Privacy & Identity Lab

# Privacy Friendly Digital Identity Wallets?

## The devil is in the details (unfortunately)!

**Jaap-Henk Hoepman**

Privacy & Identity Lab
iHub
Radboud University
Karlstad University
University of Groningen

✉ *jhh@cs.ru.nl // 🖱 www.cs.ru.nl/~jhh // 🖱 blog.xot.nl // @xotoxot*

# Introduction

## What is eIDAS?

- Regulation covering eID and Trust Services

## Why eIDAS 2.0?

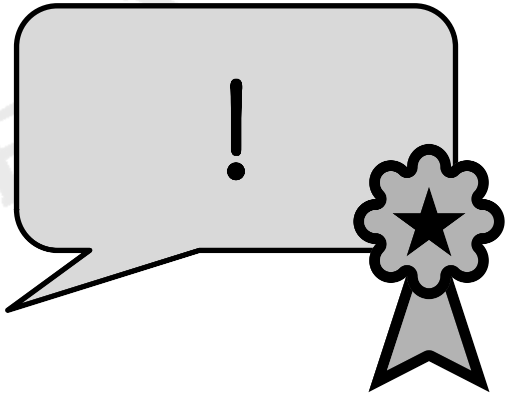
- eIDAS 1.0 not succesful: little cross border use of national eIDs
- Threat of Apple/Google Wallets

## What's new in 2.0?

- European Digital Identity Wallet
  - An app on a smartphone
  - Issued by Member States
    - according to a common standard (the Architecture Reference Framework, latest version 1.4.0, May 22, 2024 )?
  - Attributes, certficates, documents: essentially a Personal Data Store
  - Supposedly privacy friendly

# Attribute Attestations (claims based authentication)

**Issuer I** claims that **Person P** has **Value V** for **Attribute A**

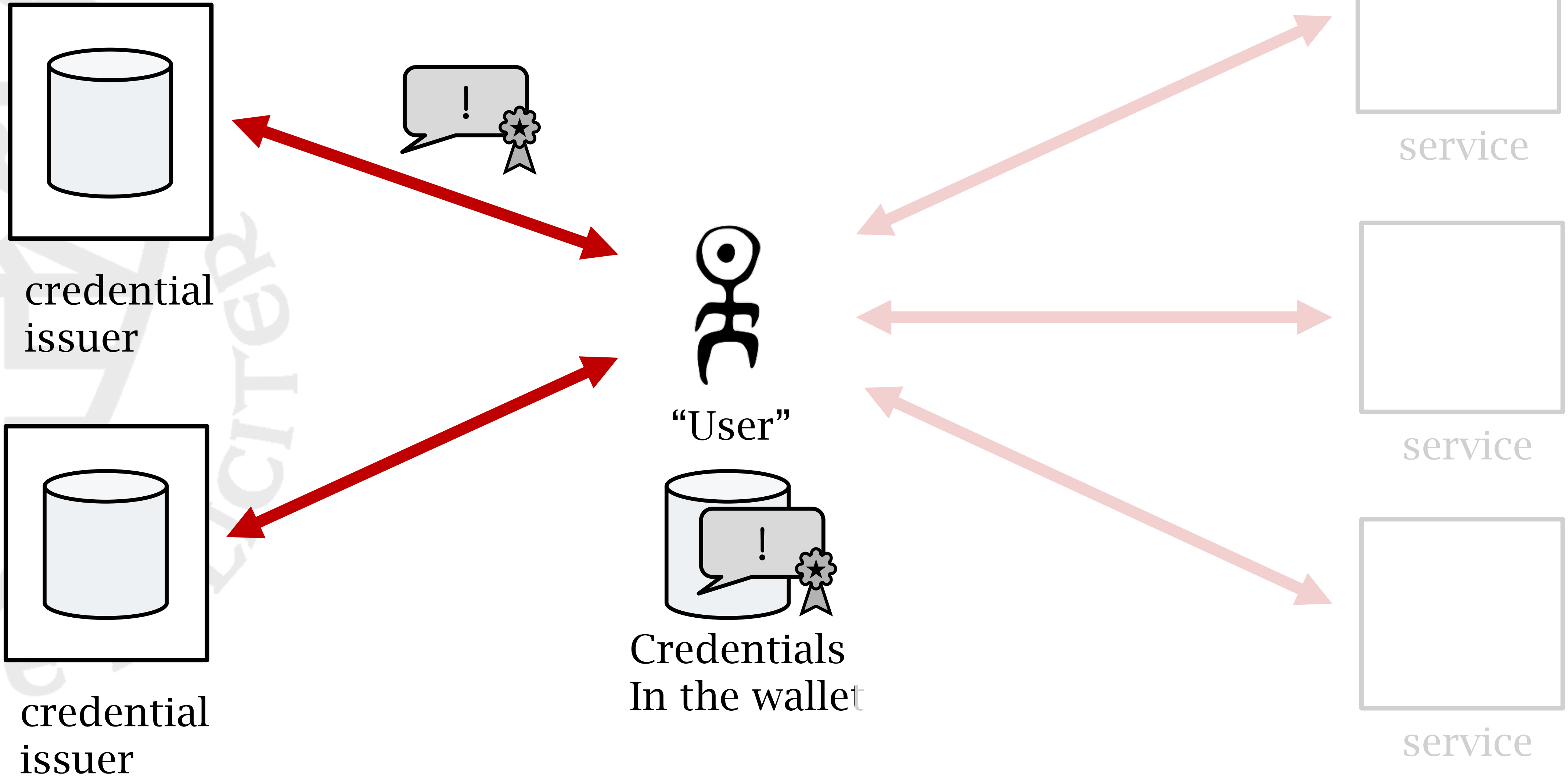


The Dutch government claims that Jaap-Henk Hoepman has the Dutch nationality

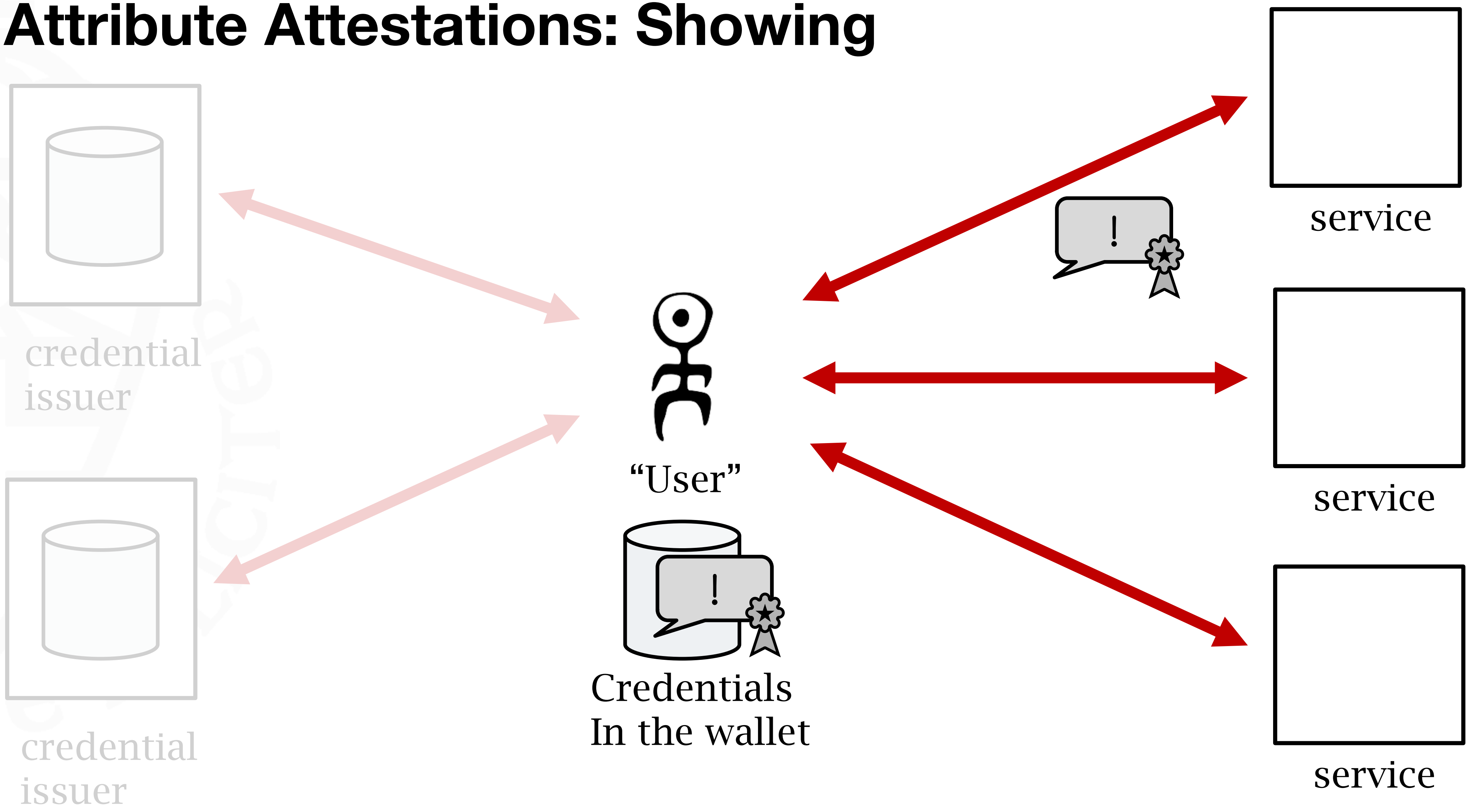
The bank claims that Jaap-Henk Hoepman has a good credit rating

The land registrar claims that Jaap-Henk Hoepman has a PhD in law

# Attribute Attestations: Issuing



# Attribute Attestations: Showing

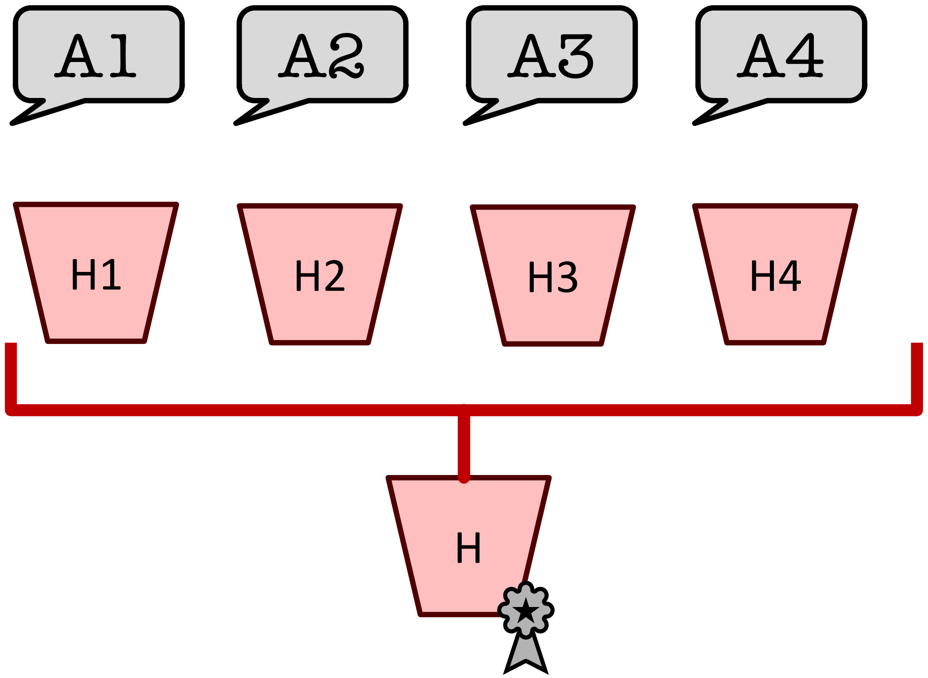


# Why use attribute attestations?

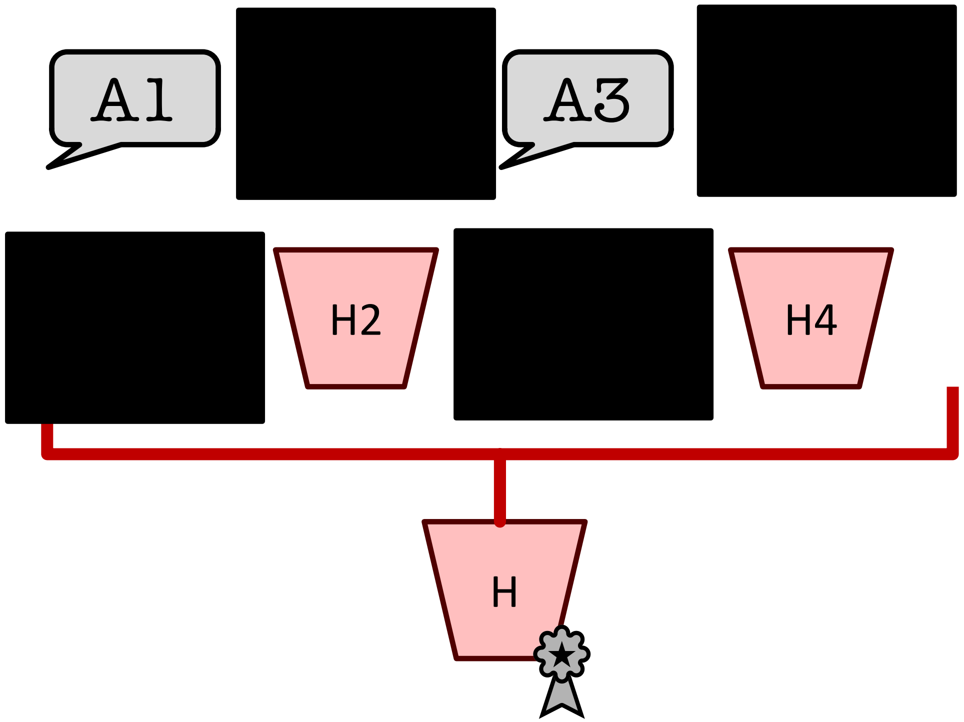
- **Selective disclosure**
  - Only reveal required attributes
- **Self-souvereignty**
  - Decide what attestations to get, and from whom
- **Decouple getting and using an attribute (issuer unlinkability)**
  - Prevent issuer from learning when and where you use an attribute
    - *Significant issue in 'social logins'*
- **Decouple successive uses of an attribute (multi-show unlinkability)**
  - Prevent profiling by relying parties (using attestation singature as persistent identifier)
- **But still guarantee security of attributes**
  - Increased by binding to a trusted hardware element

# Attribute attestations in eIDAS 2.0 are lame, however

- Essentially a set of singed (salted) hashes



- Selective disclose: reveal preimages of the associated hashes



# Why is this lame?

- **Selective disclosure**
  - Only reveal required attributes
- **Decouple getting and using an attribute**
  - Issuer knows signature; signature revealed to relying party
  - When relying parties collude with issuers, users can be profiled
- **Decouple successive uses of an attribute**
  - See above
    - *Proposed solution: issue many attestations (with different salts) in batch, use once and then throw away; but this is cumbersome; and will it be mandatory?*
- **But still guarantee security of attributes**
  - Increased by binding to a trusted hardware element

# Better to use true Attrbute Based Credentials

- **Based on Zero Knowledge proofs and special signature schemes (BBS)**
  - Don't reveal signature, but prove you have it
- **True unlinkability**
  - Between issuer and relying party
  - Multi-show (at one or among several relying parties)
- **Efficient implementations exist**
  - With proper security proofs

- **But:**
  - Not using "state approved" cryptographic primitives
  - Not implemented in current secure trusted hardware components
    - *device binding seen as very important security property*
    - *could be solved using traditional crypto, while using modern crypto ABCs*

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

# Revocation

- **Revoking attestations**
  - URL to revocation status included in attestation
  - Added by issuer
  - Always checked by relying party
- **This breaks issuer unlinkability!**
  - Every use is checked
  - Using server determined by the issuer
  - Revealing IP address of RP

- **Revoking wallets**
  - By revoking the Wallet Instance Attestation
- **But but….**
  - This allows Wallet Instance Attestation Issuers to trace each and every time when and where wallet is used!

# Preventing over-authentication?

- **Relying parties must register**
  - And get access certificate that authenticates them to wallet
  - Unfortunately does not contain list of allowed attribute requests!
- **Users must check attribute requests**
  - These are logged
  - And can be reported
- **Issuer can specifiy disclosure policy with attestation**
  - Restricting at which relying party attestation can be used
  - But… how does issuer know which RPs to trust???
  - Also: not responsibility of individual issuers, but of overall scheme authorities! I.e. the Commission!

# General observations

- **Technical specifications (Arhcitecture Reference Framework)**
  - Determine real security/privacy properties
  - Developed without much oversight or academic/civil society participation
- **In general a problem with standardisation**
  - Participation costs time and money
  - Influence depends on level of participation
  - Stakeholders with a direct (financial) interest can/will invest more

Dozen vague implementing regulations

One clearly defined standard (e.g. ARF)

# Questions?

[Monty Python’s
Argument Clinic sketch]