EBSI Verifiable Credentials explained

CHAPTER 3

EBSI DIDs

June 2022

European Commission | ebsi European Blockchain

EBSI, explained – first edition

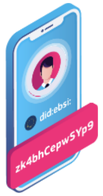
What are the different chapters of this first edition?

01.

Verifiable Credentials Explained

02.

Verifiable Credentials in action



03.

Decentralised Identifiers (DID) Methods

04.

Digital Identity

05.

Issuers Trust Model

06.

Open ID Connect for Verifiable Credentials

07.

Digital Wallets

03. EBSI DID methods explained – Index

What are you going to learn in this Chapter?

03.1

What is a DID and why two DID methods in EBSI?

03.2

How does the DID method v1 work? (Legal persons)

03.3

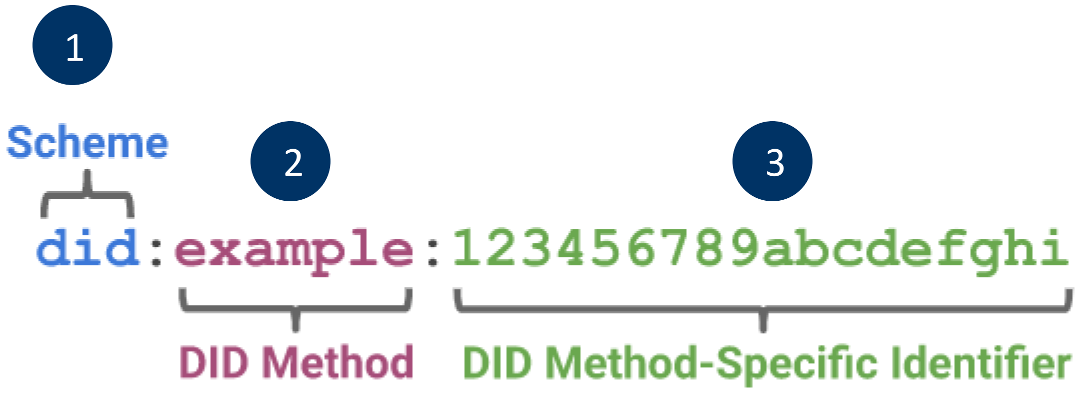
How does the DID method v2 work? (Natural persons)

03.1

What is a DID and why two DID methods in EBSI?

What is a W3C Decentralised Identifier (DID)?

A DID is just a long string that does not provide any meaningful information about a natural or legal entity. DIDs and DID Documents are generated by their owners with their wallet or back-office systems.


1
Scheme
2
3
did:example:123456789abcdefghi
DID Method
DID Method-Specific Identifier

According to the W3C standard, a DID is always made of three parts:

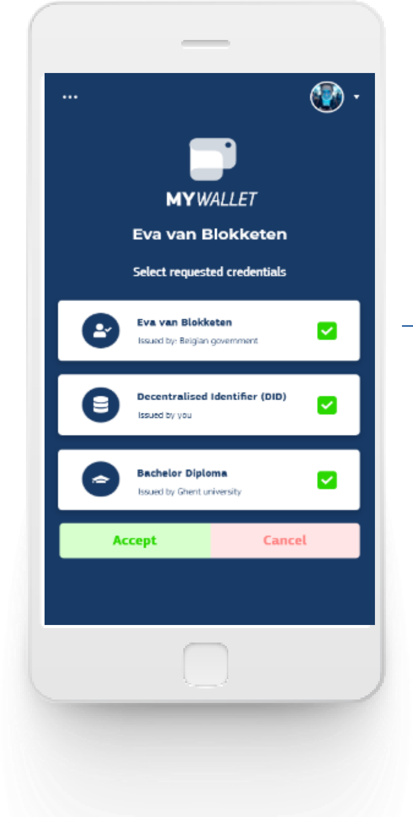
1. the first part is **always the three letters “did”.**
2. the second part defines the identifier for the DID method, .
3. the third field is a **completely unique random number that follows** method-specific generation rules.

Why it is important?

DIDs are used to ensure the authenticity of issuers and holders in machine verifiable documents known as Verifiable Credentials (VCs).

Wallet

Verifiable Credentials in a Digital Wallet



MY WALLET

Eva van Blokketen

Select requested credentials

- Eva van Blokketen ☑
  Issued by: Belgian government
- Decentralised Identifier (DID) ☑
  Issued by you
- Bachelor Diploma ☑
  Issued by Ghent university

Example of an Verifiable Credential – W3C Specification

```
{
 "@context": [
   "https://www.w3.org/2018/credentials/v1",
   "https://essif.europa.eu/schemas/vc/2020/v1"
 ],
 "id": "https://essif.europa.eu/tsr/53",
 "type": [
   "VerifiableCredential",
   "VerifiableAttestation",
   "VerifiableAccreditation",
   "DiplomaVerifiableAccreditation"
 ],
 "issuer": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
 "issuanceDate": "2020-06-22T14:11:44Z",
 "credentialSubject": {
   "id": "did:ebsi:zDnaeSGrMFB9kCxnPYWaeMrRyun2HLVHjDNUf76ccy4ZfHU24",

  (....)
```

DID of the Issuer

DID of the Holder

EBSI has two DID methods

What are the different DID methods supported by EBSI and why?

v1

Designed for frequent key rotation, DID documents stored on EBSI ledger

EBSI DID method specification v1 oriented for Legal Persons

v2

Designed for full privacy preservation, DID documents only stored on the wallet

EBSI DID method specification v2 oriented for Natural Persons

What are the differences between the DID methods

One method is oriented for legal persons (Issuers) and the other for natural persons (Holders)

	v1 EBSI DID method specification v1 oriented for Legal Persons	v2 EBSI DID method specification v2 oriented for Natural Persons
By whom it is used?	• To be used for **Legal Persons (Issuers).**	• To be used for **Natural Persons** because no information is kept in EBSI's ledger.
How is it generated?	• DID and DID document are **generated by a back-office application** or **a wallet-like application**.	• DID and DID document are **generated and stored on the wallet**.
Where is it recorded?	• DID document is **recorded on EBSI's ledger**. No coupling between DID and Public Key, enabling frequent key rotation by Issuers.	• DID document **not recorded on EBSI's ledger as** DID ownership can be cryptographically verifiable because it contains a JWK thumbprint of the Public Key – hence if holder proves ownership of the private key, it proves ownership of the DID.
How does it work?	• Verifiers retrieve the DID document from EBSI to confirm ownership of DIDs and to verify the signature of Verifiable Credentials using the Issuer's public key for assertion.	• The wallet includes the DID and DID document when presenting information to Verifiers or when asked to confirm the DID ownership by Verifiers or by Issuers.

Overview of EBSI DID methods

Overview of EBSI DID methods

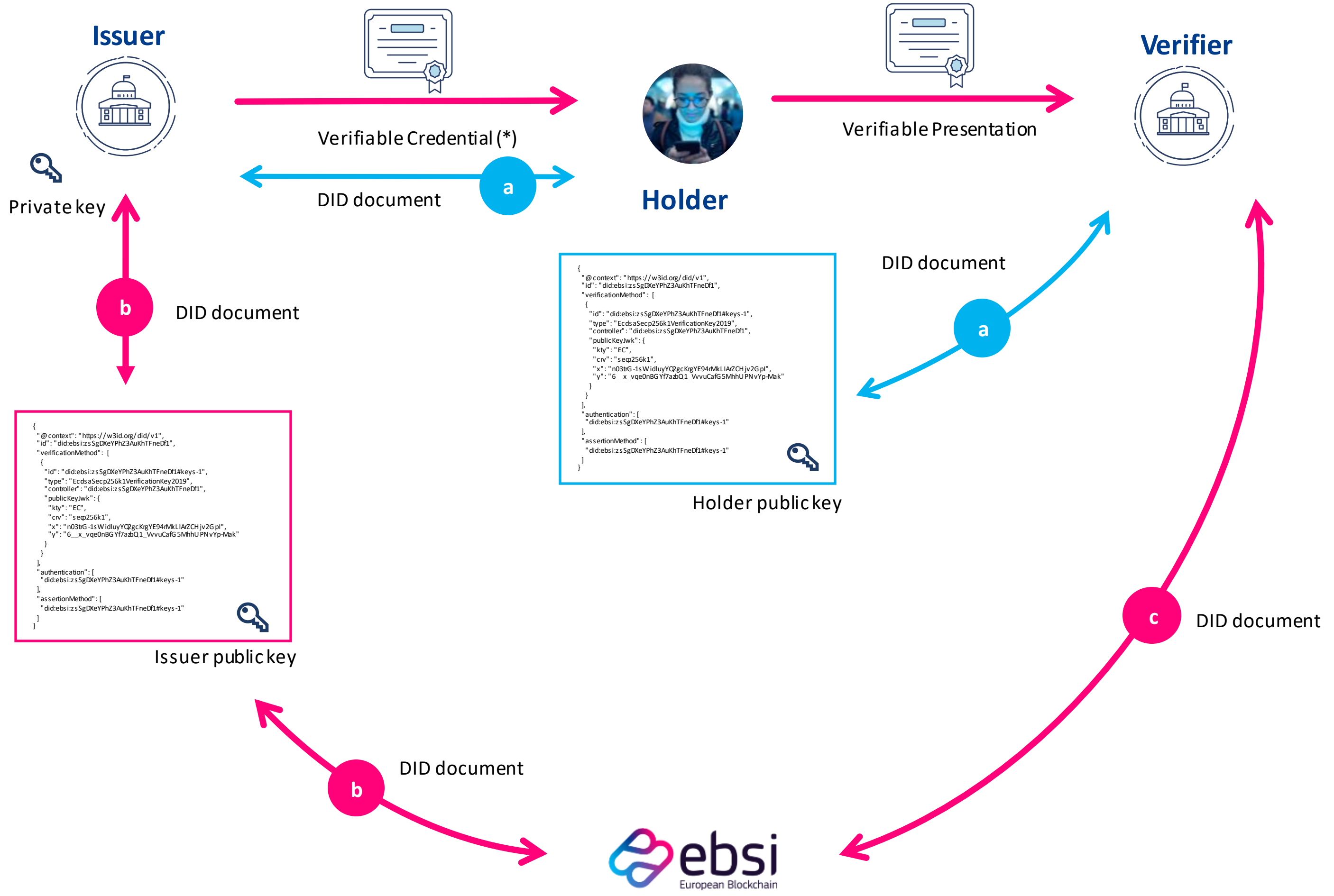
a DID Documents of Natural Persons are provided by the wallet

DID documents or DIDs of Natural Persons <u>are not</u> recorded on EBSI.

b *DID documents of Legal Entities are recorded on EBSI. See EBSI DID method specification v1*

c Verifiers retrieve DID Documents of Issuers/ Legal Persons from EBSI using a link such as: https://api.test.intebsi.xyz/did-registry/v2/identifiers/did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1

(*) Issuers must also be able to receive DID documents from user/holder wallets, at the time of issuing Verifiable Credentials, to confirm DID ownership.



```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
  "verificationMethod": [
    {
      "id": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "controller": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
      "publicKeyJwk": {
        "kty": "EC",
        "crv": "secp256k1",
        "x": "n03trG-1sWidluyYQ2gcKrgYE94rMkLIArZCHjv2GpI",
        "y": "6__x_vqe0nBGYf7azbQ1_VvvuCafG5MhhUPNvYp-Mak"
      }
    }
  ],
  "authentication": [
    "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1"
  ],
  "assertionMethod": [
    "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1"
  ]
}
```

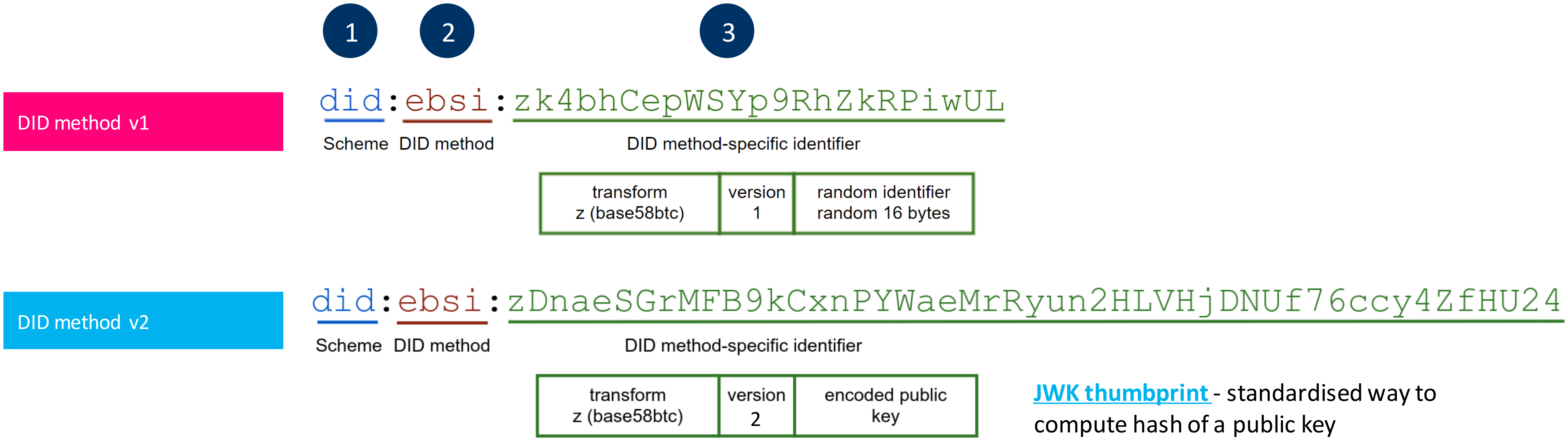
Issuer public key

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
  "verificationMethod": [
    {
      "id": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "controller": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
      "publicKeyJwk": {
        "kty": "EC",
        "crv": "secp256k1",
        "x": "n03trG-1sWidluyYQ2gcKrgYE94rMkLIArZCHjv2GpI",
        "y": "6__x_vqe0nBGYf7azbQ1_VvvuCafG5MhhUPNvYp-Mak"
      }
    }
  ],
  "authentication": [
    "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1"
  ],
  "assertionMethod": [
    "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1"
  ]
}
```

Holder public key

Let’s look back at EBSI’s DIDs

The structure is made of three parts in both methods but the DID method v2 will use a standardised way to compute hash of a public key



DID method v1

did:ebsi:zk4bhCepWSYp9RhZkRPiwUL

Scheme DID method DID method-specific identifier

transform z (base58btc)	version 1	random identifier random 16 bytes

DID method v2

did:ebsi:zDnaeSGrMFB9kCxnPYWaeMrRyun2HLVHjDNUf76ccy4ZfHU24

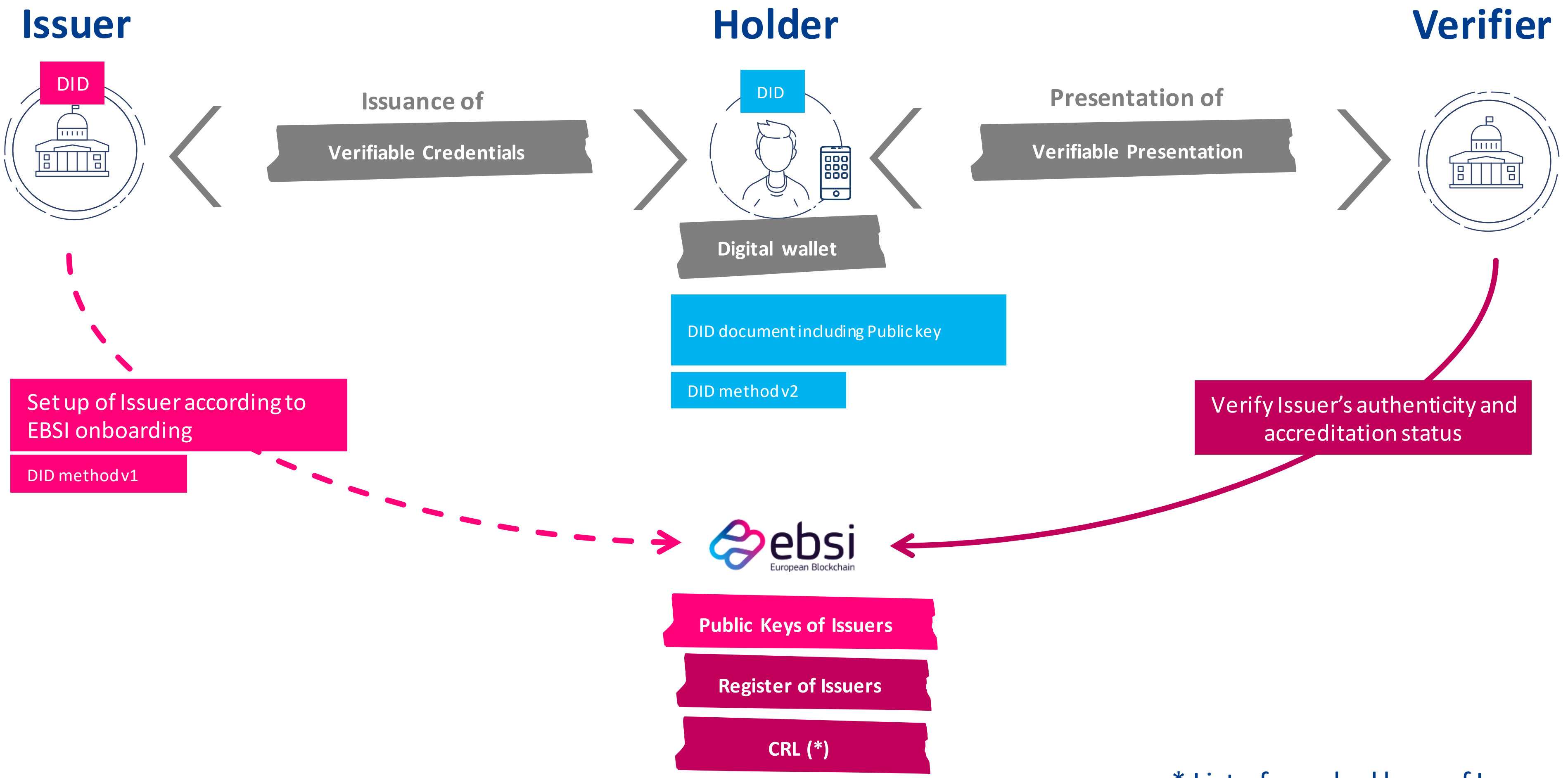
Scheme DID method DID method-specific identifier

transform z (base58btc)	version 2	encoded public key

JWK thumbprint - standardised way to compute hash of a public key

The EBSI DID methods applied

The DID methods applied to the basic information exchange scenario



DID document including Public key

DID method v2

Set up of Issuer according to EBSI onboarding

DID method v1

Verify Issuer's authenticity and accreditation status

Public Keys of Issuers

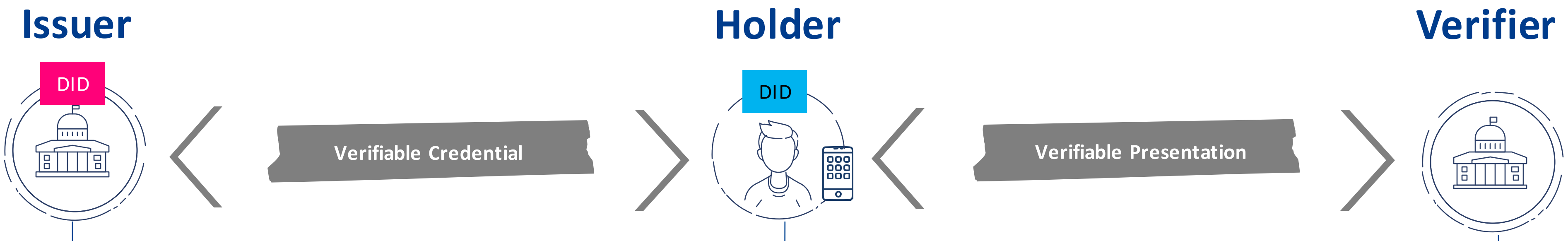
Register of Issuers

CRL (*)

* List of revoked keys of Issuers

How does it work?

Step 0. Issuers are onboarded, wallets are setup and verifiers apps created



Issuer

DID

Verifiable Credential

Holder

DID

Verifiable Presentation

Verifier

DID method v1

- Registration of self-issued Public Keys and DID (as part of a DID document) on EBSI
- Accreditation of issuers of VCs by Trusted Accreditation Authorities (TAOs)
- Registration of Issuers in EBSI's issuer register

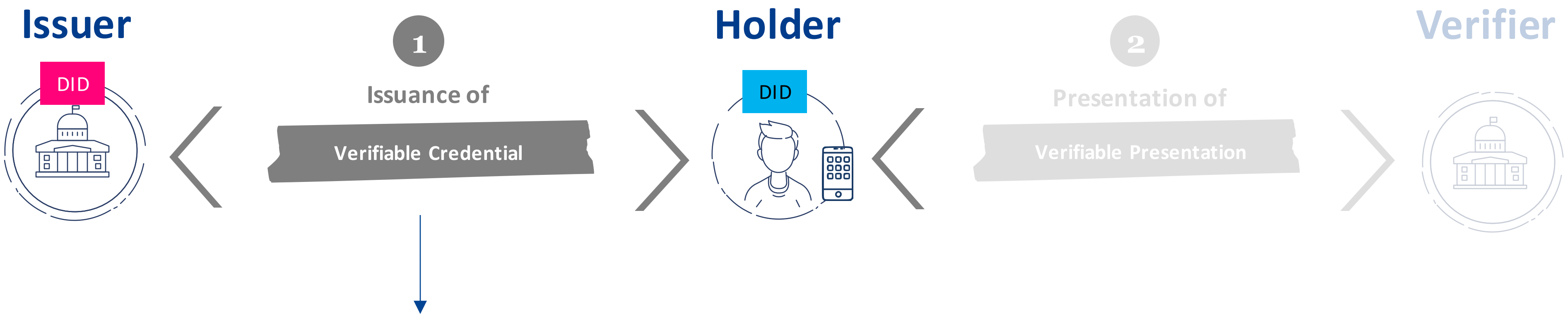
DID method v2

- Selects an EBSI conformant wallet
- Wallet generates a DID, a Public and a Private key. All of them part of a DID document stored on the wallet
- Request VCs from issuers

- No specific onboarding or set-up apart from the creation of a Verifier App that can exploit EBSI's APIs and interact with EBSI conformant wallets

How does it work?

Step 1. Issuance of a Verifiable Credential which is then stored on an EBSI conformant wallet



What does it contain?

Credential Metadata

> The DID of the entity that issues the credential
> The status of the credential (Issuance Date, Expiry date)

Claim(s)

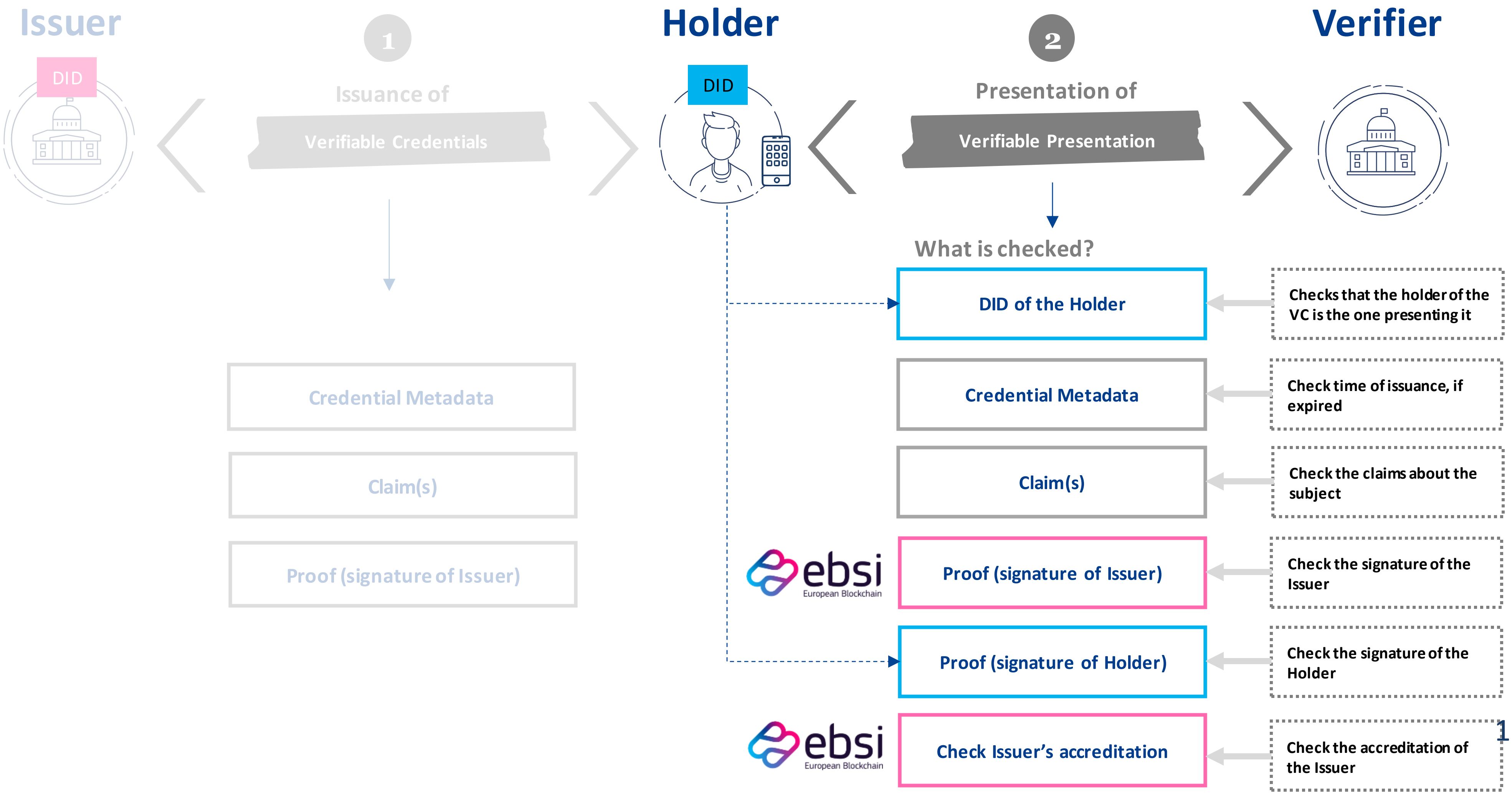
> The DID of the Holder of the credential
> The claims about the subject (What the issuer asserts about the subject)

Proof (signature of Issuer)

> Digital proof to make the credential tamper-evident (One or more cryptographic proofs that can be used to detect tampering and verify the authorship of a credential).

How does it work?

Step 2. Presentation of a Verifiable Credential for verification



Issuer

DID

1

Issuance of

Verifiable Credentials

Credential Metadata

Claim(s)

Proof (signature of Issuer)

Holder

DID

2

Presentation of

Verifiable Presentation

Verifier

What is checked?

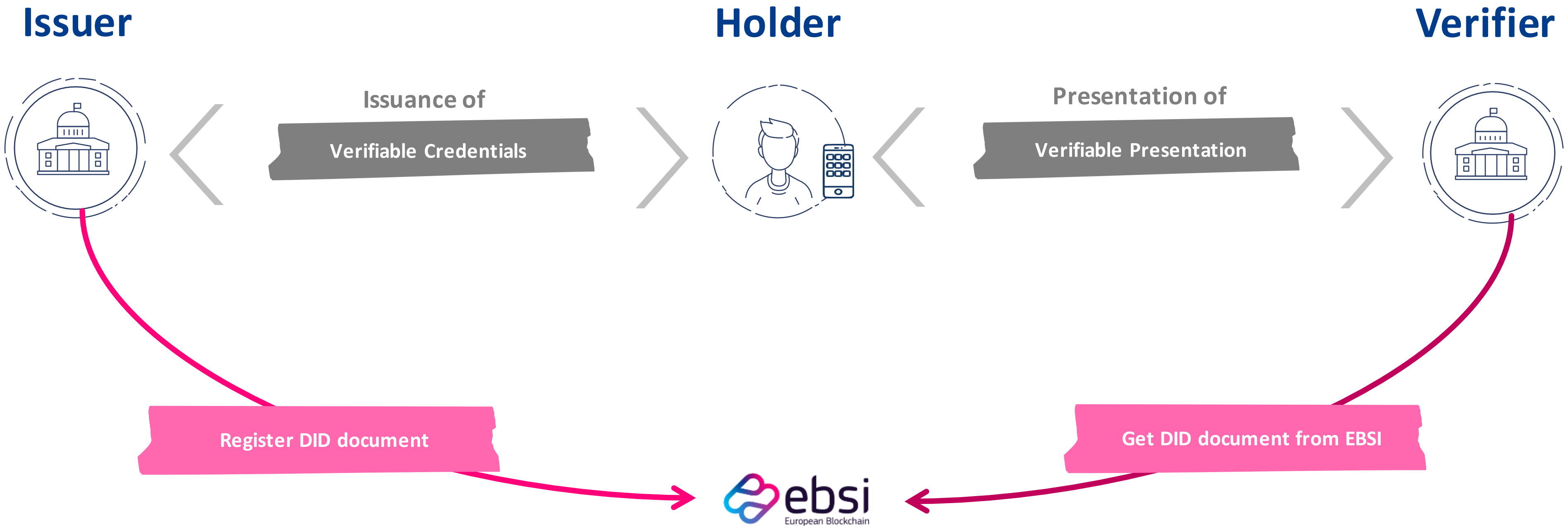
DID of the Holder	Checks that the holder of the VC is the one presenting it
Credential Metadata	Check time of issuance, if expired
Claim(s)	Check the claims about the subject
Proof (signature of Issuer)	Check the signature of the Issuer
Proof (signature of Holder)	Check the signature of the Holder
Check Issuer’s accreditation	Check the accreditation of the Issuer

03.2

How does the DID method v1 work?

EBSI DID method specification v1 for legal persons > ISSUERS

Simplified conceptual flow

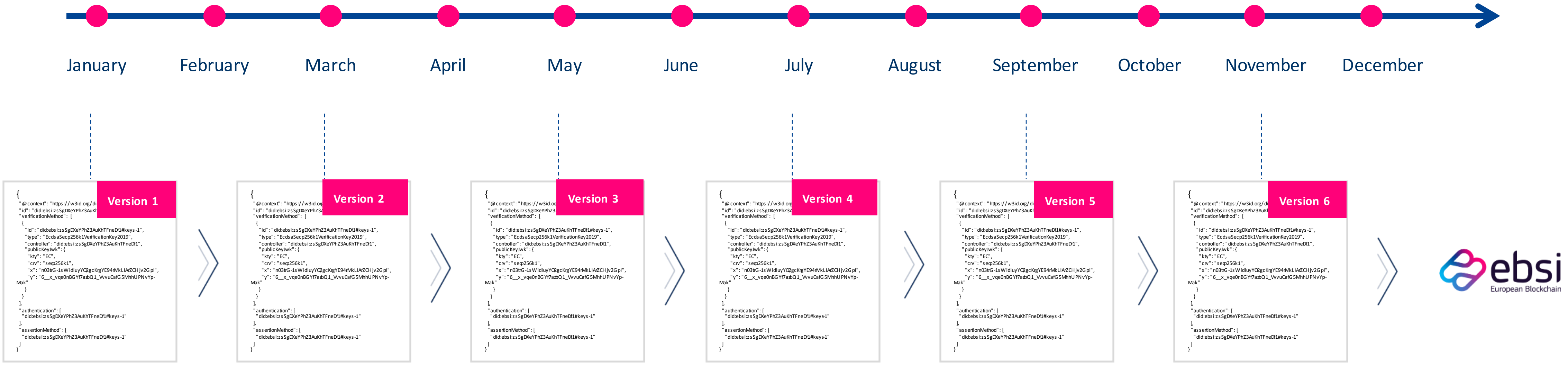


- The Issuer creates DIDs according to EBSI's DID scheme profile (did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1).
- The Issuer also creates the cryptographic keys associated to a given DID.
- The Issuer records this information on EBSI in the form of a DID document.
- The DID document can be retrieved from EBSI by Issuers and Verifiers using a simple URL).

DID method v1 enables Issuers to flexibly manage their keys and their real-time access by Verifiers

Issuer

The use of DIDs and DID documents registered on EBSI, as defined in DID method v1. enables Issuers to rotate their keys, i.e., to update their cryptographic keys regularly (e.g. every other month) without impacting the Verifiers as they can easily retrieve the right version of the DID document from EBSI. This enables a much smoother and secure management of keys in large ecosystems. Furthermore, issuers can have multiple active keys bound to their DID.

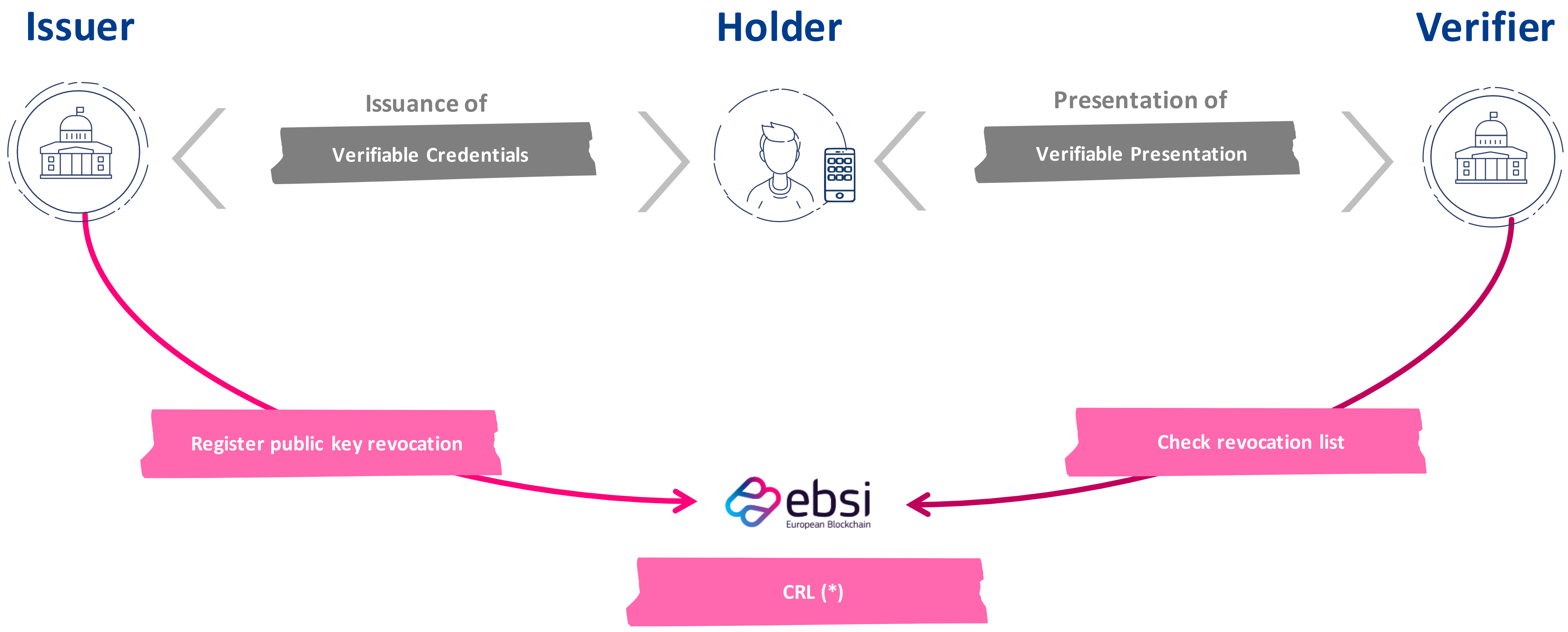


```
{
  "@context": "https://w3id.org/di
  "id": "did:ebsi:zsSgDXeYPhZ3AuKh
  "verificationMethod":  [
    {
      "id": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "controller": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
      "publicKeyJwk": {
        "kty": "EC",
        "crv": "secp256k1",
        "x": "n03trG-1sWidluyYQ2gcKrgYE94rMkLIArZCHjv2Gpl",
        "y": "6__x_vqe0nBGYf7azbQ1_VvvuCafG5MhhUPNvYp-
Mak"
      }
    }
  ],
  "authentication": [
    "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1"
  ],
  "assertionMethod": [
    "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1"
  ]
}
```

Important Note! Rotation of keys minimises the number of Verifiable Credentials revoked because of the revocation of the Issuer’s signing keys.

Revocation of Issuer’s keys

Assuming that an issuer issues 20 credentials every month, 240 credentials per year, and changes its key pair every other month. Should a key pair be comprised, the one of March/ April, the issuer would be required to re-issue about 40 credentials when revoking the key pair instead of the 240 credentials if the Issuer would have used the same keys during the whole year.



* List of revoked keys of Issuers

DID lifecycle of legal persons

DID lifecycle of legal persons

01

Creation. In the background, by a back-office application or a wallet-like application:

- Creates DID and
- Private and public key of the DID control key
- Creates an EBSI ledger address derived from the public key of DID control key
- Creates the additional keys and the
- DID document (including public key of DID control key).

02

Registration of hash of DID document

Wallet registers hash of DID document on EBSI. To do so:

- DID document is shared with EBSI so that EBSI can check that the issuer has all private keys associated to the public keys shown in the DID document (DID document is not persisted).
- If all controls are passed, the DID document is registered on EBSI.

03

Update of DID document

DID document is updated with new public keys back-office application or a wallet-like application.

04

Registration of updated DID document

Updated DID document is registered on EBSI following similar controls at creation step.

05

Update DID document to deactivate DID

Creation of DID document without any keys in it using a back-office application or a wallet-like application.

06

Registration of updated DID document

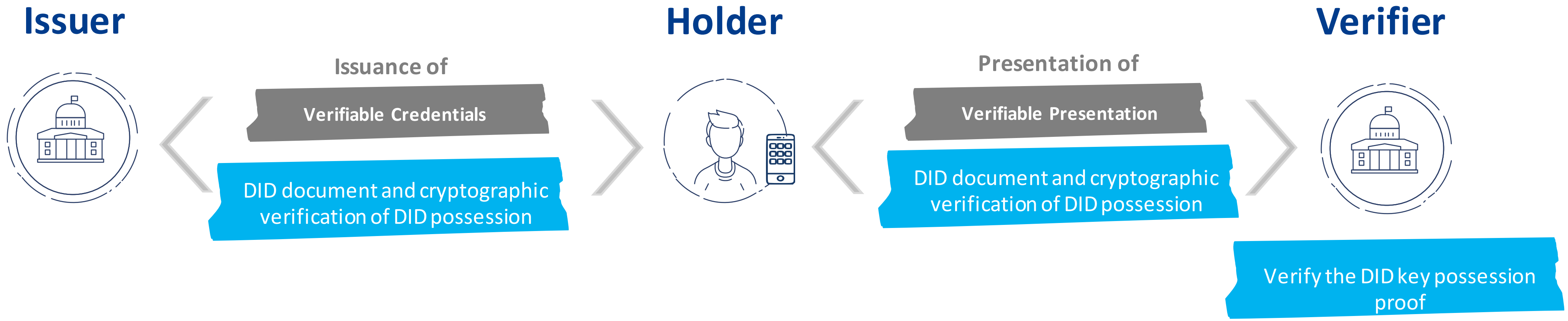
DID Document without public keys is registered on EBSI following similar controls at creation step.

03.3

How does the DID method v2 work?

EBSI DID method specification v2 for natural persons

Simplified conceptual flow



Issuer

Issuance of

Verifiable Credentials

DID document and cryptographic verification of DID possession

Holder

Presentation of

Verifiable Presentation

DID document and cryptographic verification of DID possession

Verifier

Verify the DID key possession proof

- The wallet creates the cryptographic keys and derives the DID according to EBSI's DID scheme profile V2 by encoding the Public Key (did:ebsi:zDnaeSGrMFB9kCxnPYWaeMrRyun2HLVHjDNUf76ccy4ZfHU24) - JWK thumbprint - standardised way to compute hash of a public key.
- In the Verifiable Credential issuance process, the Holder shares the DID document (public key) and proves the possession of the DID by confirming possession of the corresponding private key to the Verifier
- In the Verifiable Presentation exchange process, the holder shares her DID document (public key) and proves the possession of the DID by confirming possession of the corresponding private key to the Verifier

Want to know more?

Key ressources

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

Annex

Key acronyms and terms used in this document

Key acronyms and terms used in this document

- **Decentralised identifier (DID):** A portable URL-based identifier, also known as a DID, associated with an entity. These identifiers are most often used in a verifiable credential and are associated with subjects such that a verifiable credential itself can be easily ported from one repository to another without the need to reissue the credential.
- **Decentralised identifier document (DID document):** Also referred to as a DID document, this is a document that contains information related to a specific decentralized identifier, such as the associated repository and public key information.
- **Issuer:** A role an entity can perform by asserting claims about one or more subjects, creating a verifiable credential from these claims, and transmitting the verifiable credential to a holder.
- **Verifiable Credential (VC):** A set of one or more claims made by an issuer. A verifiable credential is a tamper-evident credential that has authorship that can be cryptographically verified. Verifiable credentials can be used to build verifiable presentations, which can also be cryptographically verified.
- **Verifiable Presentation (VP):** Data derived from one or more verifiable credentials, issued by one or more issuers, that is shared with a specific verifier. A verifiable presentation is a tamper-evident presentation encoded in such a way that authorship of the data can be trusted after a process of cryptographic verification.
- **Verifiable data registry:** A role a system might perform by mediating the creation and verification of identifiers, keys, and other relevant data, such as verifiable credential schemas, revocation registries, issuer public keys, and so on, which might be required to use verifiable credentials.
- **Verifier:** A role an entity performs by receiving one or more verifiable credentials, optionally inside a verifiable presentation for processing. Other specifications might refer to this concept as a relying party.

Source: W3C

Technically speaking, what is a DID document?

Every DID is matched to a single and unique DID document which can be versioned.

- Every Decentralised Identifier (DID) is associated to the Public Keys used by Verifiers for verification of electronic Signatures in a DID document.

- A DID document contains the cryptographic public keys used to verify Verifiable Credentials.

- **According to DID method v1,** Issuers must have a DID document stored on EBSI that they can manage.

- According to DID method v2, Holders do not have a DID document on EBSI. Their DID document is stored and shared by the wallet.

Example of an EBSI DID document

```
{
 "@context": "https://w3id.org/did/v1",
 "id": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
 "verificationMethod": [
   {
    "id": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1",
    "type": "EcdsaSecp256k1VerificationKey2019",
    "controller": "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1",
    "publicKeyJwk": {
     "kty": "EC",
     "crv": "secp256k1",
     "x": "n03trG-1sWidluyYQ2gcKrgYE94rMkLIArZCHjv2GpI",
     "y": "6__x_vqe0nBGYf7azbQ1_VvvuCafG5MhhUPNvYp-Mak"
    }
   }
 ],
 "assertionMethod": [
   "did:ebsi:zsSgDXeYPhZ3AuKhTFneDf1#keys-1"
 ]
}
```

Public key

Reference to Public key

What is a public / private key?

What is a public / private key and when it is used?

What is a public / private key?

Public / private key cryptography uses a pair of keys:

- a **public key** and a private key that are mathematically related to each other (but not associated with the DID).
- the **private key** must remain secret and cannot be shared (e.g., it must stay in the wallet of the Holder).
- the **public key(s)** used by Issuers and Holders are made public in the DID document without reducing the security of the process.

When is it used?

Electronic signatures use public and private keys to enable trust between Issuers and Verifiers also between Holders and Verifiers:

- **When created, Verifiable Credentials are signed by Issuers (using their private key) and checked by Verifiers (using the public key in the DID document of a given Issuer) to ensure their integrity and authenticity.**
- **When sharing information, Verifiable Presentations are signed by Holders (using their private key) and checked by Verifiers (using the public key in the DID document of the Holder) to ensure their integrity and authenticity**.

What is a DID registrar / verifiable DID registry?

The use of DID requires an underlying registrar system which may be a distributed ledger, decentralised file system, database, or any other form of trusted data storage. **EBSI is the registrar of all EBSI DIDs**.

The DID registrar is only used in DID method v1 for DID documents of Issuers.

DID Scheme and DID Method

EBSI defines the DID scheme and the DID method specification including how :

1. Verifiers can resolve DIDs and obtain DID Document(s) of Issuers from EBSI so that:
   - Verifiers can obtain the latest version.
   - Verifiers can obtain any previous revision of the DID Document.
2. Verify that DIDs comply with EBSI's DID schema.
3. Registration of DID Documents (including any subsequent updates).
4. Deactivate DIDs.

What does this ensure?

- The **uniqueness** of DIDs of Issuers.
- **Non-repudiation and immutability** of the DIDs and DID Documents of Issuers.
- That the same controlling key is **NOT registering two different DIDs**.
- That **only the controlling key of a specific Issuer** can manage the DID.

What is a DID control key?

What is a DID control key, by whom it is used and why?

The DID control key is only used in DID method v1 for managing DID documents of Issuers.

DID control key is a key pair that is used by

- **Issuers** to register, update or deactivate their DID Documents (which include the public key of the DID control key).

- **Natural persons** do not have one because their DID document is not registered on EBSI.

Why the DID control key is used?

The **private keys of DID control keys are used to sign transactions that register, update and deactivate DIDs on EBSI's ledger**. The hash of the public key of DID control keys will always be stored on EBSI's ledger as part of EBSI's ledger's transaction. It is important to note that:

- One DID control key can only manage only one single DID.
- A DID can be managed by multiple DID control keys.
- DID control keys MUST be used ONLY for managing DIDs.

https://ec.europa.eu/ebsi

ebsi

European Blockchain