E-ID Hands-on Workshop

Keeping identities safe and sound

Center for Digital Trust

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Agenda

9:15am - welcome coffee

9:30am - Overview of E-ID Landscape - Dr. Imad Aad

11:30am - lunch

1:00pm - Hands-on training, led by C4DT:

- Starting with signed verified credentials and selective disclosure
- Using BBS+ to add unlinkability
- Introducing zero-knowledge proofs to reduce information leakage

4:30pm - Wrap-up: Lessons learned and how to go forward

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

C4DT - TLDR

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
armasuisse

bevvs
The wise side of data

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Bundesamt für Informatik und Telekommunikation BIT

CICR

ELCA

17 96
LOMBARD ODIER
LOMBARD ODIER DARIER HENTSCH

Microsoft

NAGRA
KUDELSKI GROUP

Roche

SICPA

SWISS POST

swisscom

Swissquote

EPFL
Center for
Digital Trust

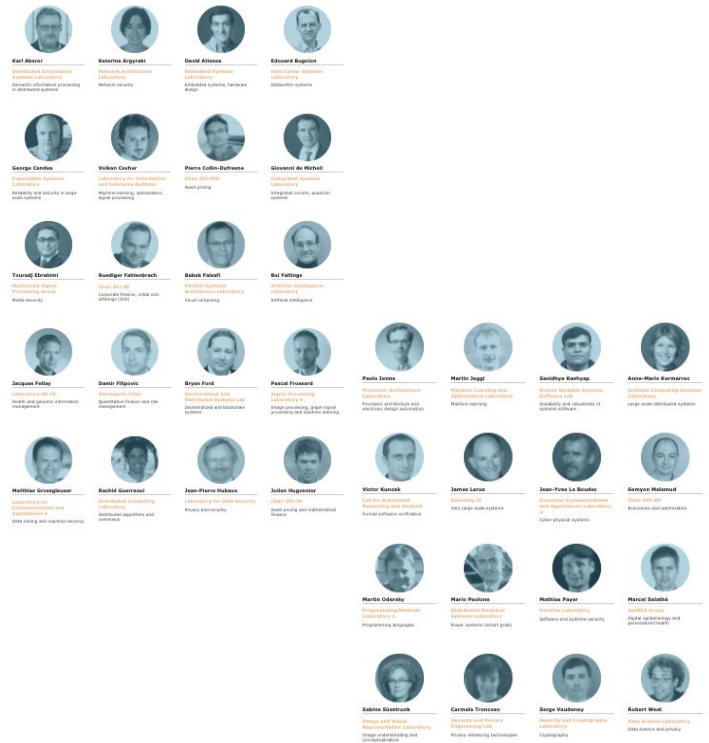
Associated Partners

csky.ai

egonym

GeneLook

SAPORO
ORDER IN CHAOS



© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

5 Domains

Interface	Community	Authorities	Proof of Concepts	Public Service
Education	Events	Initiatives	Hands-on workshops	GovTech
Experts	Workgroups	International	Library reviews	

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Team of 15 (+2) People


Sandra Hünsch


Valérie Meillaud


Jean-Pierre Hubaux


Olivier Crochat

ACADEMY
TRAINING

AGENCY
PROJECTS

EMBASSY
COMMUNITY

FACTORY
SOFTWARE

POLICY
Governance


Stéphanie Milliquet


Imad Aad


Linus Gasser


Katherine Loh


Melanie Kolbe-Guyot


Alaeddine El Fawal


David


Carine


Ahmed


Matthias


Paola Daniore

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

C4DT Factory - Overview

INCUBATOR

- Project Presentation
- Paper -> Real world
- Demonstrators / Market

KNOWLEDGE

- Explore subjects
- Article / Blog posts
- Conferences / Teaching

HANDS-ON WORKSHOPS

- 1-day trainings
- Share latest research
- Real-world input

COMMUNITY

- Research Software Engineers
- EPFL Labs - Factory - Partners
- Conferences

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

C4DT Factory in October / November 2024

[Curtain Call for our Demonstrators]

C4DT Factory Update
2024/10
Nov. 1st

[Deepfake round-table and workshop]
Nov. 19th & 26th

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Overview of E-ID Landscape

Dr. Imad Aad, C4DT

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

E-ID Hands-on Workshop

Keeping identities safe and sound

Center for Digital Trust

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Program

1. Signing simply with RSA
2. Unlinkable proofs using BBS+
3. Predicate proofs with ZKPs
4. ZKP Considerations

For subjects 1-3:

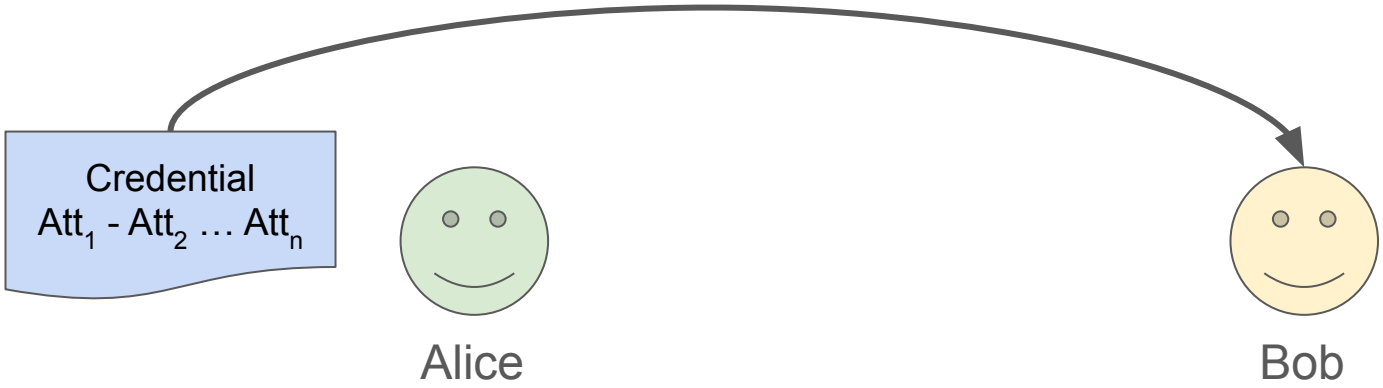
1. Short theory
2. Jupyter exercises
3. Discussion
4. Longer coding exercise

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

1 - Signing Simply with RSA

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Attribute Sharing



© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Attribute Sharing - 1st Problem

Are the attributes correct?

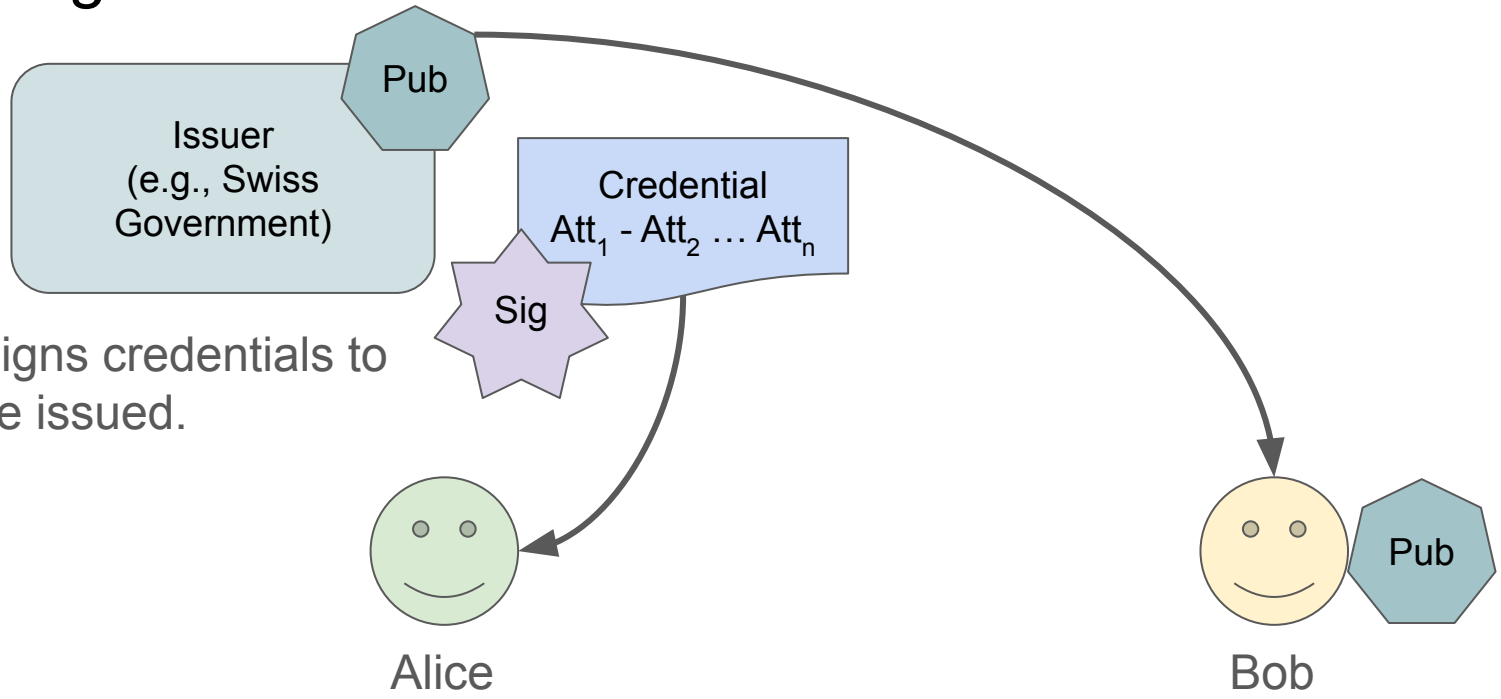
Credential
Att_1 - Att_2 … Att_n

Alice

Bob

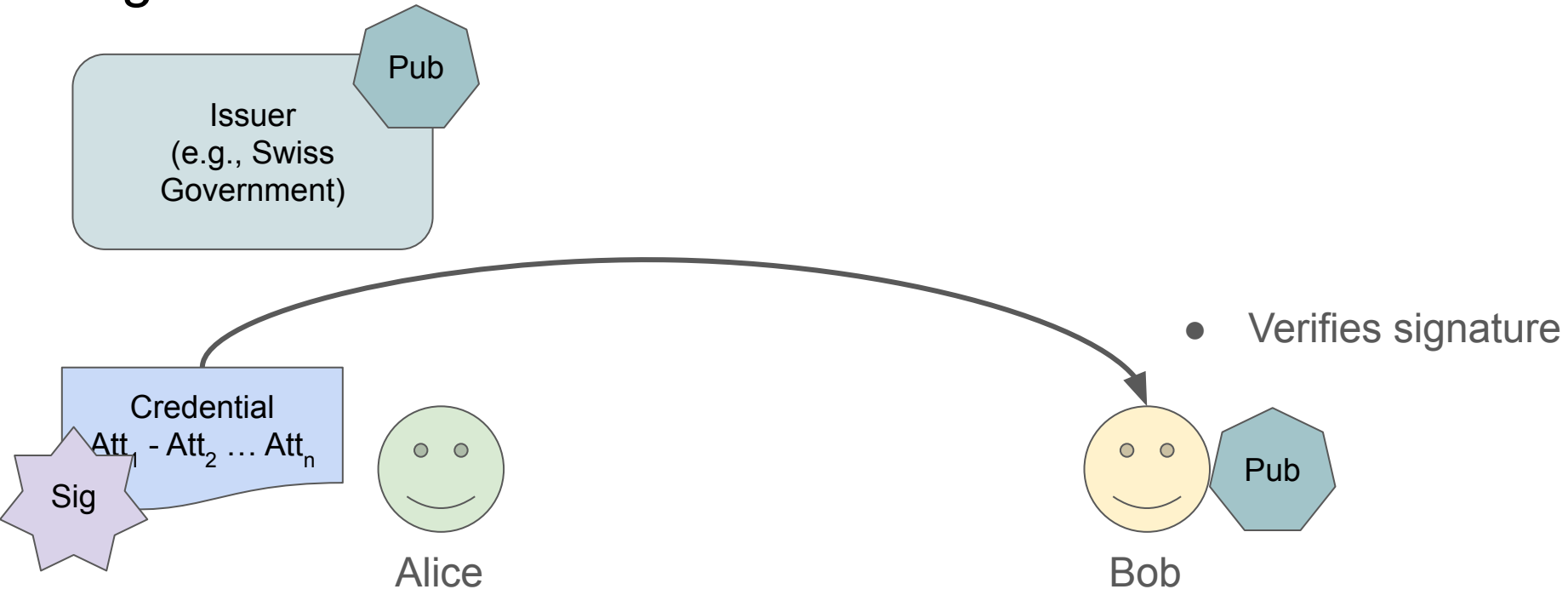
© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Signature from Issuer



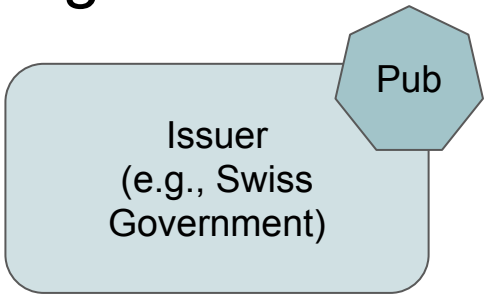
© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Signature from Issuer

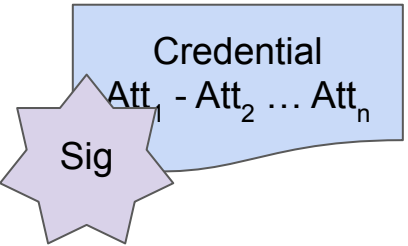


© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

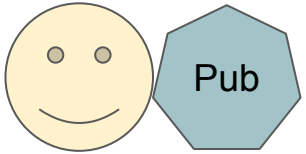
Signature from Issuer - 2nd Problem



Bob learns all attributes



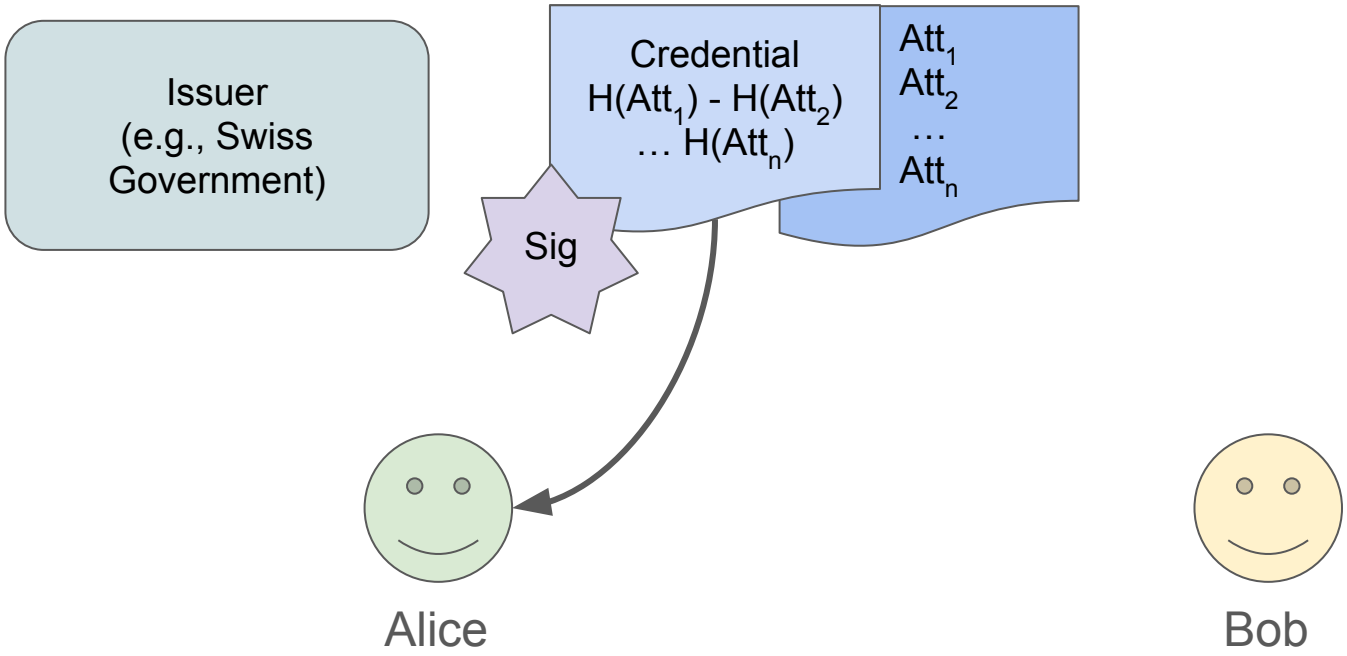
Alice



Bob

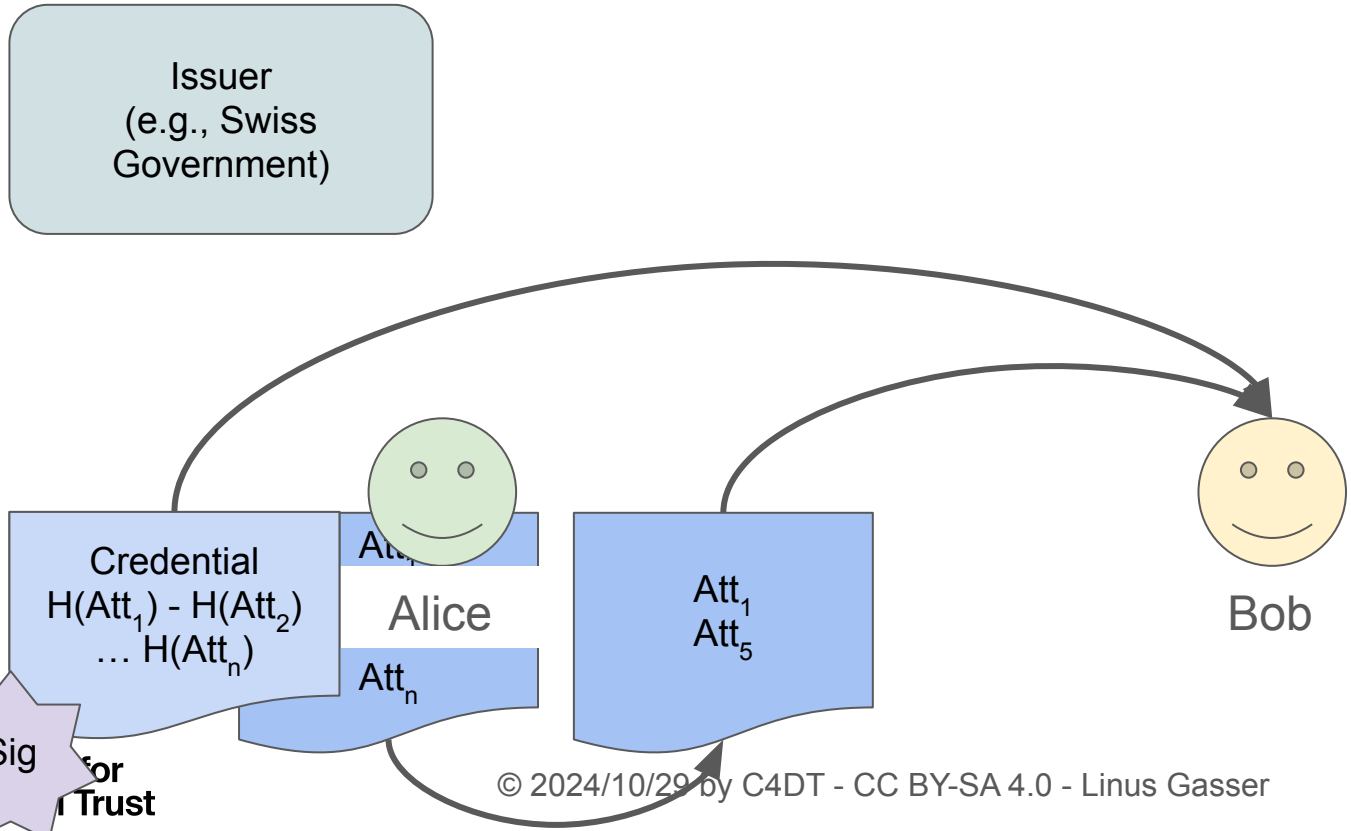
© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Selective Disclosure



© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Selective Disclosure

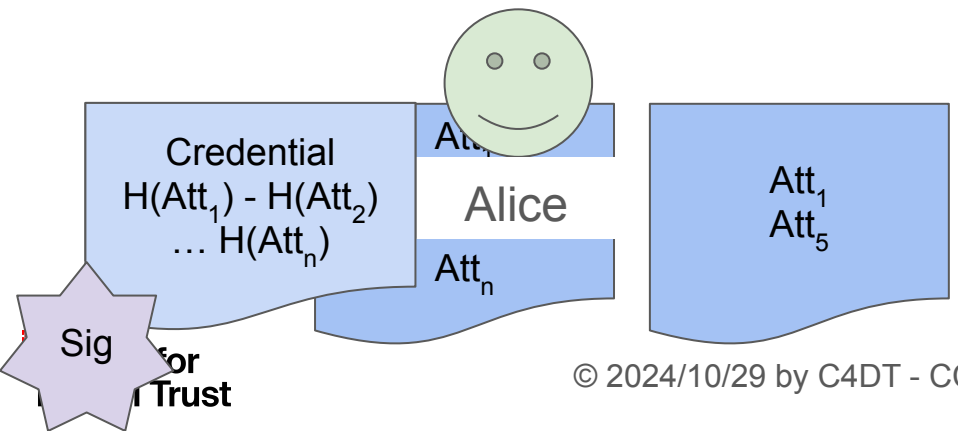


© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Selective Disclosure

Issuer
(e.g., Swiss Government)

- Verifies signature
- Learns only disclosed attributes 1 and 5

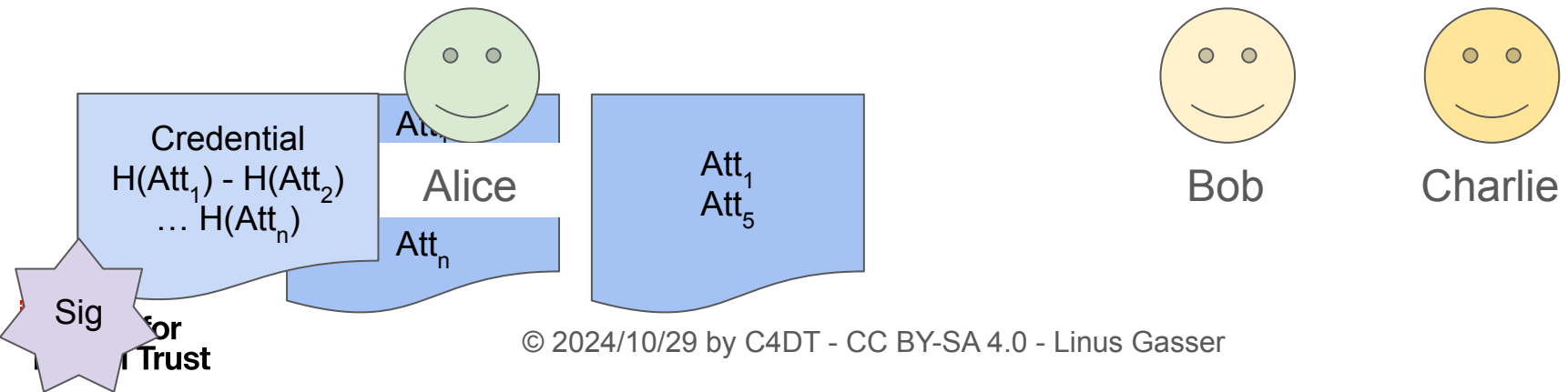


© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Selective Disclosure - 3rd Problem

Issuer
(e.g., Swiss Government)

Linkability: Bob and Charlie can correlate Alice's attributes



© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Exercise 1 - Signing Simply with RSA

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Wrap-up slide

- The issuer allows the verifier to trust the data from the holder
- Selective disclosure can hide personal data to the verifier
- For low-entropy data, even cryptographic hashes do not provide anonymity
- LD-JSON Verified Credentials from EU Digital Wallet are linkable

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

2 - Unlinkable proofs using BBS+

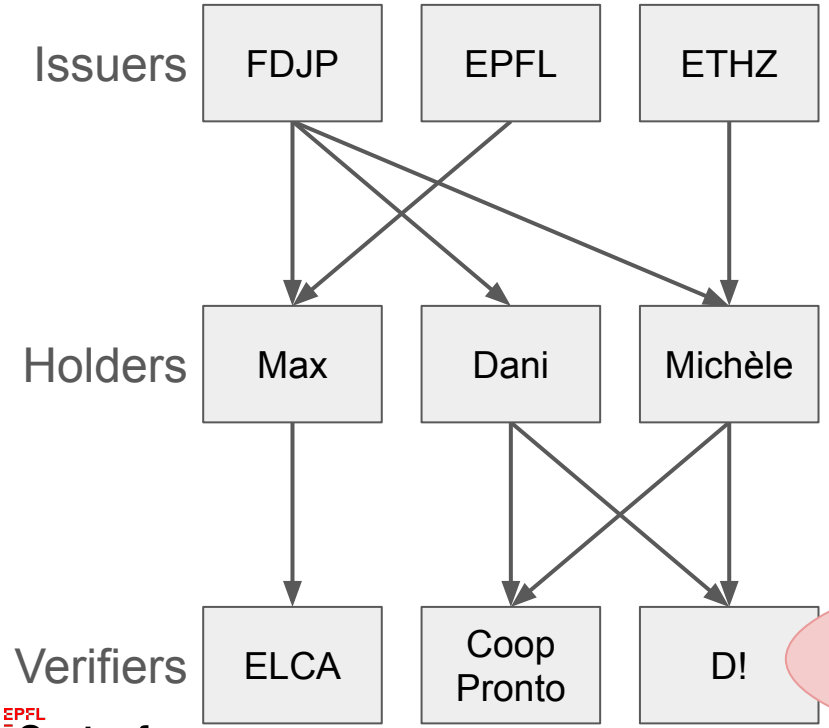
© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Why Unlinkability?

- No correlation between visits
- Reduces attack surface if data leaks
- Privacy / Profiling
  - less knowledge about visitors -> less influence
  - no following of holders -> physical security (e.g., stalkers)

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

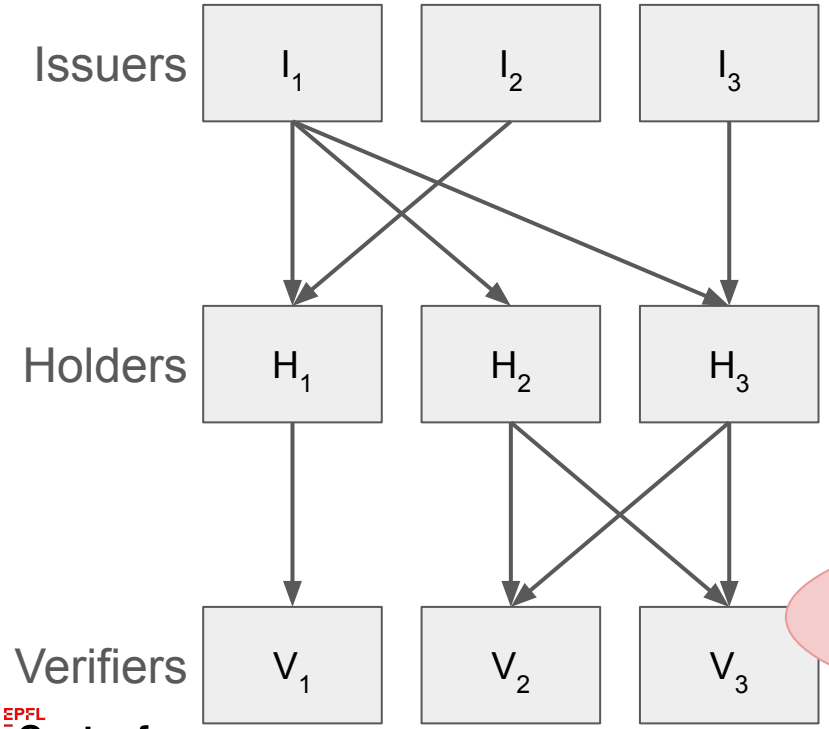
Unlinkability Vows (in addition to anonymity)



1. **Validity check** by Coop and D! on Dani
   unlinkable by the FDJP
   -> movement tracking
2. **Validity check** by D! on Dani and Michèle
   unlinkable by the FDJP
   -> counting of usage by a verifier
3. **Has CH Master Degree** check by ELCA
   unlinkable to EPFL or ETHZ
   -> discrimination against a school
4. **Age** check by Coop and D! on Dani
   unlinkable by Coop and D!
   -> user profiling

© 2024/10/29 by C4DT - CC BY

Unlinkability Vows (in addition to anonymity)



1. **I** has **Val(V_x(H_1))** and **Val(V_y(H_2))**
   movement tracking: **H_1 =? H_2 ∀ x,y ∈ 1..3**
2. **I** has **Val(V_1(H_x))** and **Val(V_2(H_y))**
   verifier usage counting: **V_1 =? V_2 ∀ x,y ∈ 1..3**
3. **V** has **Attr(H_x(I_a))**
   school discrimination: **a =? 2,3 ∀ x ∈ 1..3**
4. **V_x** has **Attr(H_1)**; **V_y** has **Attr(H_2)**
   user profiling: **H_1 =? H_2 ∀ x,y ∈ 1..3**

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

How to Make it Unlinkable

1. and 2. - validity or revocation check
   - Cryptographic accumulators - slow and potentially huge
3. Issuer hiding
   - Create "meta issuer" - issuer of issuers
4. User profiling
   - BBS+ signatures

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Avoid User Profiling with BBS+

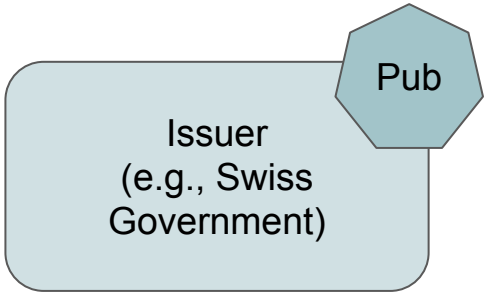
If $\mathbf{V_x}$ has **Attr($\mathbf{H_1}$)**; $\mathbf{V_y}$ has **Attr($\mathbf{H_2}$)**, it's difficult to verify if $\mathbf{H_1 =? H_2}$, $\mathbf{\forall\ x,y \in 1..3}$

- Issuer signature needs to be blinded (valid but different each time)
- Hashes of the non-disclosed fields need to be blinded
- BBS(+) to the rescue
  - Zero-knowledge proof:
    Here is a proof that I know a signature of the following hash(es)
  - BBS: original paper, security proof only later
  - BBS+: added a random factor to create a security proof
  - BBS#: extension proposed by Orange to do holder binding
  - Short BBS: not using pairing-based cryptography

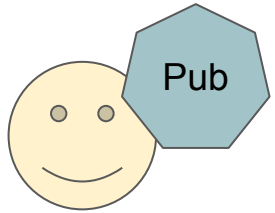
Blinding disclosed fields -> Predicate Zero Knowledge Proofs, not in BBS+!

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

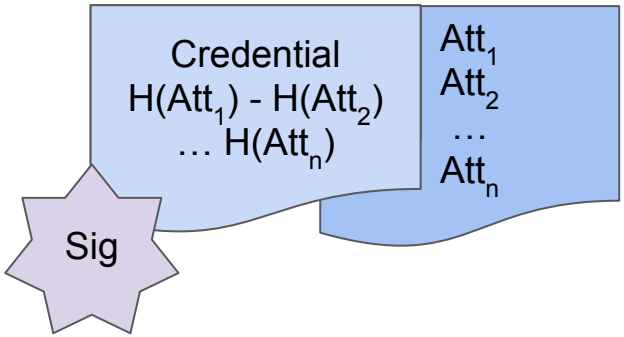
BBS+ in One Slide

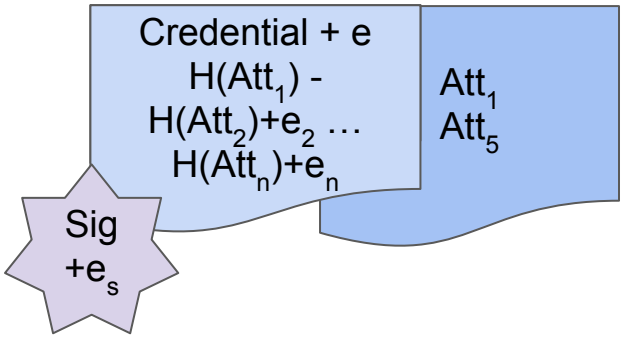


Alice



Bob





Can verify
Sig + e_s
against the
blinded credential
using Pub_{Issuer}

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Exercise 2 - Unlinkable proofs using BBS+

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

What we Learnt

- BBS+ creates unlinkable proofs
- It can selectively disclose fields chosen by the holder
- Hover, the disclosed fields might still be used to link proofs

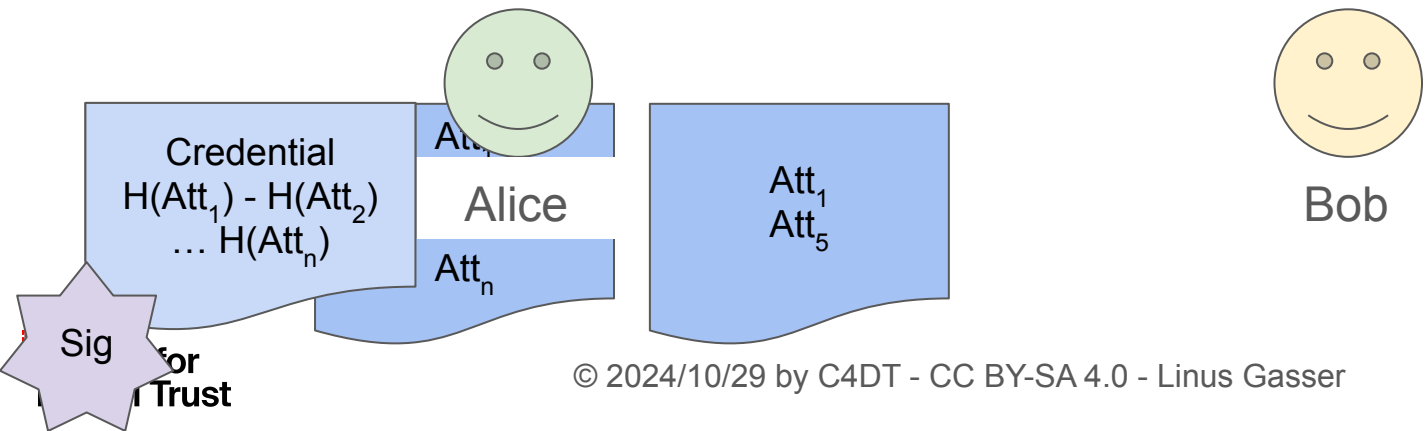
© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Selective Disclosure - 4th Problem

Issuer
(e.g., Swiss Government)

Too Much Information: Bob learns more than necessary.

- Verifies signature
- Learns only disclosed attributes 1 and 5



© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Unlinkability - and Now?

Disclosed values are fully visible, for example

- Birthdate (when you only want to prove you're > 65)
- Salary (instead of proving you earn less than 30k)
- Address (reduction for a ticket bc you live in VD)

This is not desirable because of:

- Privacy: you don't want to give away that data
- De-anonymization: when combining fields, you can get a very small anonymity set (male, 1.1.1978, 1015)

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

3 - Predicate Proofs with ZKPs

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Zero Knowledge Proofs 101

 Setup	All agree on the statement **x** which should be fulfilled	Common reference string (**CRS**)		
 Prover			Creates proof **p** for private data **w** fulfilling **x**	
 Verifier				Can verify that **p** fulfills **x** w/o knowing **w**

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

An Example of a Statement

Wanting to buy a ticket with a reduction for retired people:

Proving the issuer signed a verified credential which includes an age >= 65:

- All agree on the condition **x**:
  - I know a signature $\mathbf{Sig}_{\mathbf{issuer}}+\mathbf{e}_{\mathbf{sig}}$ to a hash $\mathbf{H}_{\mathbf{A}}+\mathbf{e}_{\mathbf{A}}$ verifiable by $\mathbf{Pub}_{\mathbf{issuer}}$ AND
  - I know a number $\mathbf{N}_{\mathbf{A}}$ which hashes to $\mathbf{H}_{\mathbf{A}}+\mathbf{e}_{\mathbf{A}}$ AND
  - $\mathbf{N}_{\mathbf{A}}$ is above or equal to 65
- The holder creates a proof **p** for **x** using their **w**
- The verifier can check **p** fulfills **x**, knowing only $\mathbf{Pub}_{\mathbf{issuer}}$

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Biggest Zero Knowledge Proof Families in 2024

Name	Foundation	Setup	Proof creation	Verification
SNARK	Bilinear pairings, elliptic curves PQ: No	Yes Time: long	Size: constant Time: fast (w/o setup)	Time: fast
STARK	Hash functions PQ: Yes	No	Size: large Time: slow	Time: fast
Bulletproofs	Elliptic curves PQ: No	No	Size: medium Time: slow	Time: medium

2024/10 - depends also on complexity of statement **x**

Some Zero Knowledge Terms

- **Completeness**: If the statement is true, an honest prover will be able to convince an honest verifier of this fact.
- **Soundness**: If the statement is false, no dishonest prover can convince an honest verifier that it is true, except with a very small probability.
- **Zero-Knowledge**: If the statement is true, the verifier learns nothing other than the fact that the statement is true.
- **Interactive**: the verifier interacts over many rounds with the prover, until they are convinced of the statement. Sigma protocols are interactive ZKPs.
- **Succinctness**: the proof size should be small, and the verification time should be fast

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Exercise 3 - Predicate proofs with ZKPs

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Wrap-up slide

The good:

- Zero Knowledge Proofs allow to minimize the data leakage from the credentials
- The docknetwork/crypto library has a very powerful mechanism to set up a ZKP statement

The bad:

- There are no standards yet - it is very new
- Some statements are still very complicated to express

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

4 - ZKP Considerations

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Difference Between ZKP Systems

- Setup: either with (zkSNARK) or without (zkSTARK, Bulletproofs)
  - with: smaller and faster proofs and verifications, but need to trust the setup
  - without: no trust needed
  - as seen in the exercises, fast advancing research turns the tables
- Statement complexity
- Setup: time and size ms to seconds; 1-100kB
- Proof creation: time and size - ms to minutes; 100B to xMB
- Verification: time - ms to seconds

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

(Lego)Groth16 <-> Bulletproofs++

- Groth16 is an "old" algorithm which is well understood
- Bulletproofs(++) is more advanced, and looks like it could replace Lego16
- LegoGroth16 is an example of combining various ZKP algorithms
- The docknetwork/crypto library adds yet another layer

Comparison in exercise:

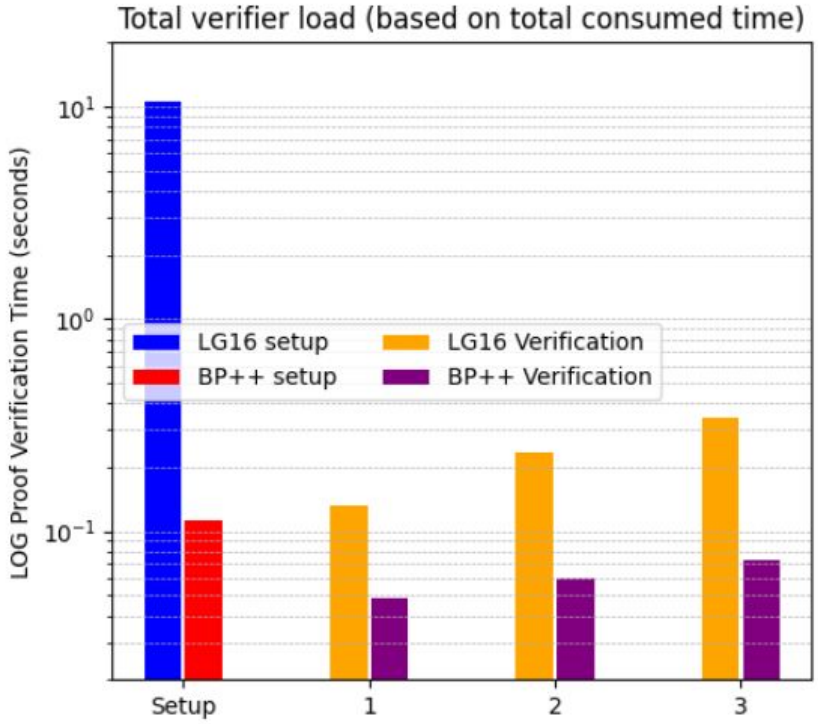
- Computation cost:
  - Server: setup and verify
  - Client: setup and create proof
- Communication cost:
  - Server -> client: setup material
  - Client -> server: proof

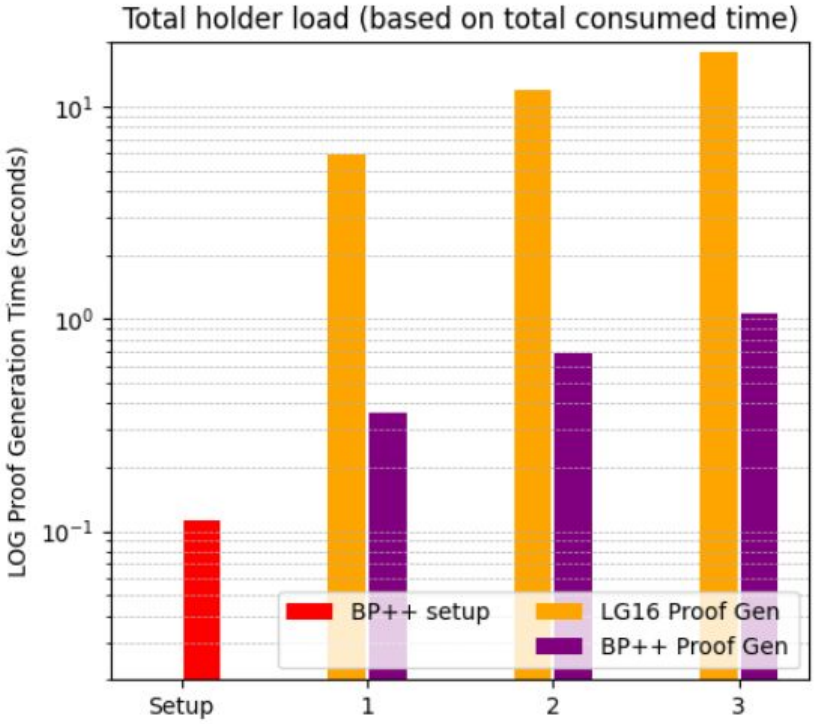
© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Exercise 4 - ZKP Considerations

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

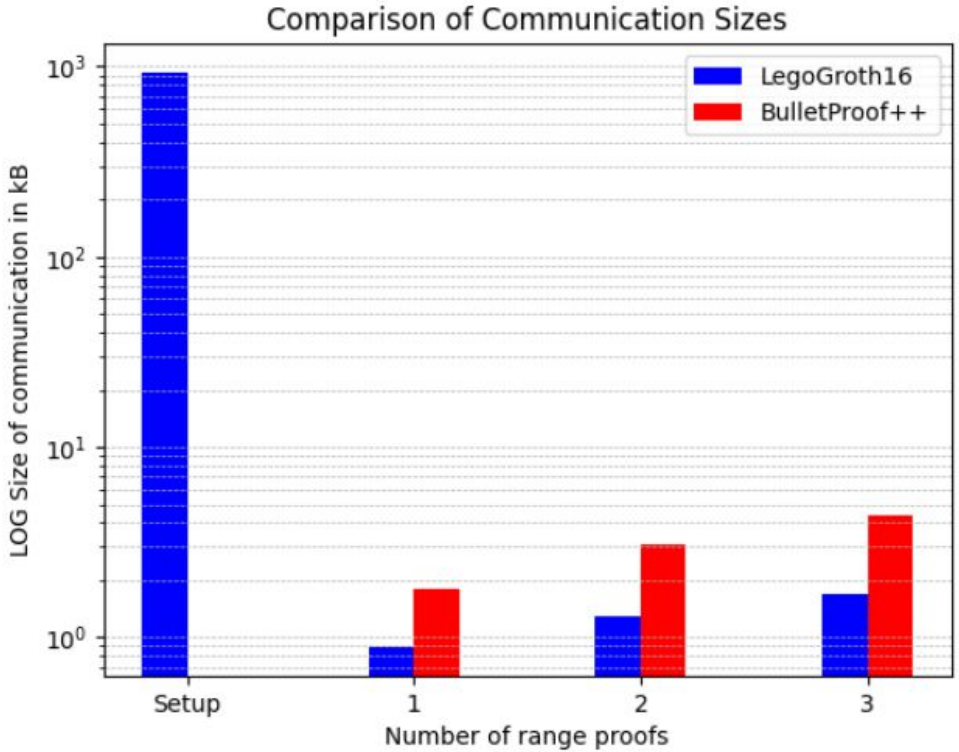
Setup and Proof Generation - Logarithmic y-scale!





© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Communication Sizes



© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Interpretation

This is very specific to the *docknetwork/crypto* library:

- Special setup to create composed proofs
- Not optimized for 'simple' range proofs

Generally:

- The setup for the LegoGroth16 can be re-used by the verifier
- The setup for Bulletproofs++ must be done every time
- The communication size for LegoGroth16 is very high

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Conclusions

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser

Setting up a Trustworthy E-ID

- What is important?
  - Convince Swiss citizens that E-ID is trustworthy
  - Use Cases for the E-ID
- Questions for the Swiss E-ID
  - ZKP for ECDSA signatures for holder binding
  - Which basic signatures scheme to use
- Standardizations
  - BBS+ has an IETF draft
  - Nothing yet for ZKPs

© 2024/10/29 by C4DT - CC BY-SA 4.0 - Linus Gasser