Maatregelen tegen overvraging in SSI

en het Yivi ecosysteem

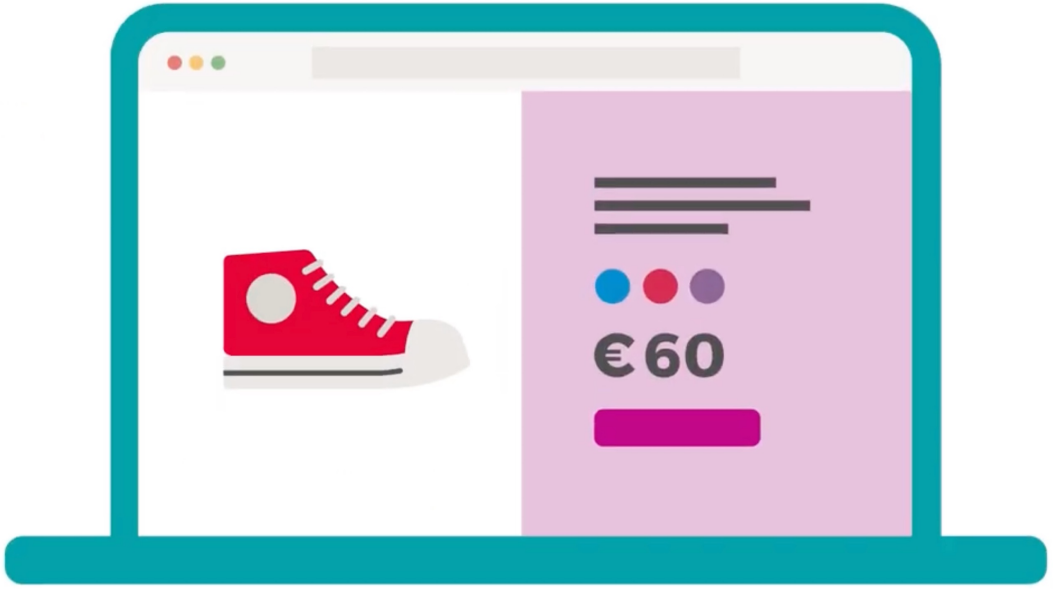
Yivi meetup, 10 november 2023

Job Doesburg

Radboud Universiteit

WAT IS OVERVRAGING

Probleem



Webshop.nl vraagt om de volgende gegevens:

- Je voornaam
- Je achternaam
- Je adres
- Je BSN

Annuleer Akkoord

Probleem

Uw toekomstige werkgever vraagt om de volgende gegevens:

- Je voornaam
- Je achternaam
- Je diploma's
- Je medicatielijst

Annuleer Akkoord

Probleem

Is klikken op “akkoord” wel échte (geïnformeerde, vrijwillige) toestemming?

- **Onwetendheid** / **onverschilligheid** van de gebruiker
- **Machtsverhouding** tussen verifier en gebruiker

Hoe kunnen we gebruikers beschermen tegen ‘onaanvaardbare’ disclosure requests?

“Requiring users to *know* which verifiers to trust is very similar to asking users to know which websites to trust, even when they have not visited them before. […]

Web browsers indicate if a secure TLS session has been established […] by displaying a lock icon next to the web site’s URL. Something similar will be needed for SSI […] to enable human users to determine if a verifier is trustworthy or not”

(Chadwick et al., 2023)

Probleem

Is klikken op “akkoord” wel échte (geïnformeerde, vrijwillige) toestemming?

- **Onwetendheid** / **onverschilligheid** van de gebruiker
- **Machtsverhouding** tussen verifier en gebruiker

- Gebruikers hebben actief hulp nodig bij het beschermen van hun eigen privacy!
  - **Zorgplicht?** Voor platform (Yivi)? Issuer? Overheid?

Probleem

Waarom overvraging een *groter* gevaar is in SSI dan in andere vormen van IdM:

- **Unsiloing van data** → meer gegevens, gemakkelijker beschikbaar
- **Geen poortwachters** → geen IdP die verantwoordelijk kan worden gehouden
- **Verlies van context-awareness** → geen intuïtieve contextassociatie met een specifieke IdP

- **Onterechte verwachtingen**: SSI wordt aangeprezen als privacy-vriendelijke technologie. Gebruikers kunnen wellicht (ten onrechte!) verwachten dat het schenden van privacy *onmogelijk* is.

- **Gedecentraliseerde opzet van SSI maakt overvraging ontransparant en lastig te detecteren**

HET HUIDIGE YIVI SYSTEEM (EN HET SSI LANDSCHAP)

- Weinig issuers, veel verifiers
- Bewuste keuze: **iedereen kan verifier zijn**
- Verifier zijn is **eenvoudig** (belangrijk voor adoptie!)

- **Yivi: “Zelf de baas over jouw gegevens. All you. All yours”**
- Gebruikers kiezen met wie zij ***hun gegevens*** delen (autonomie).
- *Ideologisch*: volledige autonomie is een *feature*

  Pragmatisch: sommige gegevens kunnen te gevoelig zijn om door iedereen opvraagbaar te zijn (zelfs met toestemming van de gebruiker)…

  → *Don’t give a monkey a gun*

BACKGROUND

- Use cases:
  - BSN
  - DNA medicatiepas (LUMC)
  - Biometrische gegevens
  - Andere use cases… (mogelijk economische belangen van issuer!)

- Ondertussen, het EU Digital Identity Architecture and Reference Framework (outline):

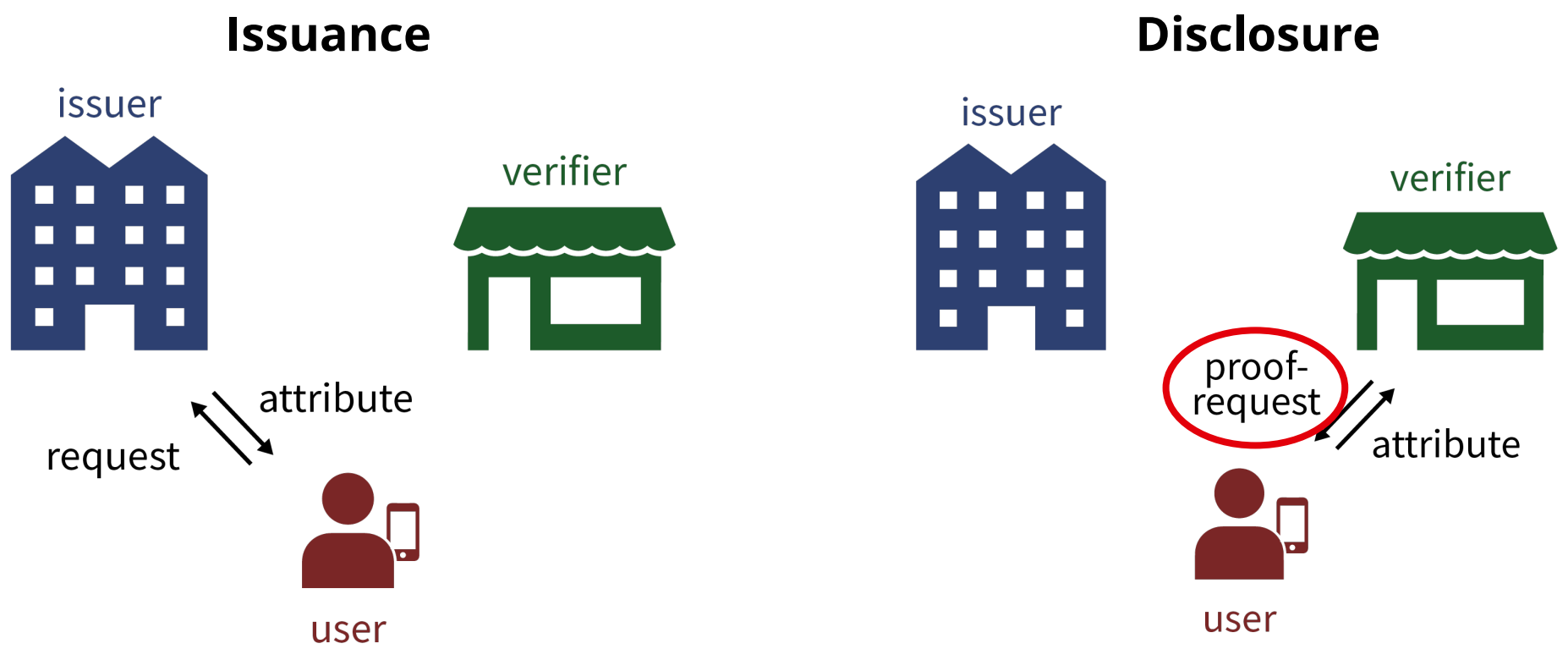
> "In addition, the EUDI Wallet **may**: […] restrict sharing certain sets of attributes with certain parties, or warn the user that the relying party may not be authorized to use/ask for these attributes."

→ dus autorisatie van relying parties wordt *een ding*…

…en huidige SSI implementaties negeren dit

OPLOSSINGEN

YIVI INFRASTRUCTUUR



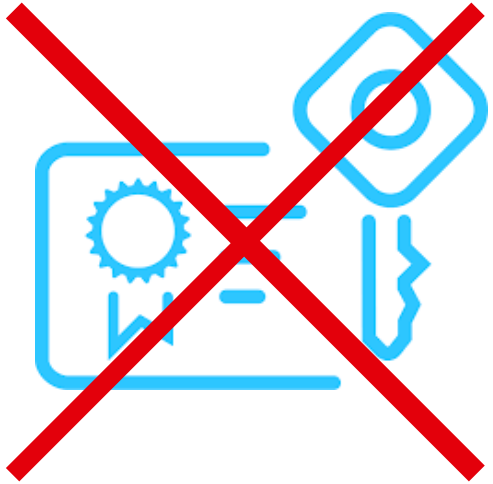
UITDAGINGEN

Uitdagingen voor proof-requests:

1. Authenticatie
2. Autorisatie

Doelen voor implementatie:

- Technisch haalbaar (makkelijk te implementeren en onderhouden)
- Gebruiksgemak voor verifiers (eenvoudige adoptie)
- Behoud van SSI principes (privacy, autonomie gebruikers)
- Minimale administratieve werklast, bij de verantwoordelijke partijen

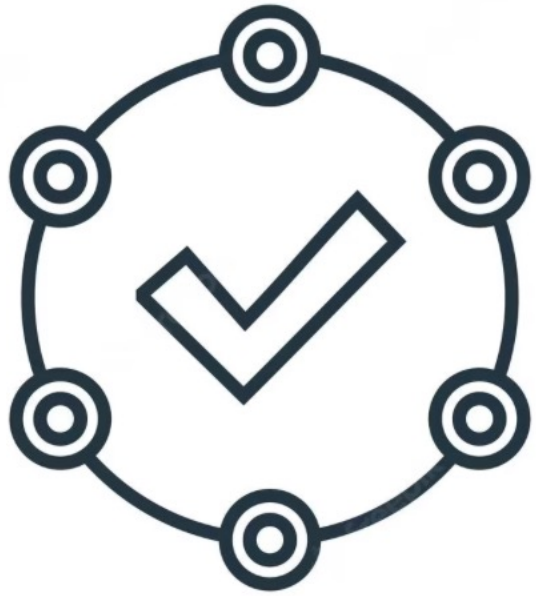


→ geen speciale PKI introduceren als dit niet nodig is

MAATREGELEN TEGEN OVERVRAGING IN SSI

OPLOSSING 1: PROTECTED ATTRIBUTES

- Attributen die enkel door een *geautoriseerde* partij kunnen worden opgevraagd
- Eenvoudig te implementeren



- Yivi: authenticatie via TLS hostnames
  (zoals reeds bestaand voor *pretty verifiers*)
  - Schema linkt hostnames naar requestor ID
  - Geen (extra) key management, TLS is al vereist!

- Autorisatie:
  - Via *issuer*-schema (lijst met geautoriseerde requestor IDs)
  - Via authorisatie-server (vergelijkbaar met revocatie-server)

⇒ **For specifieke, (zeer) gevoelige attributen (issuer's verantwoordelijkheid)**

OPLOSSING 1: PROTECTED ATTRIBUTES

```
<IssueSpecification version="...">
    ...
    <Attributes>
        <Attribute id="BSN">
            <Name>
                <en>Burgerservicenummer</en>
                <nl>Social security number</nl>
            </Name>
            ...
            <AuthorisedRequestors>
                <RequestorID>
                    pbdf-requestors.someauthorisedparty
                </RequestorID>
            </AuthorisedRequestors>
        </Attribute>
        ...
    </Attributes>
</IssueSpecification>
```

Issuer scheme

```
[
    ...
    {
        "id": "pbdf-requestors.someauthorisedparty",
        "name": {
            "en": "Example requestor",
            "nl": "Voorbeeld requestor"
        },
        "hostnames": [
            "authorised-requestor.example.com"
        ],
    },
    ...
]
```

Requestor scheme

OPLOSSING 2: CERTIFIED DISCLOSURE REQUESTS

- Protected attributes zijn **geen generieke oplossing** tegen overvraging
  - Denk aan een boekenwinkel die om e-mailadres vraagt voor een bestelling
  - *Context* van een data request is essentieel!

  → Beoordeling van disclosure request door derde partij vereist

- Centrale autoriteit
  - Duur & onrealistisch op globale schaal
- Open publieke zelf-registratie (enkel authenticatie)
  - Democratische instanties en belangengroepen kunnen audits uitvoeren
  - Transparantie → stimulans voor zelfregulering
- Hybride aanpak!

⇒ Geen perfecte technische oplossing, maar degelijke maatregel in de praktijk

OPLOSSING 2: CERTIFIED DISCLOSURE REQUESTS

Requestor scheme

```
[
    ...
    {
        "id": "pbdf-requestors.someauthorisedparty",
        "name": {
            "en": "Example requestor",
            "nl": "Voorbeeld requestor"
        },
        "hostnames": [
            "authorised-requestor.example.com"
        ],
        "certified_requests": [
            {
                "disclose": [
                    [
                        "pbdf.pbdf.email.email"
                    ]
                ],
                "reason": {
                    "1": {
                        "en": "To send you a newsletter",
                        "nl": "Voor het versturen van een
                            nieuwsbrief"
                    }
                }
            },
            ...
        ]
    },
    ...
]
```

USABILITY ASPECTS

- Wallet moet context van een disclosure request weergeven
  - Wie ontvangt de data? *(pretty verifiers; niet alleen hostname)*
  - In welke context? (!)
  - Waarom is deze data noodzakelijk? (data minimisatie)

- *Permissive* / *strict* wallets (waarschuwen of voorkomen)
  - Verschillende types waarschuwingen
  - Standaardgedrag (habituation) / cognitive bias tegengaan
  - Over het algemeen: permissive > strict

- Denkrichting: categorisering van credentials en verifiers in “spheres”
  - Bijv. ‘gezondheid’, ’onderwijs’, ‘financieel’
  - Detecteer en waarschuw bij *sphere transgression*

CONCLUSIES

- **Protected attributes**: *issuer's verantwoordelijkheid*
- **Certified disclosure requests**: *third-party verantwoordelijkheid*
- Hybride implementaties zijn mogelijk; systemen kunnen naast elkaar bestaan

- User experience design is van belang!

- TLS-based authenticatie en scheme-based autorisatie is eenvoudig voor Yivi en verifiers
  - Schaalbaarheid kan problematisch zijn op lange-termijn

    → Federated schemes + *Just-In-Time*-scheme retrieval kunnen dit probleem mogelijk oplossen

We moeten nadenken over de verantwoordelijkheden die bij gebruikers neergelegd worden

Maatregelen tegen overvraging in SSI

en het Yivi ecosysteem

Yivi meetup, 10 november 2023

Job Doesburg

Radboud Universiteit