Bachelor Thesis

Unlinkability of Verifiable Credentials in a practical approach

June 14, 2024

Joel Robles | TI

Inhaltsverzeichnis

- Ziel
- Self-sovereign Identity
- Problem
- Verifiable Credentials & Verifiable Presentations
- OpenID Connect for Verifiable Presentations
- Fazit
- Ausblick

Ziel

Was ist das Ziel?

Die Analyse, ob eine Implementation von Verifiable Credentials mit dem BBS Signature Scheme in der realen Welt, unverknüpfbarkeit beibehält

Self-sovereign Identity

Self-sovereign Identity (SSI)

- Eine Person (**holder**) soll entscheiden können, wer was über sie wissen darf
- Holders dürfen wählen was sie offenbaren und was nicht, auch bekannt als **selective disclosure**
- Erstes Problem:
  - Holder zeigt eine Staatliche ID
  - Ist eine Menge von Daten oder eine Menge von **Attributen**
  - Die Person welche verifiziert sieht alle Attribute
- Zweites Problem:
  - Holder zeigt Attribute einer Person die diese verifizieren will, bekannt als **verifier**
  - Holder zeigt die gleichen Attribute einem zweiten **verifier**
  - Der holder kann ge-**linked** werden
- Heutiger stand - Holder haben keine Kontrolle über ihre Attribute
- Zukünftiger stand dank SSI - Holder haben volle Kontrolle über ihre Attribute

Trust Triangle

- Wie weis ein verifier das eine Menge von Attributen (**credential**) valid ist?
- Er vertraut dem issuer!
- Beispiel: Schweizer ID hat Hologramme

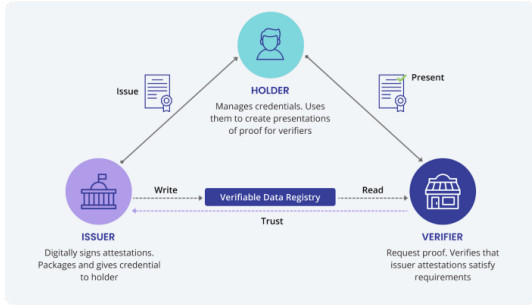

Abbildung: Trust triangle

Problem

Was ist das Problem?

- Wir haben ja schon funktionierende credentials?
- Zukünftig werden diese digitalisiert
- Kreiert verschiedene Probleme
- Wie umsetzen & Sicherheits-Probleme

Verifiable Credentials & Verifiable Presentations

Verifiable Credentials (VC)

- Verifiable Credentials sind eine digitale repräsentation von Physischen credentials

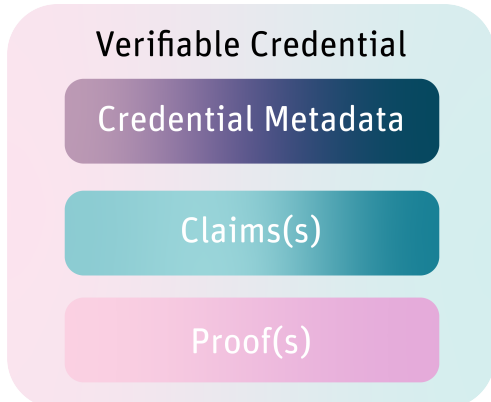

Abbildung: Beispiel VC

Verifiable Credentials (VC)

- Verifiable Credentials sind eine digitale repräsentation von Physischen credentials
- JSON-LD repräsentiert Attribute als **key-value pairs**
- Beispiel:
  - Vorname auf einer ID
  - Repräsentiert als {first_name: "John"}
  - “first_name” ist der key und “John” ist der value

Abbildung: Beispiel VC

VCs and BBS

- Warum werden sie **Verifiable** Credentials genannt?
- Der verifier kann ein VC, welches ihm präsentiert wurde (**Verifiable Presentation**), verifizieren, aufgrund Kryptographischen Signaturen
- Diese zeigen, dass das credential seit der Ausstellung nicht verändert wurde
- Wir nutzen das BBS Signature Scheme (**BBS**)
- Diese Schema bietet **selective disclosure** und **unlinkability**
- Aber wie unlinkability? - Der Verifier braucht die Signatur
- BBS kann **proofs** generieren
- Diese beweisen das der holder die Signatur kennt, ohne diese zu offenbaren
- Fungieren als neue Signatur für das selectively disclosed VC
- Weiter sind proofs unverknüpfbar zwischen jeder Generierung

Verifiable Presentation (VP)

- Ein holder würde gerne ein VC präsentieren
- Dafür werden **Verifiable Presentations** genutzt
- BBS kann nur staments signieren
- Der **RDF** canonicalization Algorithmus, welcher statements aus key-value pairs generiert

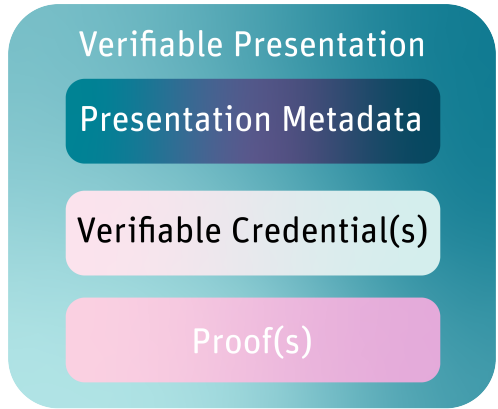

Abbildung: Beispiel canonicalized VP

Sicherheitsüberlegungen von VC/VPs

- Permutation von statements
- Verknüpfbarkeit von Identifikatoren & Metadaten

OpenID Connect for Verifiable Presentations

Transport zwischen holder und verifier

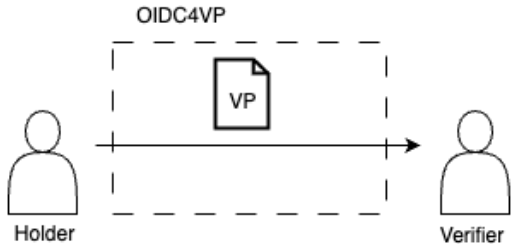

Abbildung: OpenID connect for Verifiable Presentations

Replay attack

- Der holder sendet dem verifier ein VP
- Ein *Man in the Middle* speichert die Vorstellung
- Der *Man in the Middle* kann das gespeicherte VP wiederverwenden
- Um dieses Problem zu umgehen, wird eine zufalls Nummer genutzt (challange-response)

Fazit

Fazit

- Was hat BBS für Vorteile?
  - **Selective-disclosure und unlinkability**
- Wie funktioneren VCs/VPs & wie kann man diese BBS verbinden?
  - **Kanonisierung durch RDF Algorithmus**
- Wie werden die VPs von holder zu verifier gesendet?
  - **OpenID connect for Verifiable Presentations**

Es funktioniert!

Ausblick

Ausblick

- Link Secrets und Blind BBS Signatures für linkability und selective disclosure analysieren
- Implementieren und testen