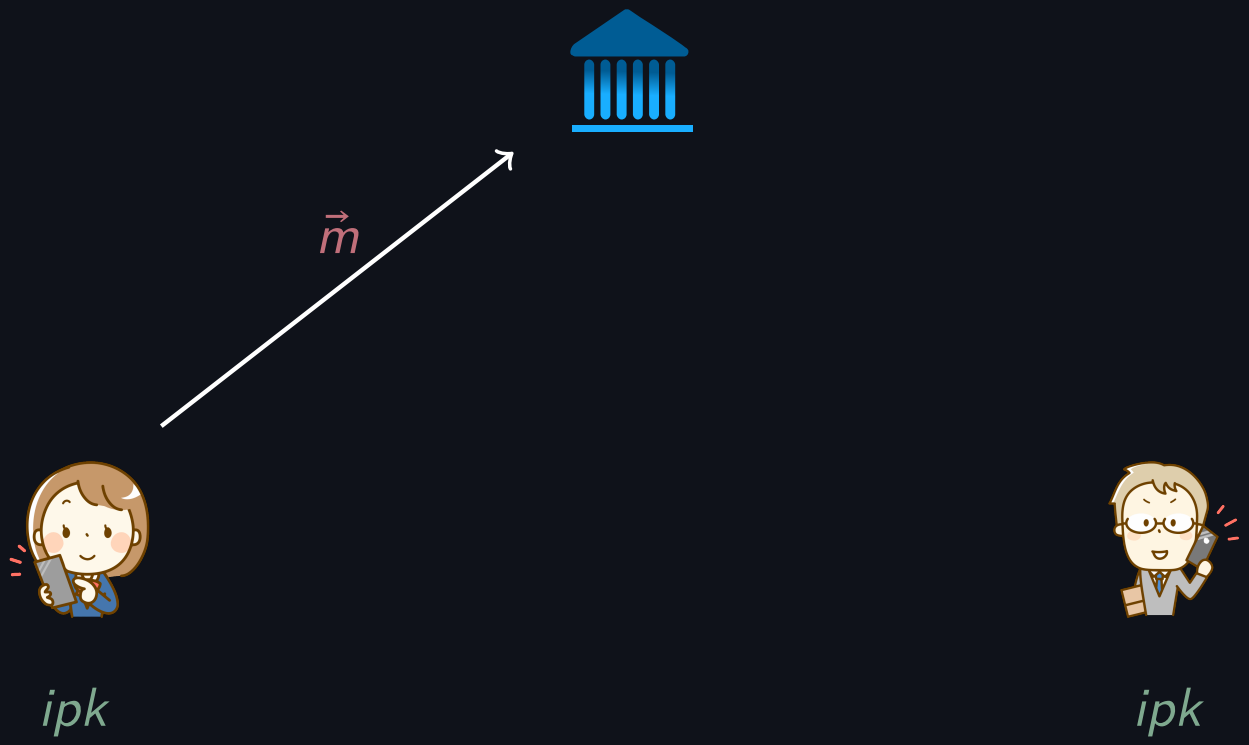


A (basic) AC Blueprint: Signature Scheme + ZKP

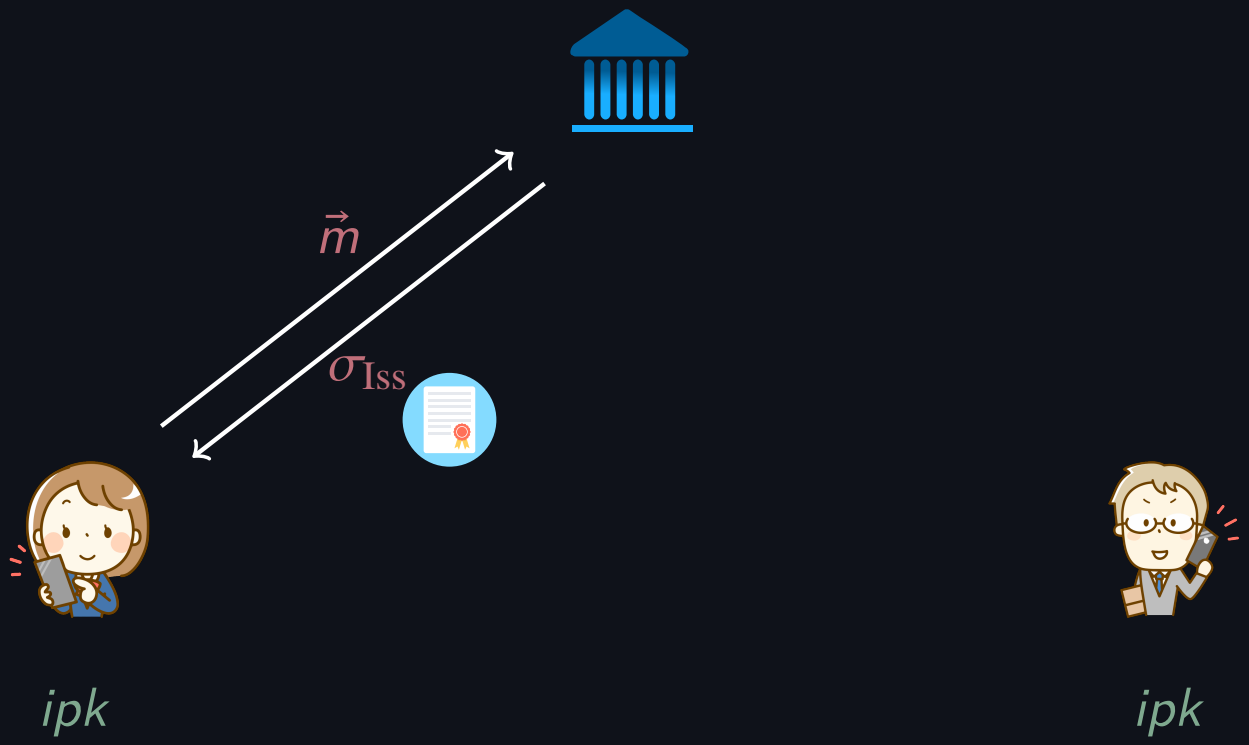
ipk

ipk

A (basic) AC Blueprint: Signature Scheme + ZKP



A (basic) AC Blueprint: Signature Scheme + ZKP

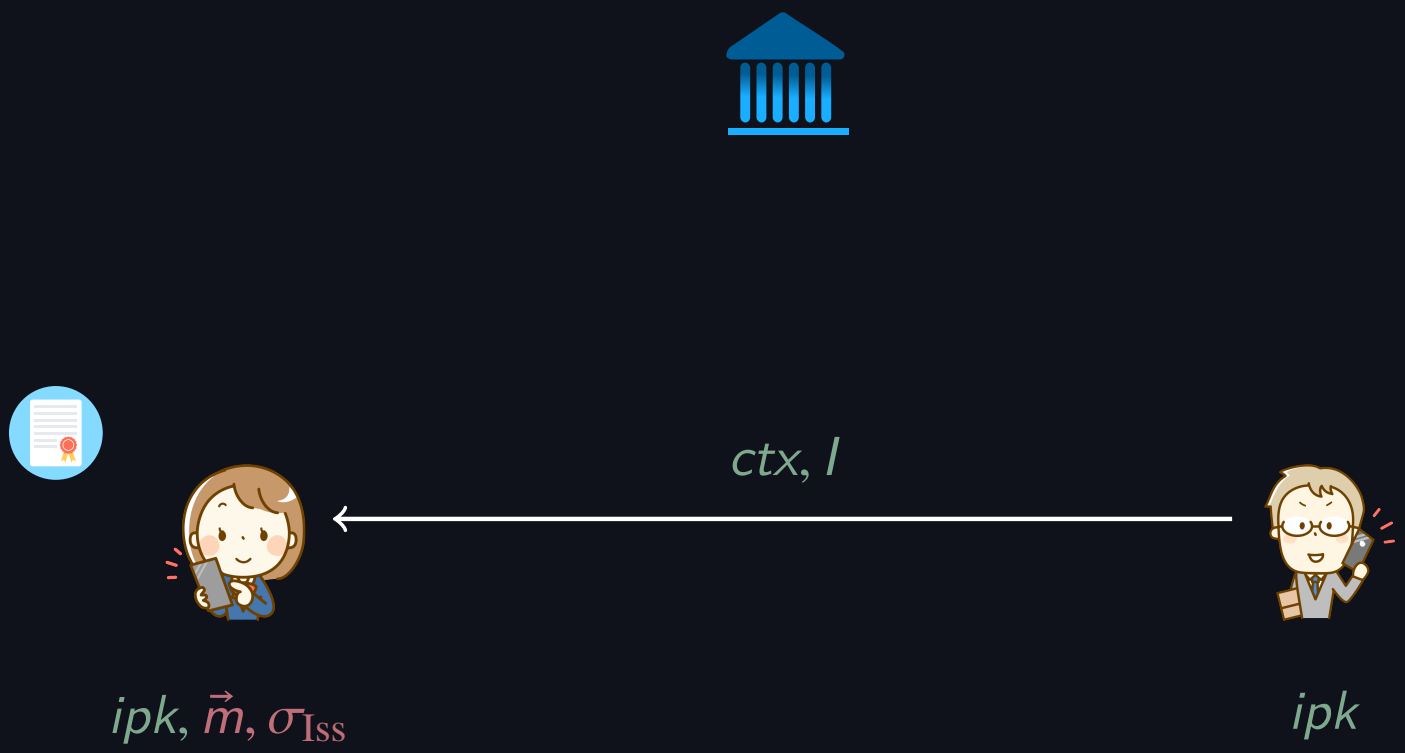


A (basic) AC Blueprint: Signature Scheme + ZKP

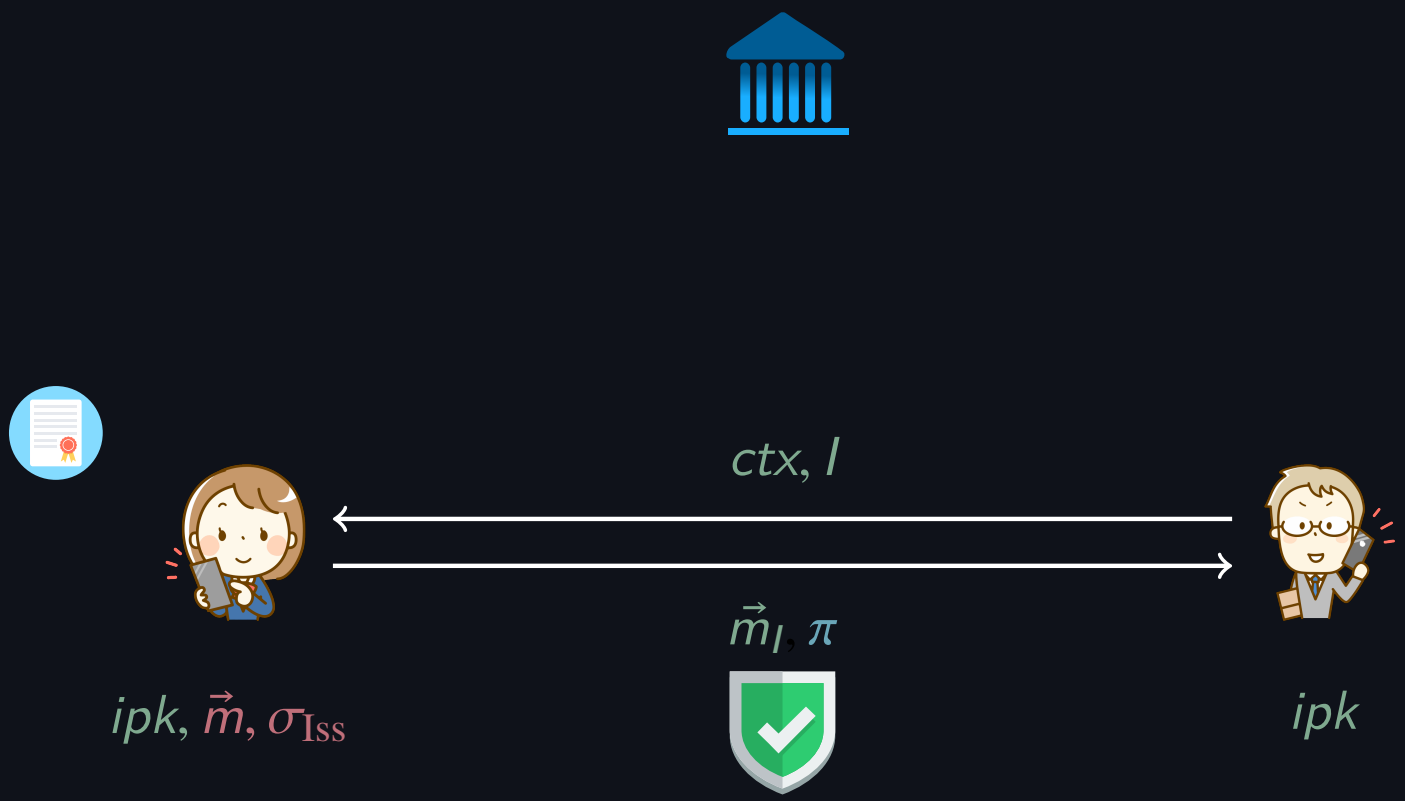
$ipk, \vec{m}, \sigma_{\text{Iss}}$

ipk

A (basic) AC Blueprint: Signature Scheme + ZKP

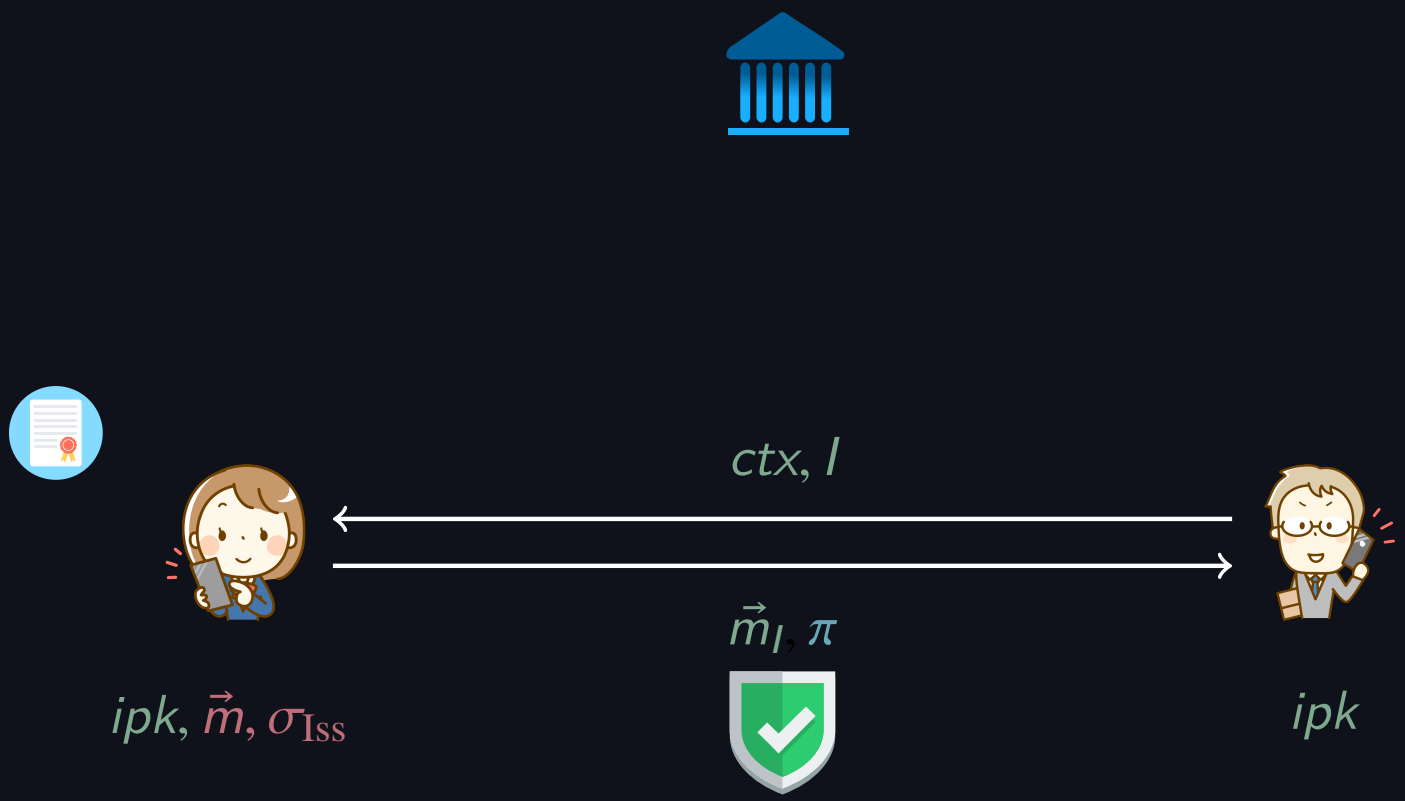


A (basic) AC Blueprint: Signature Scheme + ZKP



π : σ_{Iss} is valid signature of $\vec{m}$ under *ipk* and "consistent" with $\vec{m}_I$

A (basic) AC Blueprint: Signature Scheme + ZKP



π : σ_{Iss} is valid signature of $\vec{m}$ under *ipk* and "consistent" with $\vec{m}_I$

A More Realistic Scenario...

More functionality:

- Predicate Proofs
- Non-Revocation
- Pseudonyms
- ***Device Binding***
- …

A More Realistic Scenario...

More functionality:

- Predicate Proofs
- Non-Revocation
- Pseudonyms
- ***Device Binding***
- …

Issue!

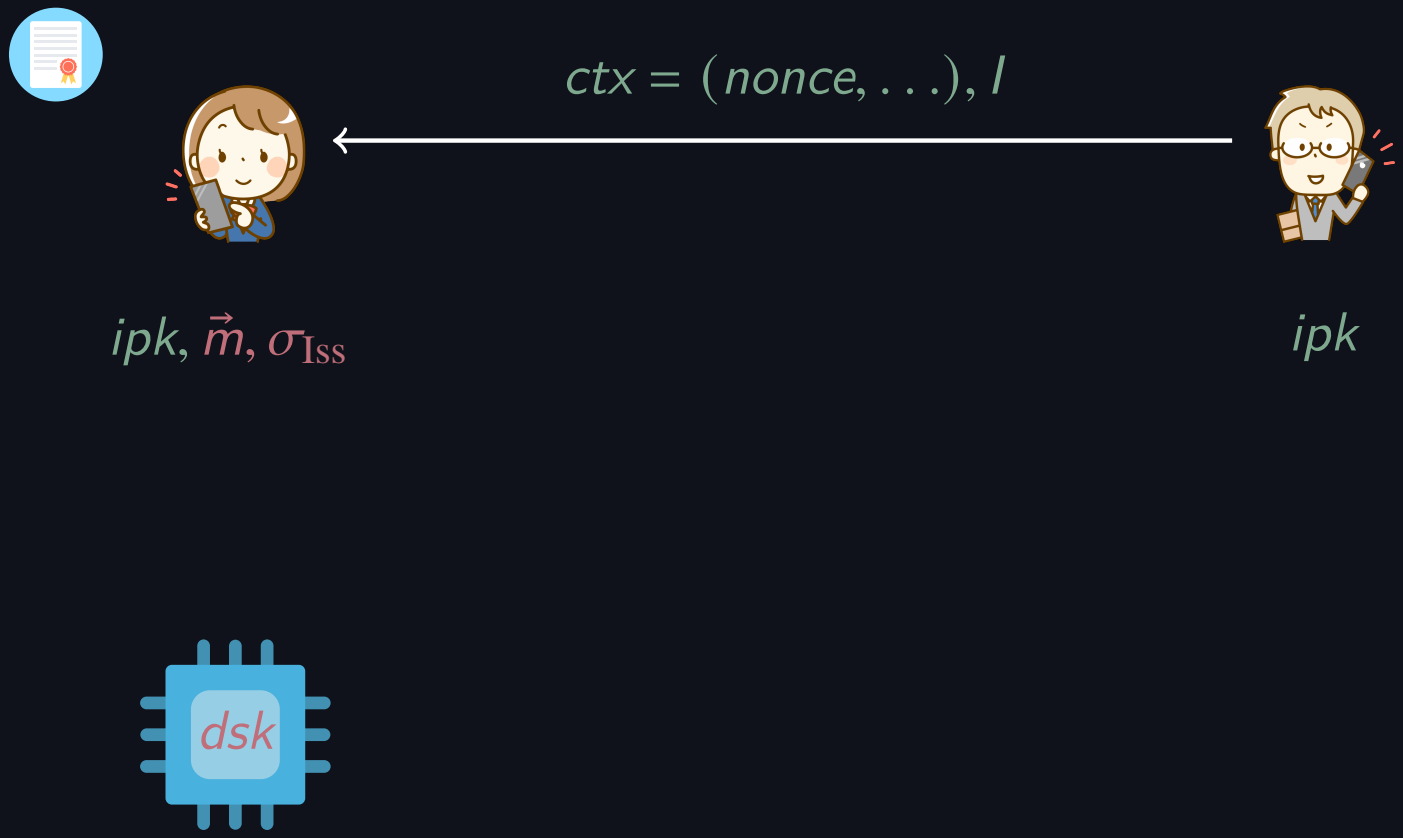
The ZKP becomes *too complex*!

AC + Device Binding

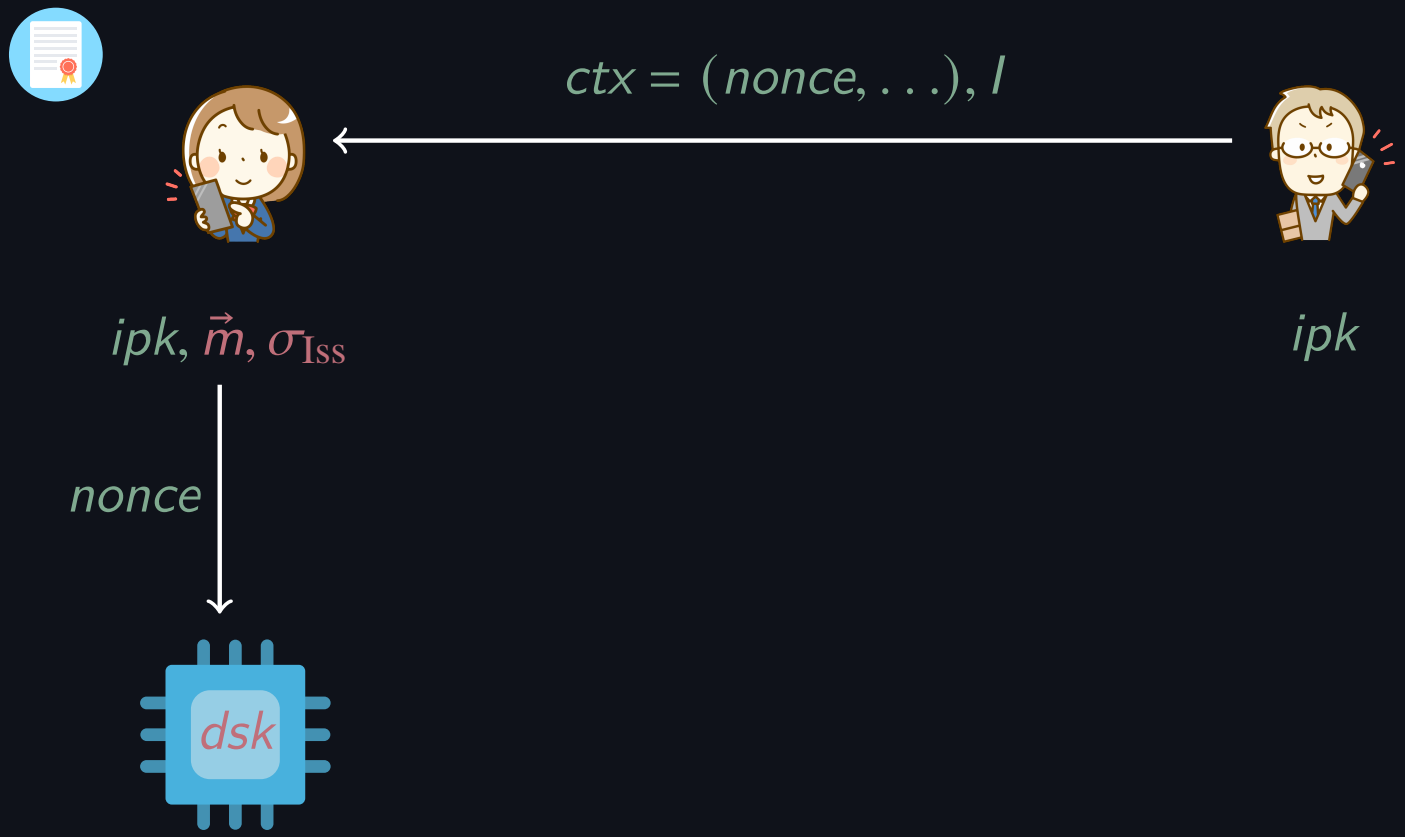
$ipk, \vec{m}, \sigma_{\text{Iss}}$

ipk

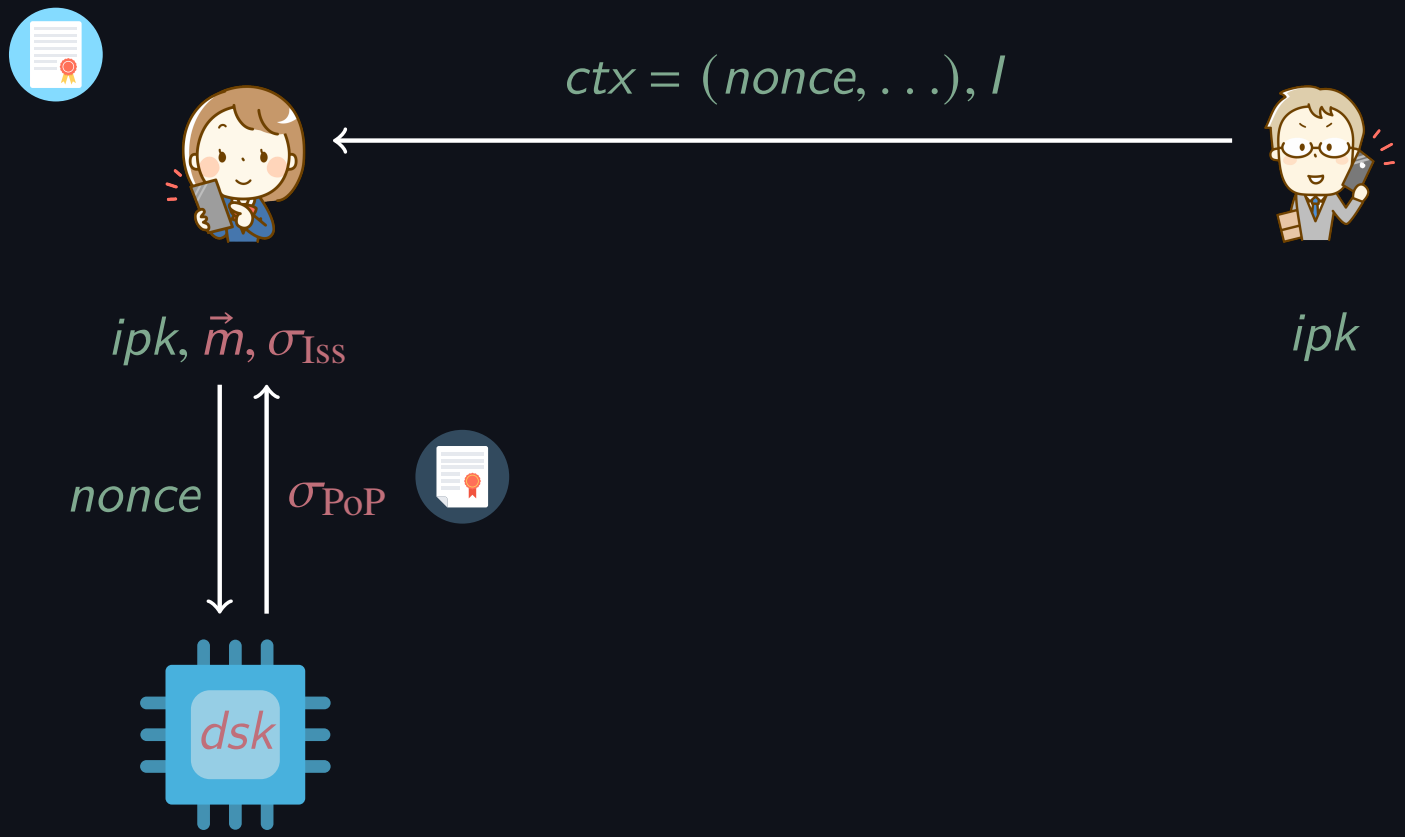
AC + Device Binding



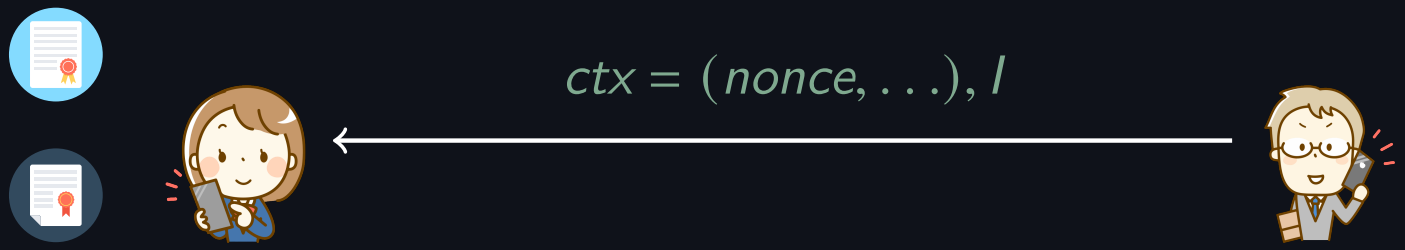
AC + Device Binding



AC + Device Binding



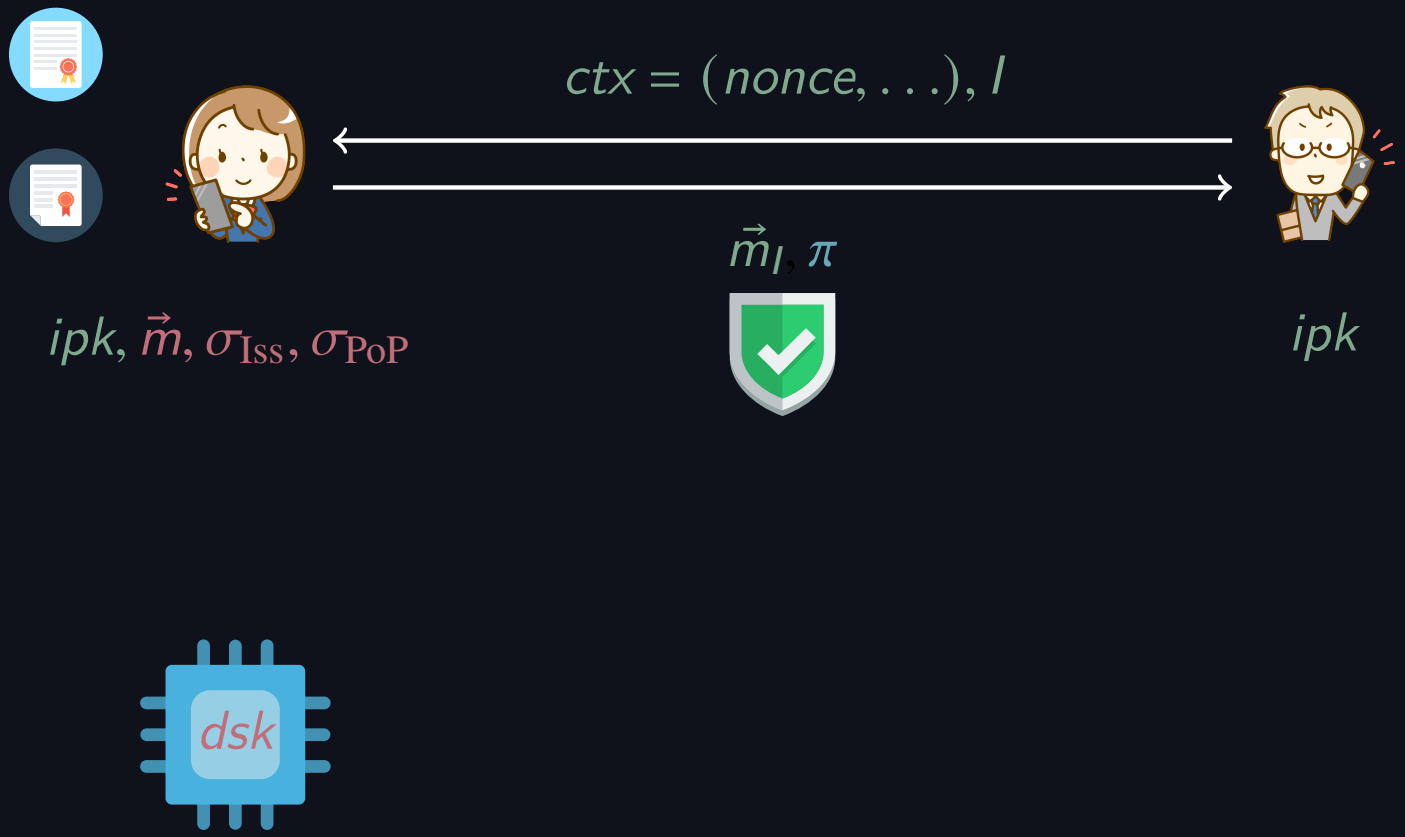
AC + Device Binding



$ipk, \vec{m}, \sigma_{\text{Iss}}, \sigma_{\text{PoP}}$

ipk

AC + Device Binding



Device Binding: An *Ugly* Reality

Question

What signature scheme to use?

Device Binding: An *Ugly* Reality

Question

What signature scheme to use?

Answer

Let's use an:

- Efficient ✔
- Super Secure / Post quantum guarantees ✔
- Native ZKPs for knowledge of a signature ✔
- Fancy niche features X, Y, Z ✔

Device Binding: An *Ugly* Reality

Question

What signature scheme to use?

Answer

Let's use an:

- Efficient ✔
- Super Secure / Post quantum guarantees ✔
- Native ZKPs for knowledge of a signature ✔
- Fancy niche features X, Y, Z ✔

or . . . let's use ECDSA (?!)

Device Binding: An *Ugly* Reality

- Most common devices $\Rightarrow$ *only ECDSA over P256*

Device Binding: An *Ugly* Reality

- Most common devices $\Rightarrow$ *only ECDSA over P256*
- Hard to change:
  - Building SEs is *hard*, *timely* and *costly*
  - Other choices would be *exclusive* and *unfair*

Device Binding: An *Ugly* Reality

- Most common devices $\Rightarrow$ *only ECDSA over P256*
- Hard to change:
  - Building SEs is *hard*, *timely* and *costly*
  - Other choices would be *exclusive* and *unfair*

ECDSA PoPs

ECDSA was *not* designed with PoPs in mind. Must use *generic, non-native ZKP techniques* which:

1. Add *complexity*
2. Incur *efficiency costs*

Monolithic Approach

Issuer	SE	ZKP	$t_{\mathcal{P}}$	$\lvert\pi\rvert$
ECDSA	ECDSA	$\mathcal{C}$	90ms - 500ms	90KB - 300KB

Table 1: Adapted from **[Vega]** and **[OpenAC]**. Run on 16vCPUs with 64GB RAM

Monolithic Approach

Issuer	SE	ZKP	$t_{\mathcal{P}}$	$\lvert\pi\rvert$
ECDSA	ECDSA	$\mathcal{C}$	90ms - 500ms	90KB - 300KB

Table 1: Adapted from **[Vega]** and **[OpenAC]**. Run on 16vCPUs with 64GB RAM

Pros:

- Compatible with existing credentials ✔

Monolithic Approach

Issuer	SE	ZKP	$t_{\mathcal{P}}$	$\lvert\pi\rvert$
ECDSA	ECDSA	C	90ms - 500ms	90KB - 300KB

Table 1: Adapted from **[Vega]** and **[OpenAC]**. Run on 16vCPUs with 64GB RAM

Pros:

- Compatible with existing credentials ✔

Cons:

- (Very) complex ZKP systems ✗
- (Very) complex circuits (ECDSA, SHA256, parsing) ✗
- Inherently large proofs ($t_{\mathcal{P}}$ vs $\lvert\pi\rvert$) ✗

Monolithic Approach

Issuer	SE	ZKP	$t_{\mathcal{P}}$	$\lvert\pi\rvert$
ECDSA	ECDSA	C	90ms - 500ms	90KB - 300KB

Table 1: Adapted from **[Vega]** and **[OpenAC]**. Run on 16vCPUs with 64GB RAM

Pros:

- Compatible with existing credentials ✔

Cons:

- (Very) complex ZKP systems ✗
- (Very) complex circuits (ECDSA, SHA256, parsing) ✗
- Inherently large proofs ($t_{\mathcal{P}}$ vs $\lvert\pi\rvert$) ✗

Idea

Let's ***minimize ECDSA use*** to the absolutely necessary!

A Modular Framework [LSZ 25]

Core idea:

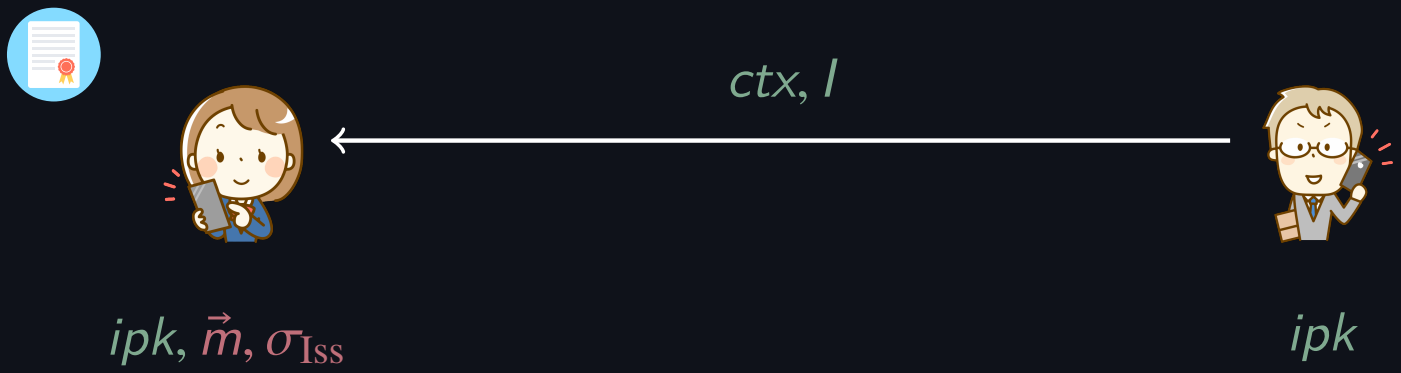
- Divide & Conquer apporach
- *Small, flexible* building blocks vs of *monolithic, complex* construction
- signature scheme + ZKP modules + commitment linking

A Modular Framework [LSZ 25]

$ipk, \vec{m}, \sigma_{\text{Iss}}$

ipk

A Modular Framework [LSZ 25]



A Modular Framework [LSZ 25]

$ipk, \vec{m}, \sigma_{\text{Iss}}, \sigma_{\text{PoP}}$

ipk

A Modular Framework [LSZ 25]

$ipk, \vec{m}, \sigma_{\text{Iss}}, \sigma_{\text{PoP}}$

m_4 C_4

m_5 C_5

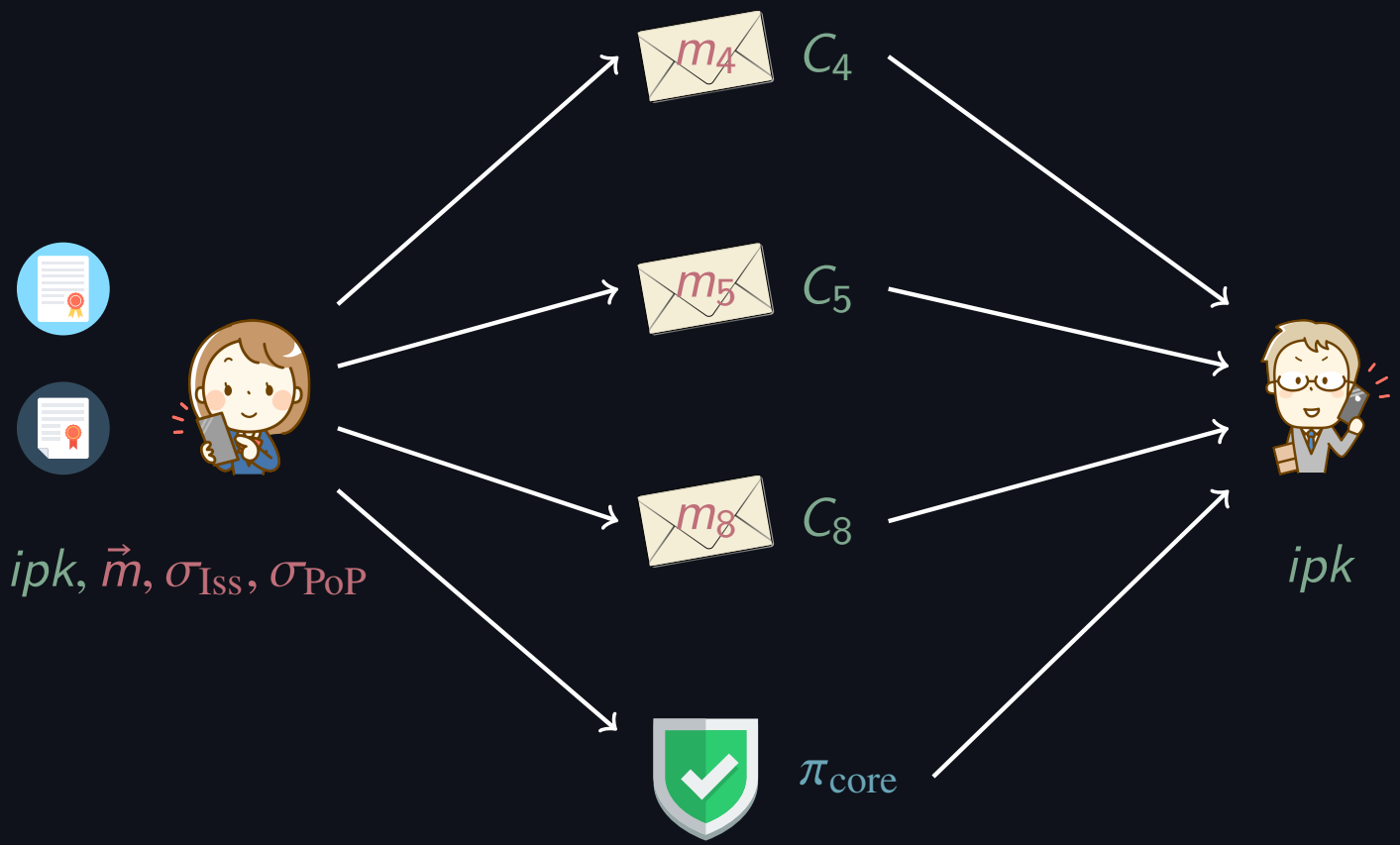
m_8 C_8

ipk

π_{core}

$\pi_{\text{core}} : \sigma_{\text{Iss}}$ is valid signature of $\vec{m}$ under ipk, "consistent" with $\vec{m}_I$ ***and*** C_4, C_5, C_8

A Modular Framework [LSZ 25]



π_{core} : σ_{Iss} is valid signature of $\vec{m}$ under ipk, "consistent" with $\vec{m}_I$ ***and*** C_4, C_5, C_8

A Modular Framework [LSZ 25]

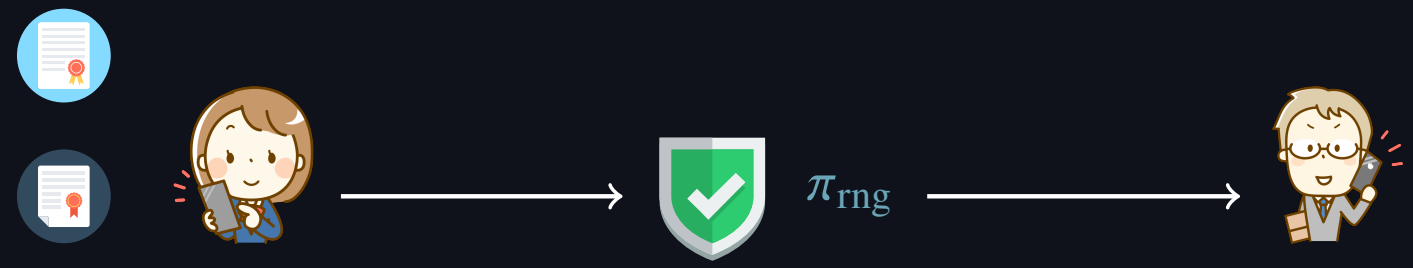
$ipk, \vec{m}, \sigma_{\text{Iss}}, \sigma_{\text{PoP}}$

π_{rng}

ipk

π_{rng} : m_4 in C_4 is in range $[a, b)$

A Modular Framework [LSZ 25]



π_{rng} : m_4 in C_4 is in range $[a, b)$

A Modular Framework [LSZ 25]

$ipk, \vec{m}, \sigma_{\text{Iss}}, \sigma_{\text{PoP}}$

π_{pop}

ipk

π_{pop} : m_8 in C_8 corresponds to a *device public key* dpk and Alice has access to the device

A Modular Framework [LSZ 25]

π_{pop} : m_8 in C_8 corresponds to a *device public key* dpk and Alice has access to the device

A Modular Framework [LSZ 25]

- Simple
- Different ZKP for each task:
  - MMS and *native* ZKP for *Core* $\Rightarrow$ *Ultra efficient!*
  - Speialized proofs/different tradeoffs for other "modules"
- Crypto Agility $\Rightarrow$ Easier *PQ transition*
- Trade-offs
- "Incremental" Standardization

Choosing the building blocks

- **Issuer:** Pairing-based + *Native* ZKP (e.g. BBS) ✔
- **Linking:** Pedersen Commitments ✔
- **SE:** ECDSA ✗
- π_{PoP}: ???

ECDSA Based PoPs [LZ 26]

Various ZKPs with *different trade-offs:*

- Efficiency
- System complexity
- Assumptions

Modularity:

- Compatible with **[LSZ 25]**
- ZKPs themselves are modular, **[KP 23]** framework

ECDSA Based PoPs [LZ 26]

π_{PoP}: *"I know* σ_{PoP} *on* nonce *under committed* dpk

Simplified statement

- Only signing one-time (short lived) messages
- Part of the signature can be revealed[1]

The problem essentially reduces to $zK = Q$ over P256 with Q committed.

[1] Terms and conditions apply: EC paper **[KHLL 26]**.

Core Challenge

$C_x = xG + r_1 H$

$C_y = yG + r_2 H$

in BLS12

$zK = Q$

$Q = (x, y)$

in P256

ECDSA Based PoPs [LZ 26]

ECDSA Based PoPs [LZ 26]

1. PoP-PLONK: Circuit over BLS12 + FFA
   - Succinct
   - Relying on BLS12
   - Heavy toolkit (but can possibly be simplified)

ECDSA Based PoPs [LZ 26]

1. PoP-PLONK: Circuit over BLS12 + FFA
   - Succinct
   - Relying on BLS12
   - Heavy toolkit (but can possibly be simplified)
2. PoP-BP: (Native) Circuit + commitment linking
   - (Very) Succinct + Fast
   - Simple Proving System + Circuit
   - Relying on “not-standard” curve

ECDSA Based PoPs [LZ 26]

1. PoP-PLONK: Circuit over BLS12 + FFA
   - Succinct
   - Relying on BLS12
   - Heavy toolkit (but can possibly be simplified)
2. PoP-BP: (Native) Circuit + commitment linking
   - (Very) Succinct + Fast
   - Simple Proving System + Circuit
   - Relying on “not-standard” curve
3. PoP-Σ: Σ-based + commitment linking
   - (Very) Simple
   - Relying on “not-standard” curve
   - Large Proofs

ECDSA Based PoPs [LZ 26]

Issuer	SE	ZKP	$t_{\mathcal{P}}$	$\lvert\pi\rvert$
ECDSA	ECDSA	C	90ms - 500ms	90KB - 300KB

Table 1: Adapted from **[Vega]** and **[OpenAC]**. Run on 16vCPUs with 64GB RAM

	Issuer	SE	ZKP	$t_{\mathcal{P}}$	$\lvert\pi\rvert$
PoP-PLONK	BBS	ECDSA	$\Sigma + C$	2.6s	~ 4KB
PoP-BP	BBS	ECDSA	$\Sigma + C$	350ms	1.5KB
PoP-Σ	BBS	ECDSA	Σ	350ms	172KB

Table 2: Run on commodity laptop (i5-1345U, 16GB RAM). Only considers π_{PoP}.

PoP-BP

Concluding Remarks

Modularity:

- Simpler constructions
- More trade-offs
- Easier Standardization
- Crypto agility

Concluding Remarks

Modularity:

- Simpler constructions
- More trade-offs
- Easier Standardization
- Crypto agility

Device binding:

- ECDSA (for now)
- Many ways to realize
- Efficiecny vs Complexity vs Assumptions

Concluding Remarks

Modularity:

- Simpler constructions
- More trade-offs
- Easier Standardization
- Crypto agility

Device binding:

- ECDSA (for now)
- Many ways to realize
- Efficiecny vs Complexity vs Assumptions

Societal impact

We discussed from the "cryptographers" lens. Digital identity will *rearrange power* and will have a deep societal impact.

Bringing AC and ZKP to EUDI

- Contributions to Standarization
  - TS 14
  - ETSI
  - IETF
- [LSZ 25]
  - reduce complexity
  - simplify standarization
- [LZ 26]
  - reduce complexity
  - address hardware limitaions
  - simplify standarization
- . . .

Based on joint works with Andrea Flamini,, Christian Mouchet, Anja Lehmann, Andrey Sidorenko

[LSZ 25]

[LZ 26]

Thank you!