OpenID for Verifiable Credentials

September 2022

EBSI, explained – first edition

What are the different chapters of this first edition?

01.

Verifiable Credentials Explained

02.

Verifiable Credentials in action

03.

Decentralised Identifiers (DID) Methods

04.

Digital Identity

05.

Issuers Trust Model

06.

OpenID for Verifiable Credentials

07.

Digital Wallets

06. OID for Verifiable Credentials explained – Index

What are you going to learn in this chapter?

06.1

What is OpenID for VCs and why is it important?

06.2

How does OpenID for Verifiable Credential Issuance work?

06.3

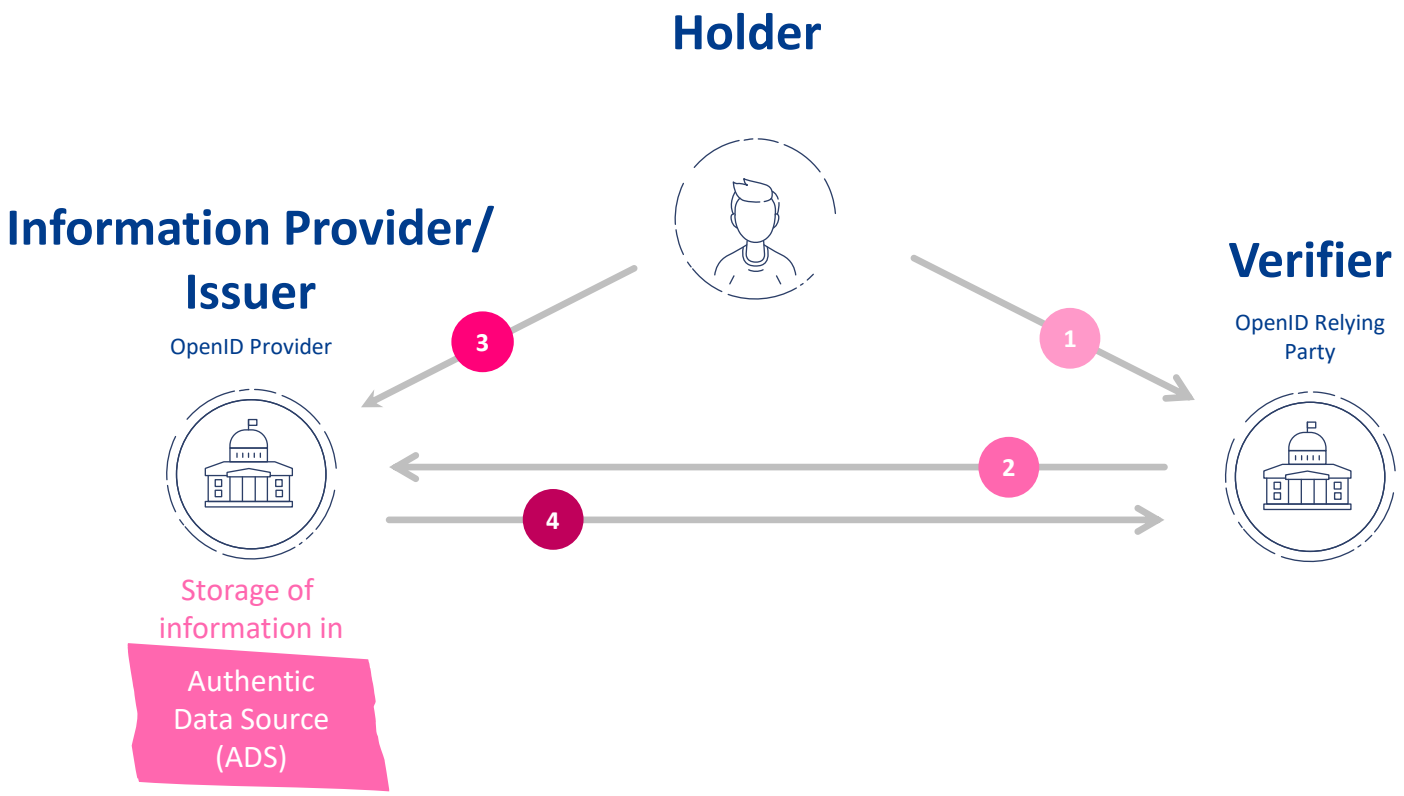
How does OpenID for Verifiable Presentations work?

06.1

What is OpenID for VCs and why is it important?

The challenge with the authorisation-based information sharing

Most protocols only allow holders to authorise verifiers that are in relationship with the issuer to access our information. As a holder, I cannot share information with the verifier I want.



1 Request access

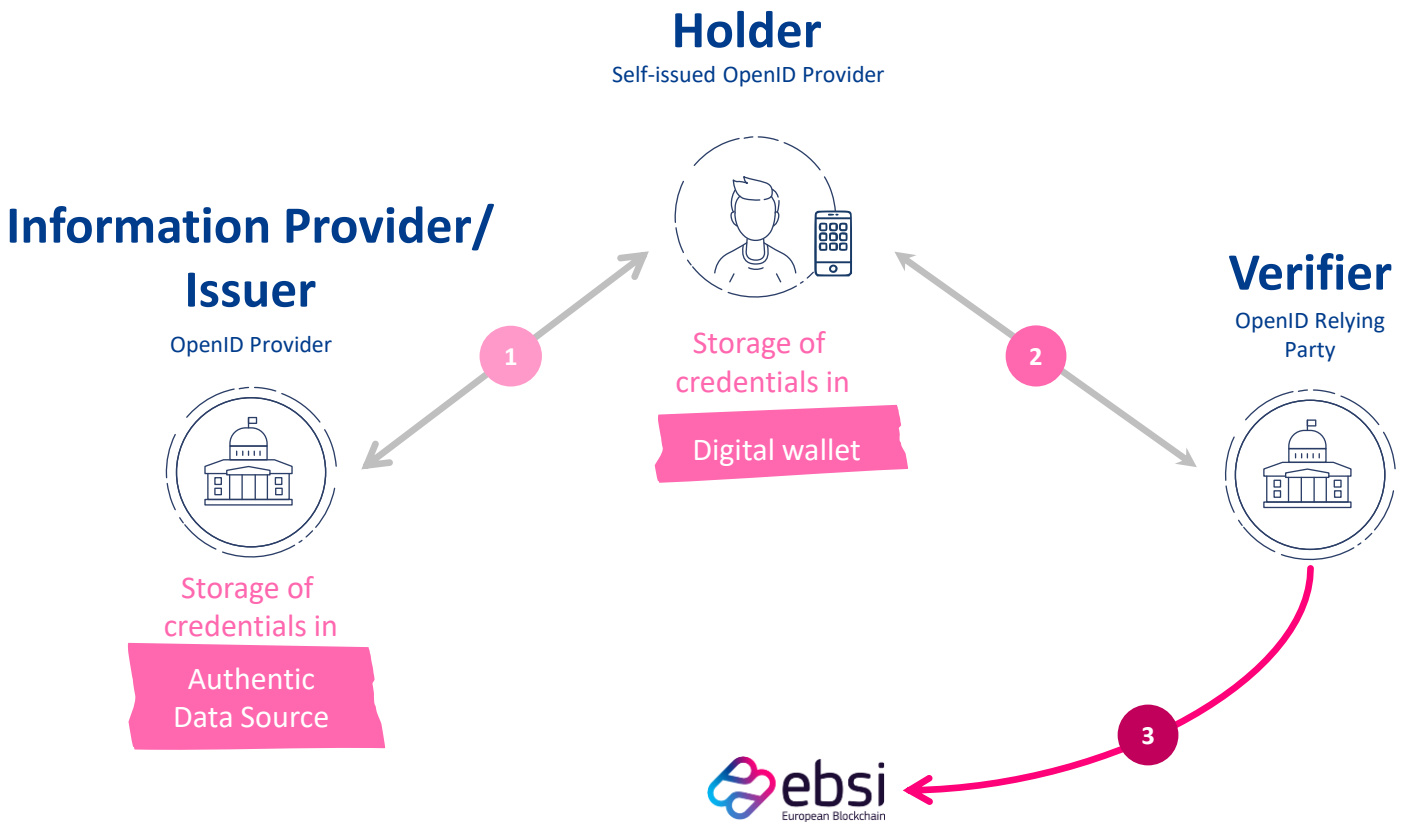
2 Request holder information

3 Authenticate and authorise the verifier to get our information form the ADS

4 Share holder information

The new model for self-sovereign information exchange

Three-step self-sovereign information exchange enables holders to share their information with anyone they want



1 Request credentials

2 Present credentials

3 Verify credentials

Four standards to make the new information exchange possible

There is no single protocol to exchange Verifiable Credentials, few alternatives exist

OpenID for Verifiable Credentials

OpenID Foundation is standardising a family of specifications OpenID for Verifiable Credentials for self-sovereign Verifiable Credential issuance and presentation.

ISO 18013-5 (mDL)

ISO is standardising 18013-5 for offline mobile driver's license exchange and 23220* for online credentials exchange.

Credential Issuance and Present Proof

Hyperledger is standardising Issue Credential and Present Proof protocols.**

WACI

DIF is standardising Wallet and Credential Interaction (WACI**) protocol for Verifiable Credentials issuance and presentation.

* OpenID for VCs is part of the standard.
** The protocol is implemented on top of the DIDComm messaging protocol.

EBSI selected OpenID for Verifiable credentials protocols

EBSI use cases selected OpenID for Verifiable credentials protocols for online Verifiable Credentials issuance and presentation

OpenID for Verifiable Credentials

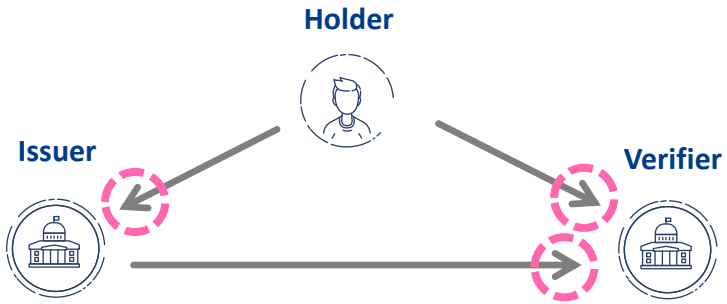
OpenID Foundation is standardising a family of open specifications OpenID for Verifiable Credentials for self-sovereign Verifiable Credential issuance and presentation.

These are open standards.

High-level of maturity, active and wide community, and protocols that are built on proven OID and OAuth industry standards.

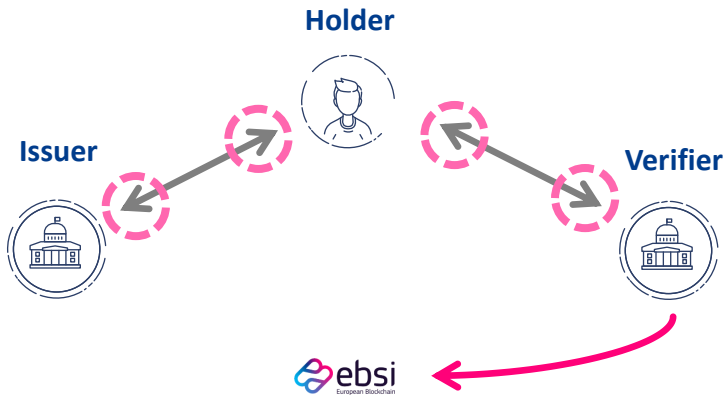
Comparing the popular OIDC with OID4VC?

Authorisation-based protocol versus Self-sovereign information exchange protocol. Both specifications are complementary as they aim solving different problems.



OAuth and OpenID Connect (OIDC)

Protocol that support authorisation-based credential exchange where the holder authorises a verifier (client) to access information on her behalf.



OpenID for VCs (OID4VC)

Protocol that supports self-sovereign credential exchange where the holder can autonomously control the exchange of credentials with any verifier she wants

OID4VCs is made of three standards. Two of them are used by EBSI

OpenID for Verifiable Credentials (OID4VCs) is a collection of three standards that enable a self-sovereign authentication, and Verifiable Credentials issuance and presentation. EBSI uses the ones for VC issuance and presentation.

1

Authentication

SIOPv2

Defines how holders can authenticate in a self-sovereign way with any actor

EBSI supports any other authentications and is not limited to SIOPv2 for holder authentication

2

Issuance

OpenID for Verifiable Credential Issuance (OID4VCI)

Defines APIs and the corresponding OAuth2-based authorisation mechanisms for the issuance of Verifiable Credentials.

3

Presentation

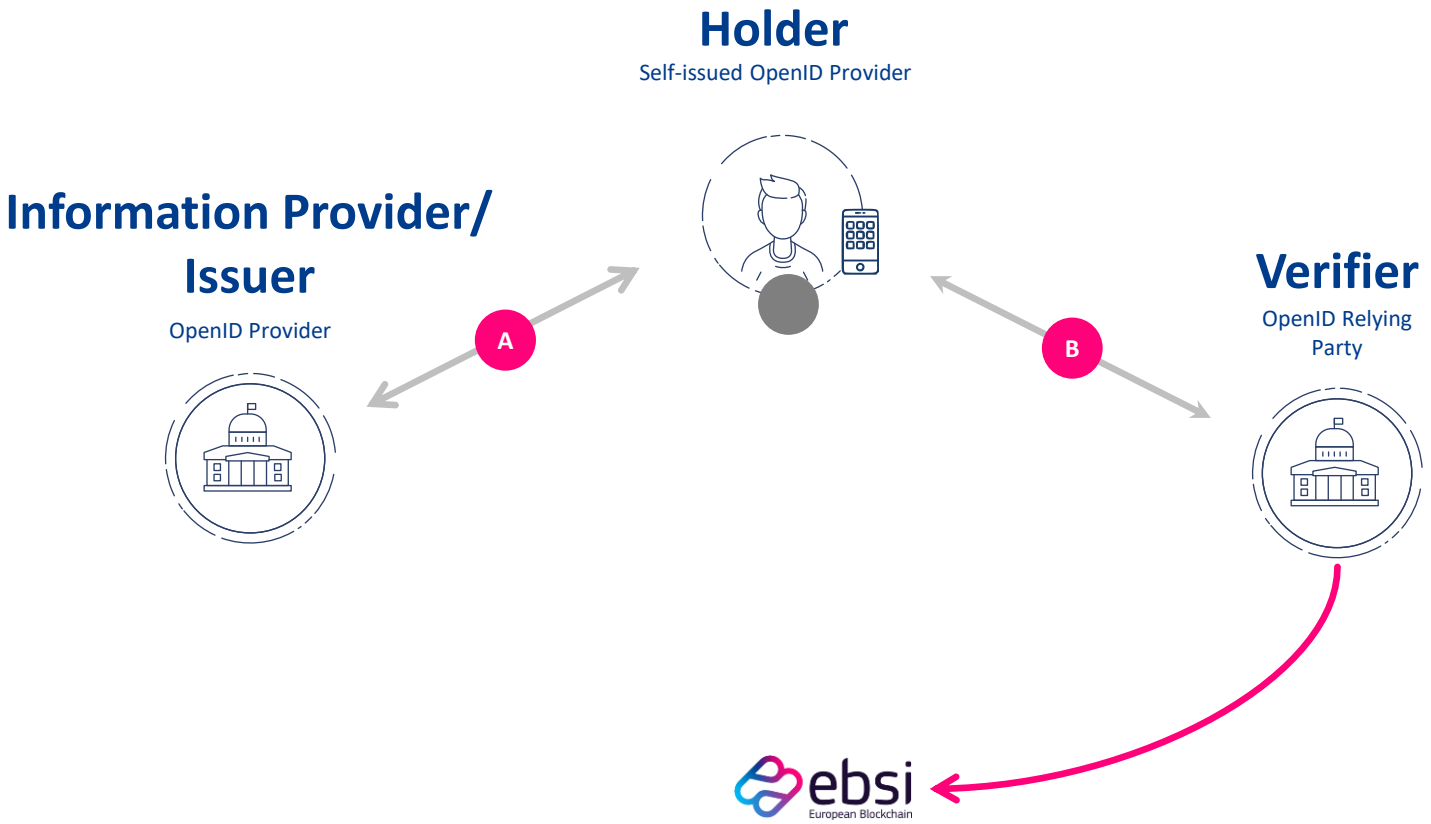
OpenID for Verifiable Presentations (OID4VP)

Defines mechanisms on top of OAuth2 to allow the presentation of claims in the form of Verifiable Credentials.

EBSI is only using OID4VCI and OID4VP for supporting the issuance and presentation of Verifiable Credentials.

Simplified view on scope of each standard.

How does it look like in a simplified view of the self-sovereign information exchange scenario?



SIOPv2 (optional, up to the wallet to use it)

A — OpenID for Verifiable Presentations (OID4VP)

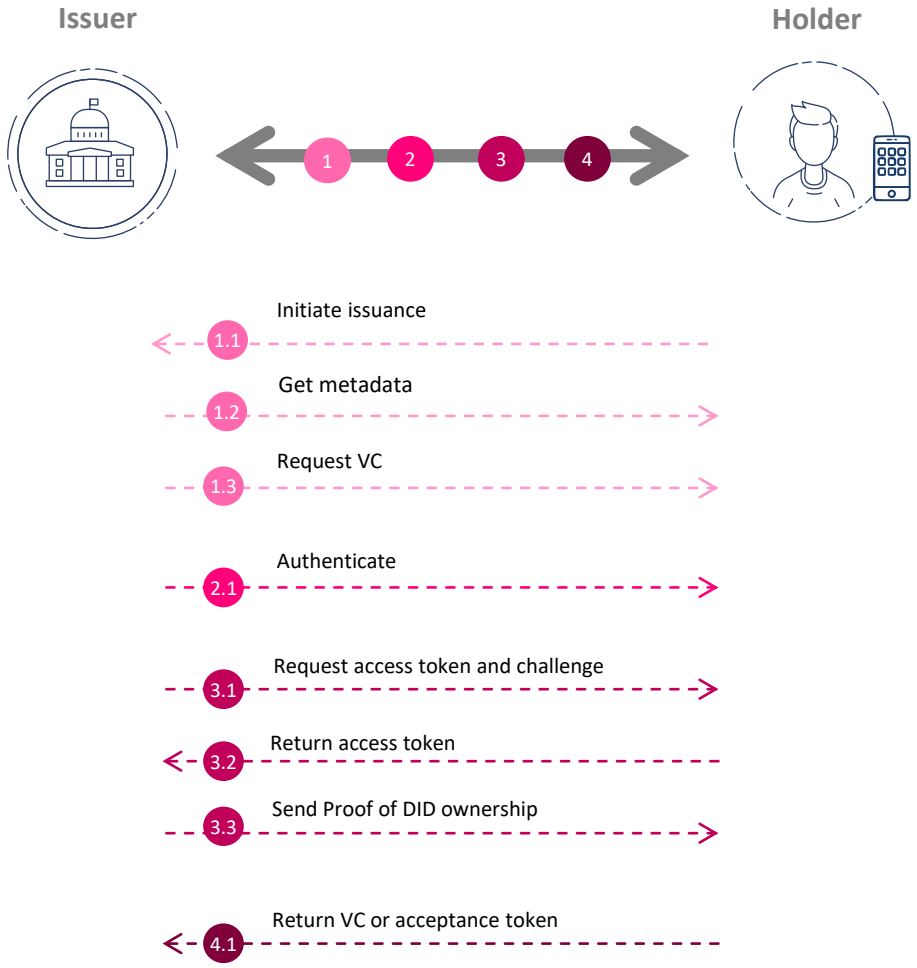
B — OpenID for Verifiable Credential Issuance (OID4VCI)

06.3

How does OpenID for Verifiable Credential Issuance work?

A. How does VC issuance work?

Verifiable Credential issuance consists of four key actions



1. Request VC

1.1 A holder initiates the issuance on the issuer's website, and the wallet receives information about the type of Verifiable Credential requested by the holder on the issuer's website via a QR code or a redirect to the wallet. This step is skipped if the user requests VC from the wallet.

1.2 Wallet obtains issuer metadata to learn about the supported flows, formats, signatures, and endpoints. OID4VCI extends the OAuth2 metadata.

1.3 Wallet requests a Verifiable Credential. The authorisation request is an extended OAuth2 authorisation request where the wallet can define the type and format of the VC and the signature type and format.

2. Authentication

2.1 The holder authenticates with the issuer via the authentication method supported by the issuer.

3. Issue VC

3.1 After a successful authentication, the wallet receives an OAuth2 code which it sends to the OAuth2 token endpoint receive an access token and a challenge to prove DID key control.

3.2 The issuer returns an access token and a challenge it is asked to sign.

3.3 The holder needs to sign the challenge to with her DID key(s) to prove control of the DID keys.

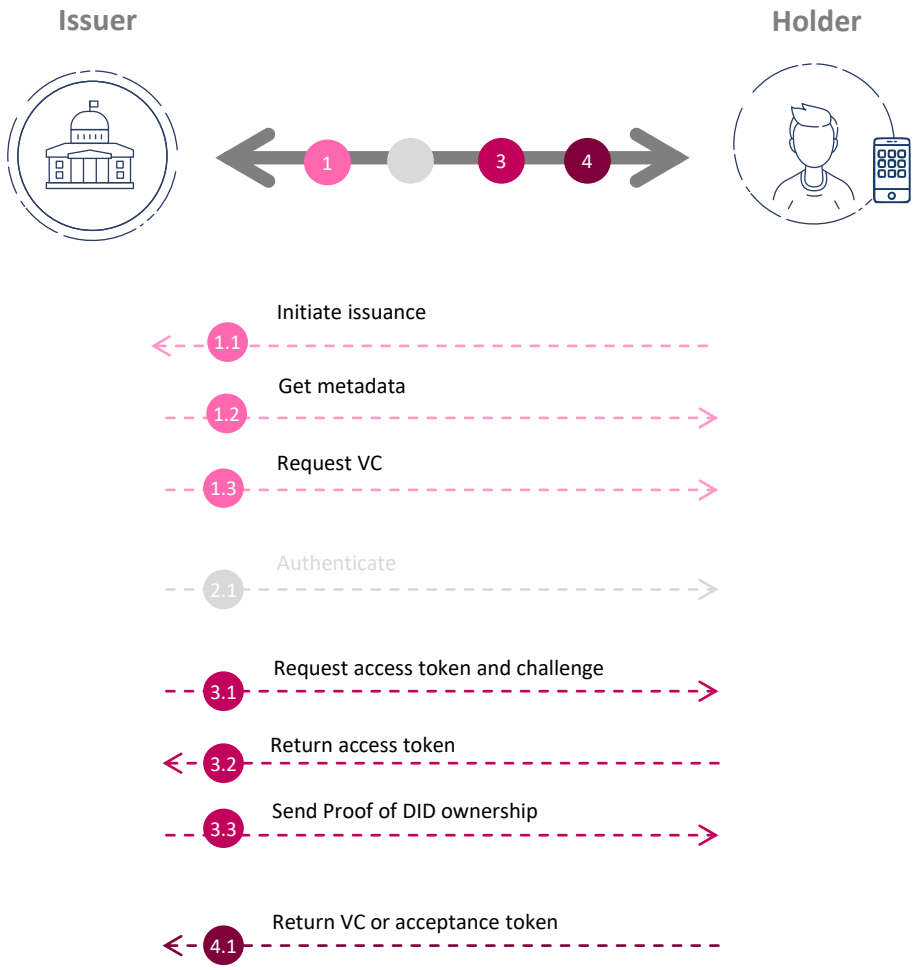
4. Collect VC

4.1 Issuer issues* a VC and notifies the wallet to collect it.

*The issuer can e-signs or e-seal the VC using eIdas e-seals. The issuance process can be just-in-time or deferred. In the latter case, the issuer returns an acceptance token the wallet can use to collect the VC once it is issued.

A. What OpenID4VCI standardises?

Verifiable Credential issuance consists of four key actions



1. Request VC

- A mechanism for the issuer to publish **metadata** about **supported VC types, formats, and signatures**
- Mechanisms to **initiate the issuance** (Via the issuer website and Via the wallet)
- Two verifiable credential **issuance flows (**pre-authorised flow and authorisation flow)
- **Authorisation request** that allows wallets to request authorisation to request issuance of Verifiable Credentials

2. Authentication

3. Issue VC

- A new OAuth2-protected **credential endpoint** for issuers where wallets collect the issued credentials
- **Mechanism to bind** the issued credentials to a cryptographic key or certificate

4. Collect VC VC

- A mechanism for **just-in-time or deferred VC issuance**

A. Cryptographic holder binding

Issuers can bind the Verifiable Credentials to DID after the holder proves it controls the corresponding private keys

1. Who is the holder requesting credentials?

As an issuer, I authenticate the holder to learn who the holder is.

2. Does the holder control the DID?

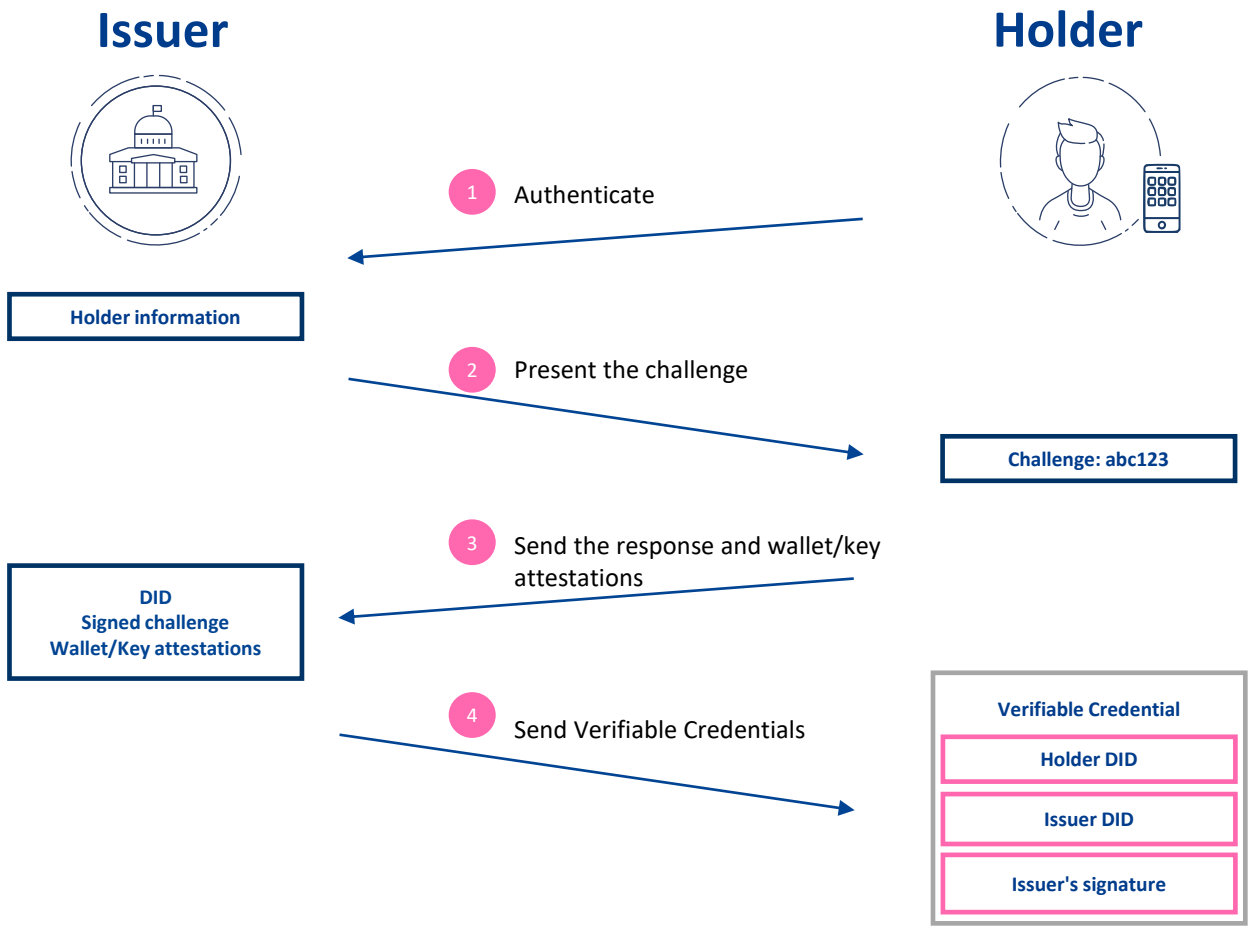
As an issuer, I create a challenge for the holder to prove she controls her DID.

3. Are holder's wallet and keys secure?

As an issuer, I verify the key and/or wallet attestations to ensure the DID keys are stored and managed in a wallet that meets the issuer's security requirements.

4. Issue Verifiable Credential(s) to the DID

As an issuer, I am sure the holder requesting the VC(s)controls the DID, hence I can issue a VC to that DID.

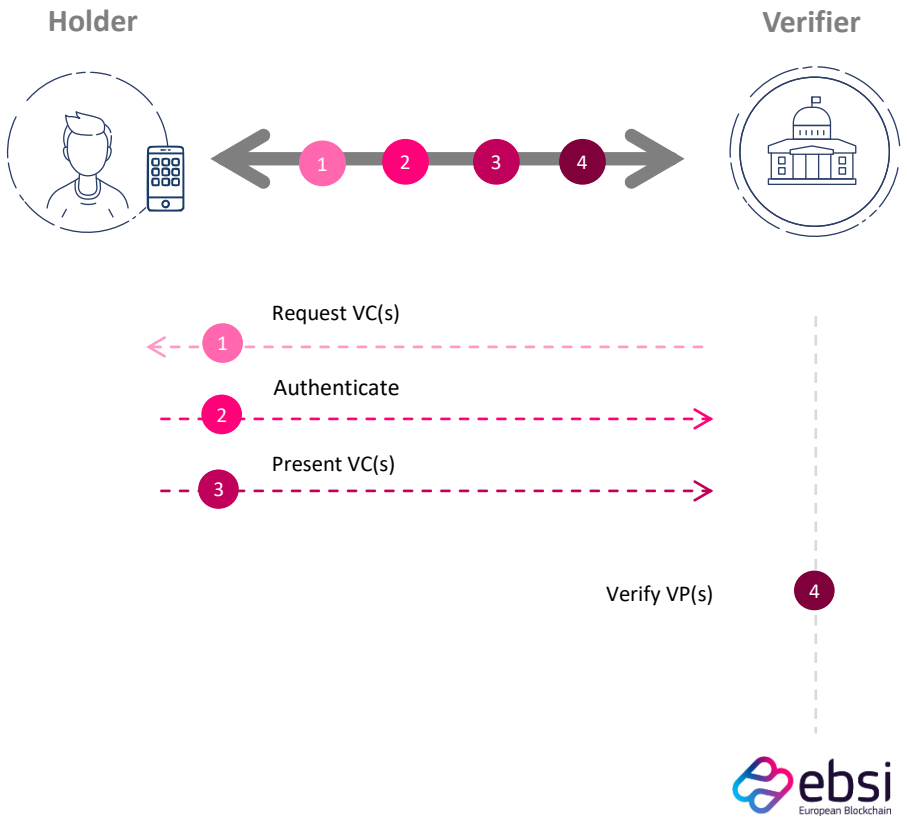


06.3

How does OpenID for Verifiable Presentations work?

B. How does VC presentation work?

Verifiable Credential presentation consists of four key actions



1. Request VC

Verifier **requests one or more VC(s)** from the holder. This is achieved by holder scanning a QR code (cross-device flow) with her wallet or via a redirect (same-device flow) to her wallet.

2. Authentication

The holder **authenticates** with the verifier via the authentication method provided by the verifier. Verifier can request a VC also after the holder is authenticated.

3. Present VC

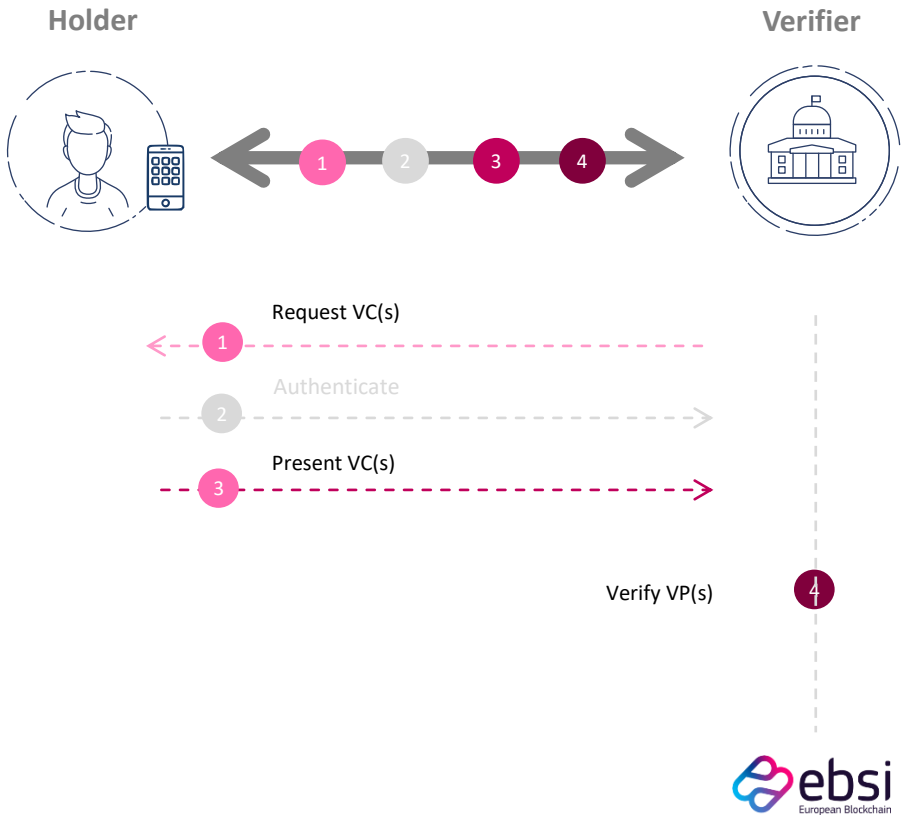
Wallet processes the VC sharing request and compiles and presents the Verifiable Presentation to the Verifier.

4. Verify VC

The Verifier verifies the Verifiable Presentation and Verifiable Credentials with the help of EBSI.

B. What OpenID4VP standardises?

Verifiable Credential presentation consists of four key actions



1. Request VC

- A mechanism for the verifiers to publish **metadata** about **supported VC types, formats, and signatures**
- A mechanism **initiate Verifiable Presentation exchange**

2. Authentication

The holder **authenticates** with the verifier via the authentication method provided by the verifier. Verifier can request a VC also after the holder is authenticated.

3. Present VC

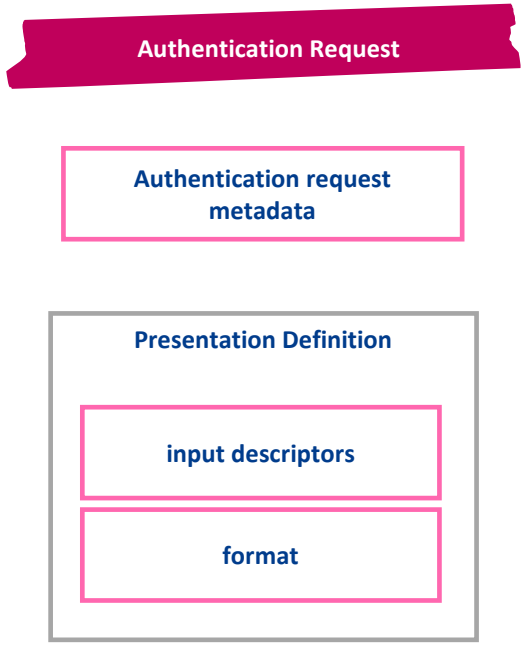
- Verifiable Presentation **flow**
- **Endpoints** for sharing Verifiable Credentials

4. Verify VC

The Verifier verifies the Verifiable Presentation and Verifiable Credentials with the help of EBSI.

B. How wallet knows which VC(s) to share?

Verifiable credentials can be requested by type or contextual criteria using Presentation Exchange expression language



Authentication request metadata

Authentication request metadata contains the standard OAuth2 authentication request claims so that the wallet can learn everything about the verifier (endpoints, supported formats, signatures, etc.), and to protect the presentation flow.

Presentation Definition

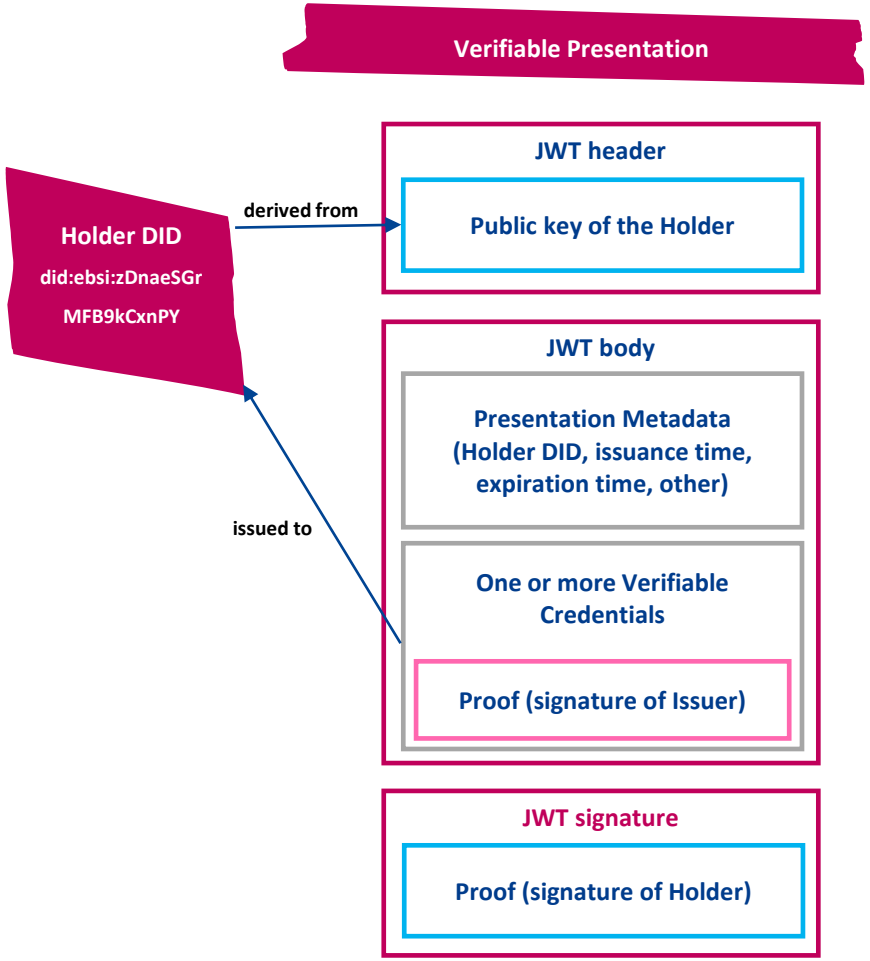
Presentation Definition is part of the Presentation Exchange expression language that enables to request one or more Verifiable Credentials by credential type or contextual criteria.

Input descriptors are used to define what information is requested by the Verifier. The verifier can request a specific VC by type by referencing the JSON schema (e.g., in the EBSI Trusted Schemas Registry) or can request VC(s) by specifying which claims the VC(s) must have (e.g., name, surname, address).

Format may be used to specify the required Verifiable Credential and Presentation formats.

B. How Verifiers verify the Verifiable Presentation(s)?

Verifiable Presentations are self-issued and self-contained



Header

Public key in the header is holder's DID public key that must verify the VP signature.

Body

Presentation metadata contains information about the holder's DID which must be derived from the public key shared in the header, and other standard VP claims.

One or more Verifiable Credentials are embedded in the VP. Verifiable Credentials must be issued to the same DID as in the presentation metadata. If VCs are issued to multiple DIDs, the holder should present multiple Verifiable Presentations.

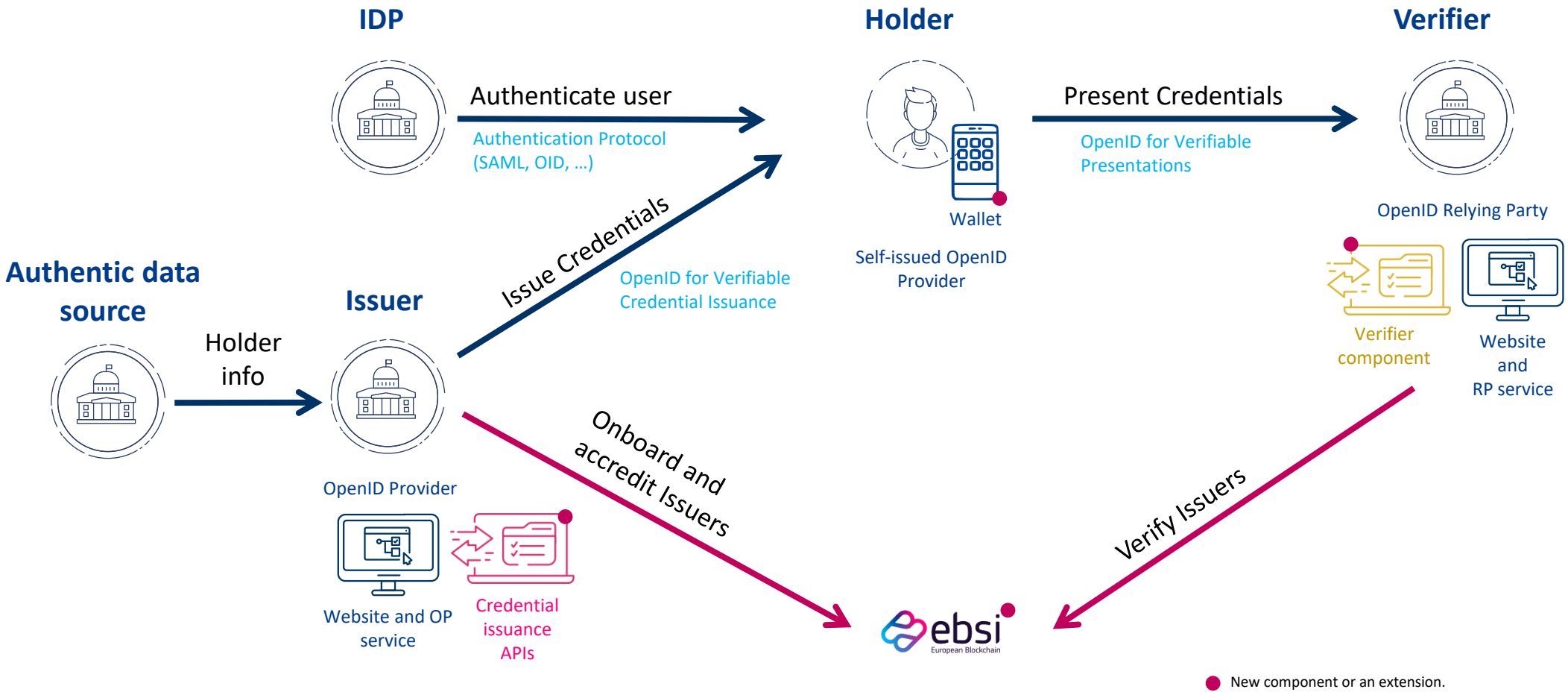
Signature

Proof is holder's signature of the Verifiable Presentation. The public key in the header must verify the signature.

A last question?

OID4VC is replicable across sector

Roles and components are the same across all sectors

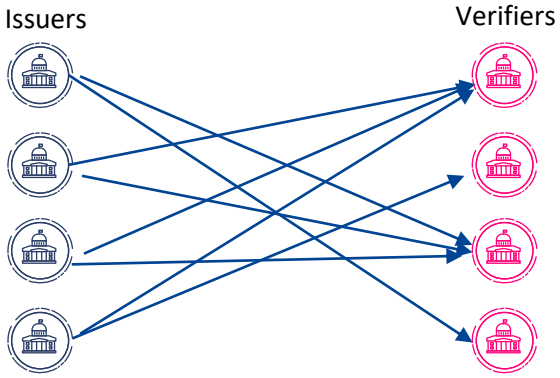


OID4VC is compatible with the EBSI issuer trust model

Issuers and Verifiers can create a trusted relationship via bilateral agreements or trust anchors/ trust model

Issuers and verifiers can establish relationship via

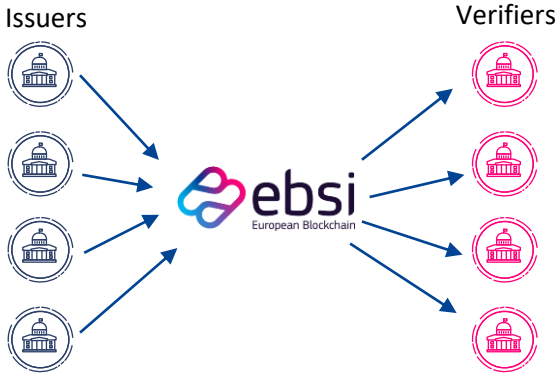
bilateral agreements



- \+ Good, if there are few issuers
- \+ Simple to design
- \- Requires custom set-up
- \- Use-case specific
- \- Hard to scale

Issuers and verifiers can establish relationship via

trust anchors/ trust model



- \+ Good, if there are many issuers and verifiers
- \- Challenging to design
- \+ No custom set-up
- \+ Applicable to a wide range of use-cases
- \+ Easy to scale (both horizontal and vertical scaling)

Want to know more?

Key resources

Explore EBSI

Explore the EBSI website

https://ec.europa.eu/digital-building-blocks/wikis/display/EBSI/Home

Check the specs

Check the EBSI Playbook

https://ec.europa.eu/digital-building-blocks/wikis/display/EBSIDOC/EBSI+Verifiable+Credentials+Playbook

Watch the demos

Watch the EBSI Demo Day

https://ec.europa.eu/digital-building-blocks/wikis/display/EBSI/EBSI+Demo+Day

https://ec.europa.eu/ebsi