PrivacyCred

and the eHealth Network Trust Framework

Release 0.9.3

Jesus Ruiz

Apr 11, 2021

Contents

In the blockchain space, many people and many projects put the technology first and then the citizen, trying to hack privacy and data protection into a system that was not designed with those requirements from the beginning.

Surprisingly, this also applies to many implementations and initiatives in the Self-Sovereign Identity (SSI) space, where many initiatives insist on registering and recording the identities of citizens in a globally shared infrastructure, even though it is not really required or desirable for many important use cases.

Even though they claim that they use cryptographic techniques (from hashes to Zero-Knowledge Proofs) to achieve privacy, many claim so without a proper PIA (Privacy Impact Assessment) of the actual implementation in a concrete system. And most importantly, they assume that registering the identities of citizens is required, without a proper justification, violating the principle of minimisation and proportionality.

To make things worst, due to the COVID-19 pandemia there has been a proliferation of initiatives and projects advocating the use of Verifiable Credentials and blockchain technologies, but lacking the principle of "citizen first, technology last". Those initiatives tend to be the most publicised, creating a generalised distrust on blockchain for use cases where strong privacy and citizen protection is critical.

Interoperability of health certificates Trust framework

However, the recent Guidelines from the eHealth Network, especially the document on interoperability of the Trust Framework (Interoperability of health certificates Trust framework) have made clear the requirements and characteristics that such solutions must comply with, striking the right balance between citizen and public health.

Even though the documents assume centralised technology for implementation,

- they provide leeway in some areas where proper use of blockchain technology and Verifiable Credentials could add value, complementing and enhancing the solution delineated in the Guidelines while maintaining the principles of **strong privacy and data protection**, **cross-border interoperability**, **inclusiveness** and **simplicity and user-friendliness**.
- the Guidelines can be generalised to be used for the correct implementation of many different use cases in many industries (not limited to Covid-19 certificates), ensuring that the same principles mentioned above are implemented.

Applying the Guidelines to the Blockchain

The rest of the document uses the Guidelines from the eHealth Network to show how correct applications using W3C Verifiable Credentials and blockchain should be implemented.

Especifically, we describe the *PrivacyCred* system, which is is designed specifically for some important use cases where especially sensitive personal data is handled. A high degree of privacy and unlinkability is the first design criteria. It also supports scenarios where physical, on-person verification of identity of holder is needed and where normal W3C Verifiable Credential flows are not fully suitable as they are normally designed currently.

1 Introduction

We describe how PrivacyCred matches the requirements from the eHealth Network, described in the document Interoperability of health certificates Trust framework. PrivacyCred is a generic credential system which is designed to be secure, privacy-preserving, scalable, performant and robust.

It is designed specifically for some important use cases where physical, on-person verification of identity of holder is needed and where normal W3C Verifiable Credential flows are not fully suitable as they are normally designed currently.

The system is compliant with the Guidelines published by the eHealth Network, but includes some differentiating characteristicas like:

- It supports both W3C Verifiable Credentials and CBOR/COSE.
- It is based on blockchain technology.
- It is interoperable with any other implementation compatible with the eHealth interoperability guidelines, whether they are implemented with blockchain or not.

- Citizens do not have to register in any Digital Identity scheme, and no citizen information is registered in the blockchain. It uses Peer DIDs which are never stored or registered in the blockchain.

In order to better describe the characteristics of the system an how it complies with the eHealth Guidelines (and how it differs in some aspects), we use *exactly* the same text of the eHealth document with sections marked like below:

PrivacyCred

Text marked like this is additional to the original document and describes some aspects specific to this implementation

For the explanation of how PrivacyCred can complement and improve the proposed eHealth Network system, first we need to draw the proposed Trust Framework in a different but completely equivalent way, in the following figure.

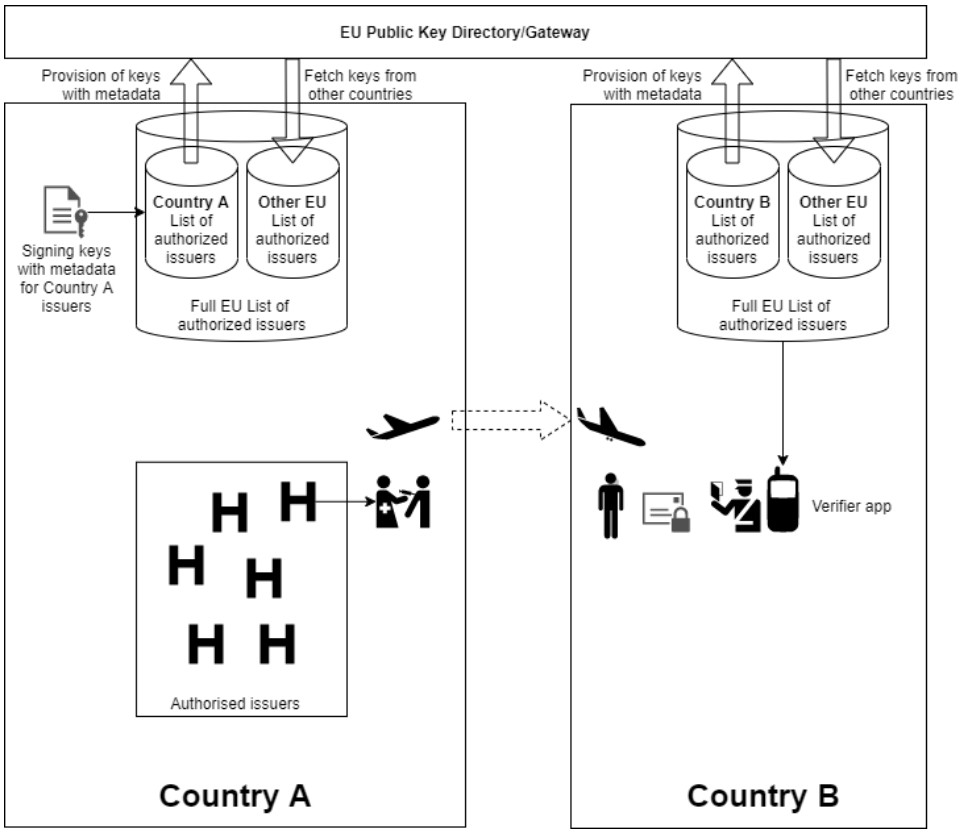

Fig. 1.1: EU PKD system

In the standard eHealth Network system, each country uploads to a central service the keys/certificates specific to that country, and downloads from that service the keys/certificates from all the other countries that use the system. In this way, the EU Public Key Directory (EU PKD) helps the different countries to maintain in each country a database with all the keys/certificates for all authorised issuers.

When one verifier entity in a country needs to verify a certificate presented by a traveler,

it can do so by checking against the local copy (meaning in the verification contry) of all keys/certificates maintained via the replication mechanism described above.

There are several ways in which a blockchain-based system like PrivacyCred could add value without modifying the essential processes or safety of the proposed system.

Option 1

Regarding the list of authorised issuers, the eHealth Network system requires that each for each country its list should be published on its PHA's website (national backend server). In addition, the list may also be published through an open API.

In Option 1, in addition to publishing the list in the website it could be published in a blockchain. In that way, the list is hyper-replicated in a secure and tamper-resistant way in all the nodes of the blockchain network.

This would facilitate verification by any entity (hotels, restaurants, etc) without overloading the website of the PHA. In other words, it implements a massively scalable and highly available read-only database for checking the keys/certificates of authorised issuers. The number of writes to the blockchain is very low (when the list of authorised issuers changes), and the reads are performed locally in each of the nodes operated by each entity participating in the blockchain.

In a sense, it would be a mechanism complementary to the open API mechanism but cheaper, more available and more scalable.

Option 2

Similar to Option 1, but publishing the full EU list in the blockchain. This could be done by a given country using the database that it has using the EU PKD, or it could even be performed by the EU entity providing the PKD service (most probably the Commission, as it happens with the European Federation Gateway Service).

Please note that in Option 1, there could be several countries that coordinate with each other and publish their lists in the same way in the blockchain, creating a single read-only list for any entity that wants to verify certificates.

Other options

In the future, there could be more "ambitious" options. For example, when EBSI (European Blockchain Services Infrastructure) is in production, it could be used as a complement or even replacing completely the EU PKD centralised system. Each country would keep their sovereignty regarding managing their authorised issuers list, but the replication of that data across the EU could be simplified enormously using the EBSI blockchain network.

In the same way, there could be different "national" or even pan-european blockchain networks that could be used by countries to "disseminate" the master lists in a safe, cheap and available way.

The eHealth Network document mentions the ICAO PKD. As the ICAO PKD site explains:

> The publication of a Master List enables other receiving States to obtain a set of CSCA certificates from a single source (the Master List issuer) rather than establish a direct bilateral exchange with each of the Issuing Authorities or organizations represented on that list. However, *the more instances of a CSCA certificate that a*

receiving State acquires — whether through multiple Master Lists, bilateral exchange, or both — *the more confident* the receiving State can be that the CSCA certificate they are using for validation is authentic. In this respect, Master Lists contribute to building and improving trust based on CSCA certificates.

The blockchain-based PKD is not intended to replace the centralized PKD (at least for the moment), but instead to complement it and provide in a secure way more places where the lists are available for verifiers. For example, the current ICAO PKD service is hosted in identical systems within two geographically separate sites (location A being located in Berlin, Germany and location B being located in Abu Dhabi, United Arab Emirates). An operator location is additionally provided within the ICAO headquarters (being located in Montreal, Canada). The two hosting sites are designed so that each of them can take over the work of the other site should one of them fail.

A blockchain-based system could provide several benefits, including:

- Greater resiliency by replicating in a simple and secure way the Master Lists and associated data.
- Better scalability, as most of the operations in the PKD system are reads (for verifications). Using a blockchain the data is hyper-replicated in a tamper-resistant way in all the nodes of the network, and the verifications can be done to servers which are very close to the geographical location of the verifier.
- An alternative method to the current download method for users of the PKD data. It is enough to operate a node in the blockchain network and the data is updated automatically when the central PKD repository is updated (assuming the update process includes updating the data in the blockchain). Nobody can tamper with the data and the history of the previous versions of the Master Lists are available if needed.

2 Annotated eHealth Network Trust Framework

2.1 Introduction

The European Council has repeatedly called for a coordinated approach[1] on interoperable vaccination certificates and the mutual recognition of test results.

The Guidelines[2] adopted by the eHealth Network[3] rest on three pillars: a minimum data set, a standard unique identifier for such proofs, and a trust framework, which provides the basis for establishing the certificates' authenticity, integrity and validity.

This document outlines the trust framework and provides the basis for discussion with Member States on the implementation of interoperable certificates in EU Member States. Further elabo-

[1] https://www.consilium.europa.eu/media/47296/1011-12-20-euco-conclusions-en.pdf

[2] eHealth Network guidelines on proof of vaccination for medical purposes - basic interoperability elements, adopted and published on 27 January 2021. Published here.

[3] The eHealth Network is a voluntary network created under article 14 of Directive 2011/24/EU on the application of patients' rights in cross-border healthcare. It provides a platform for Member States' competent authorities responsible for eHealth.

ration on the specifications of the technical implementation will follow. The document may be subject to future modification as the COVID-19 situation evolves.

Fig. 2.1: Mock-ups of a paper and digital vaccination certificate, as an example

The trust framework defines the rules, policies, protocols, formats and standards needed to ensure that Covid-19 health certificates are issued in such a way that their authenticity and integrity can be verified and trusted.

The trust framework shall be flexible enough to encompass different use cases. It defines provisions that allow both digital and analogue, off-line and on-line versions of the COVID-19 health certificates, as well as the associated verification.

2.2 Business needs and requirements

The journey of the Covid-19 health certificate is completed in 3 distinct steps:

1. the collection and registration of data about the medical events for competent authorised entities in a health information system,
2. the issuance of the Covid-19 health certificate, and
3. the presentation of the Covid-19 health certificate to a verifier (e.g. a border guard or a healthcare professional) for its verification.

A certificate relies on a minimum dataset. Included in the minimum dataset is a Unique Vaccination Certificate/assertion identifier (UVCI), which could be used as a link to the underlying data registry. The use of UVCI or other methods for online verification will be defined in more detail in the next versions of the Trust Framework.

PrivacyCred

Fig. 2.2: Main steps of the vaccination journey, as an example of the generation and use of a health certificate

The PrivacyCred implementation uses a UUID Version 4 which is truly random as specified in RFC 4122. The UUID has no relationship at all with any data inside the certificate and so is unlinkable. Each certificate has a different UUID even if they refer to the same person.

The verifier of a certificate should be able to establish that:

- The certificate has been issued by an authorised entity;
- The information presented on the certificate is authentic, valid, and has not been altered;
- The certificate can be linked to the holder of the certificate;

2.2.1 Main design principles and business requirements

The design of the trust framework for EU-interoperable issuing of COVID-19 health certificates and verification of their integrity and authenticity relies on key design principles listed below. The list is not prioritised. Instead, the trust framework that is specified later in the document attempts to optimise as many of the key design principles as possible.

Cross-border interoperability. National implementations of certificates that comply with the specifications of the trust framework should be interoperable. This means that if Countries A and B implement the specifications, it should be possible for a verifier in Country B to verify a digital vaccination certificate that has been issued in Country A.

Cross-border interoperability should be ensured across EU and EEA countries. The Trust Framework should not prevent interoperability with the solutions designed on a global level, such as the one being developed by the WHO or ICAO. This is one of the primary design principles and it has implications in all components of the proposed trust framework.

Data protection (including data minimisation, purpose limitation, etc.). The trust framework should protect the data of the involved individual stakeholders (most importantly, certificate holders). This covers several data protection dimensions catered by the General Data Protection Regulation, including purpose limitation and data minimisation. In practice, only the bare minimum set of data that is required for the supported use cases

should be processed (data minimisation) and the purpose of data collection should be checked against the use cases (purpose limitation). Similarly, only the bare minimum set of data that is required for the supported use cases should be presented to a specific verifier (data minimisation) and the purpose of data presentation should be checked against the use cases (purpose limitation). In order to achieve the latter, the trust framework may support different presentation datasets for different verifier scenarios. The data protection principle has a strong impact on the specification of the Minimum Dataset and the design of the use cases of the trust framework.

Data security and privacy by design and by default. Abuse of data by actors (especially, the certificate verifiers and holders) and forgery should be prevented by any reasonable means. The trust framework should by design and default ensure the security and the privacy of data in the compliant implementations of digital vaccination certificate systems, ensuring both security and privacy. Available tools should be used for restricting access to data and preventing malicious use of data, while the establishing of the authenticity of data and its link to the certificate holder should be ensured. The design should prevent the collection of identifiers or other similar data which might be crossreferenced with other data and re-used for tracking ('Unlinkability'). Further discussions are needed as to the technological aspects and timeline for the incorporation of these features in the trust framework.

Inclusiveness (especially medium-neutrality). The trust framework should be inclusive both towards Member States' approaches and the individual citizen ('no citizen left behind'). The design of the trust framework should attempt to maximize its support for diverse contexts (e.g., high resource vs low resource contexts). To enable this, the trust framework should support a spectrum of certificate presentation media from plain paper certificate to augmented paper certificates (e.g., paper certificate with printed machinereadable parts such as barcodes, QR codes, Machine Readable Zones) and to purely digital certificates (e.g., in-app certificates).

PrivacyCred

The PrivacyCred system supports paper certificates, augmented paper certificates, QR codes and purely digital certificates.

In addition, contrary to many SSI Verifiable Credentials implementations, the PrivacyCred system is implemented as a PWA (Progressive Web App) that can be used simply in a mobile browser (or tablet/PC) without installing anything in the device or registering for anything.

However, the user can install the PWA in the device if the user so wishes to facilitate future uses. In any case, the system does not require any type of registration of the identity of the user.

Simplicity and user-friendliness. It is very important that the trust framework is designed with simplicity and user-friendliness of the possible implementation of digital certificate systems in mind. More formally, the trust framework should not have features or functionalities that would unnecessarily complicate the resulting implementation of a digital vaccination certificate system or make them unnecessarily difficult to use. Lack of simplicity could increase the time it takes to implement the compliant digital vacci-

nation certificate systems, while lack of user friendliness could hinder the uptake of the resulting implementations. User-friendliness is relevant for quick and easy processing, specifically to certificate holders and to verifiers.

PrivacyCred

The PrivacyCred system follows the rule of *Occam's Razor* eliminating any feature or functionality which is not strictly required for the use case.

This not only provides simplicity and user-friendlyness but also provides a system easier to understand and maintain which is more secure and robust.

Implementation flexibility. The trust framework specifications should provide implementers with a variety of options when developing digital vaccination certificate systems according to the trust framework specifications. This key design principle aims at reducing the implementation time and leveraging/reusing existing infrastructures in Member States. To satisfy this principle, the trust framework specifies, whenever possible, a list of alternative methods, flows, architectures and implementation options, for example alternative presentation media, verification options, implementation technologies, etc. whilst still guaranteeing the same level of trustworthiness.

Modularity and scalability. This is strongly linked with the previous key design principle. The trust framework architecture should be modular and easily scalable, for instance, to additional usage scenarios, use cases and types of certificates. The trust framework already supports different usage scenarios (e.g. alternative settings in which certificates may be requested or verification may take place). Examples of other types of certificates that could be supported by potential extensions of the trust framework include certificate of negative COVID-19 tests and certificates of recovery from COVID-19, while examples of other use cases that could be supported are travel or (participation in) leisure activities (i.e. proof of vaccination for non-health-related purposes in domestic or international settings). Decisions related to ethical, societal or political questions pertaining to the use cases should be tackled separately. To satisfy this key design principle, special attention has been paid in the design of the trust framework architecture with clear separation of the steps of the user story detailed below.

PrivacyCred

The PrivacyCred system was designed from the ground up to fully support use cases like certificates of negative COVID-19 tests, certificates of recovery from COVID-19, vacccination certificates, and their potential use in travel and leisure activities.

It has been designed explicitly to maintain total unlinkability among certificates and each of them are treated as if they were a unique "piece of paper" but with the power of the digitalization. That means that the system is not suitable for use cases where linkability is required. In any case, that linkability should be implemented in the existing information systems of the health entities issuing the certificates.

Open standards. The trust framework should rely for its implementations on open standards,

to the extent that this is possible. This will greatly contribute to the interoperability of the resulting implementations, in addition combined with open governance and open source implementations, it will instil trust in the involved stakeholders.

2.2.2 User roles

The user roles that are associated with the supported user stored of the trust framework are presented in the table below.

Table 2.1: Roles in the system

ROLE	DESCRIPTION	EXAMPLES
Certificate Issuer	The trusted entity that issues and signs a statement/credential/certificate.	For paper certificates, a healthcare organisation or healthcare authority. For digital certificates, an electronic medical record system, an IIS, a HP portal, a patient portal, a system used by another relevant authority.
Certificate Holder	The person in possession of a certificate.	A person, their guardian, legal representative or another authorized person.
Certificate Reader	The actor (a person or a computer system) analysing the contents of a certificate presented by a certificate holder.	A healthcare professional or another person or a system entitled to the detailed information on the certificate (e.g., a healthcare appointment system).
Certificate Verifier	The actor (a person or a computer system) checking the validity of a certificate presented by a certificate holder.	An authority, an online system used by the certificate holder (for example, an online check-in)

2.2.3 ID binding and verification

An important parameter of the trust framework pertains to the identity of the subject of the certificate i.e., the person for whom the certificate is issued. The identity of this subject shall be bound to a certificate when the latter is issued (ID binding) and has to be verified when the certificate is being presented and verified (ID verification). These two processes (ID binding at the Issuance step and ID verification at the Presentation and Verification step) prevent possible impersonation attempts (i.e., a person fraudulently presenting a certificate that has been issued to someone else as if it were their own), and are in line with the data security and privacy by design and default principles of the trust framework.

The processes of ID binding and/or verification may be optional for some usage scenarios in the scope of the trust framework. For instance, in some settings the simple presentation of the certificate to a healthcare professional for medical purposes might be enough without additional actions for proving the ownership of the certificate if complemented by good clinical practices. However, in those usage scenarios where the aforementioned process cannot be omitted, the trust framework, adhering to the simplicity and user-friendliness principle, shall rely on (nationally and/or internationally) established methods for ID binding and verification.

In other words, the trust framework does not specify in its architecture dedicated components or modalities for undertaking the ID binding and verification process.

The recommended methods for performing ID binding and verification employ nationally issued identity proof documents, such as national IDs and passports. Such identity proof documents should be presented at the time of issuance (ID binding) and verification (ID verification) of the certificate and therein personally identifying information should be compared against the information in the certificate.

PrivacyCred

The PrivacyCred system assumes nationally issued identity proof documents for ID binding, both at issuance and verification.

Contrary to many SSI Verifiable Credentials implementations, the PrivacyCred system does not require any registration on the part of the user (like registering her DID in the blockchain or any other repository).

The only personal information managed by the system is the one in the minimum dataset as specified in the document Guidelines on verifiable vaccination certificates - basic interoperability elements from the eHealth Network. The personal data elements are incorporated to the certificate and not used for any other thing or purpose. It is assumed that the minimal person identification data specified in the eHealth Network document can be used to perform the ID binding with a national ID, passport or any other suitable nationally issued identity document.

2.3 Trust architecture

This chapter provides an overview of the trust architecture and describes its main components. The chapter contains requirements directed at the Member States acting in the roles of issuers and verifiers.

The WHO is developing a global trust framework based on a similar approach. The framework is centred around the Global Health Trust Anchor operated and governed by the WHO and based on the technical specifications derived from ICAO's Public Key Directory (PKD) model.

2.3.1 Overall description

The EU trust framework is designed to be largely decentralised.

As per the digital contact tracing apps and the European Federation Gateway Service, this reflects the divergent structures and approaches within the EU Member States. That is to say, it aims to avoid centralisation where possible in line with the principle of flexibility.

However, there are some centralised elements:

1. Roots of trust stored in a common directory/gateway (EU Public Key Directory/Gateway), similar to the public key certificate provision process established in the EFGS.

2. Governance model.

The main elements of the system are outlined in Figure 3 and described further below.

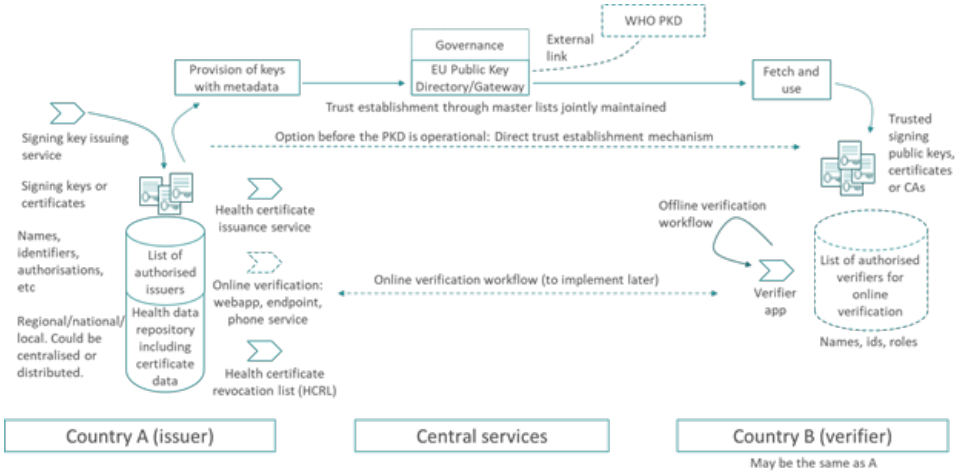

Fig. 2.3: Overall architecture of the system (solid lines = first version of the trust framework specifications; dashed lines = future versions of the trust framework specifications)

Country A (country of issuance)

The country of issuance, through its competent health entities, is responsible for the recording of health data and issuing certificates. It is also possible for the issuer to deliver a certificate based on reliable information received from other sources.

The Country A that is participating to the interoperability scheme shall issue certificates at least in the form of the augmented paper (paper augmented with digital artefacts such as barcodes or QR codes). In addition, Country A may issue certificates stored as purely digital files in apps or computers.

The Country A shall assign a national Public Health Authority (PHA) responsible for the system. The name of the PHA shall be communicated to other members of the interoperability scheme through the eHealth Network secretariat.

SafeIsland

The SafeIsland system assumes exactly the same process as explained above.

List of authorised issuers

A system used by Country A for maintaining details about healthcare organisations authorised to issue health certificates.

The list should be established and maintained by every Country A, and it should be published on its PHA's website (national backend server). In addition, the list may also be published through an open API.

SafeIsland

The SafeIsland system provides a blockchain-based list of authorised issuers. It can be complementary to a national backend server for publishing the list, or even a complete replacement. It is enough that the PHA operates a node in the blockchain network and publishes the data in that node. Automatically, the data is securely hyper-replicated to all other connected nodes where many different types of verifiers can securely and efficiently query the data for verification. This could be very useful for example for certificates used in travel and leisure where the number of verifiers can be much higher than just the airports.

The list should contain the data set following the description given in Table 1.

Table 2.2: List of certified issuers, data set for each entry

Group and cardinality	Element	Description	Data type and cardinality
Issuer identification 1..1	Country	Country of the issuer	Coding (ISO 3166-1 alpha-2) 1..1
Issuer identification 1..1	Name	Name of the issuer	String 1..1
Issuer identification 1..1	Identifier	Identifier of the issuer	Identifier (format to be defined later) 1..1
Issuer identification 1..1	Public key or PKI certificate	Public key or PKI certificate assigned to the issuer	Text (format to be defined later) 1..*
Issuer identification 1..1	Online verification webapp	Address of the online verification webapp, if offered by the issuer	URI 0..1
Issuer identification 1..1	Online verification endpoint	Address of the online verification service, if offered by the issuer	URI 0..1
Issuer authorization 1..*	Health certificate type	The type of health certificates the issuer is capable and authorised to issue	Coding (value set: vaccination certificate) 1..1
Issuer authorization 1..*	Validity from	Start of the authorisation period	dateTime 1..1
Issuer authorization	Validity to	End of the authorisation period	dateTime 1..1

Health data repository

A repository used by Country A for storing health information and information about the issued health certificates.

The system may be part of an Immunization Information System (IIS), a laboratory system or it may be stored by national, regional or local electronic health record systems, or on paper. The system may be centralised on the national level or it may be largely distributed.

Every Country A may use their own arrangements for establishing and maintaining the health data repository. An overall description of the arrangements shall be made publicly available by each Country A.

Signing key issuance service

A service such as a Certificate Authority (CA) or another arrangement used by Country A for issuing signing key pairs or certificates, to be used for signing health certificates.

The term "signing key" in this document refers to keys or certificates issued to legal and natural persons and used for creating electronic seals and signatures. No difference is made between electronic signatures and electronic seals in this document, and the terms "signature" and "signing" are used to refer to both of them.

Country A may use any public or private CA (or another option) in order to issue signing keys or certificates used for signing health certificates.

Signing keys or certificates

Digital signature keys or certificates used by Country A for signing health certificates.

Signing keys or certificates shall only be provided to entities with active authorisation according to the published List of authorised issuers.

Member States should have a clear policy for revocation of health certificates, including refresh rates for verifiers.

Provision of keys with metadata

A process executed by Country A in order to register the signing keys or certificates to the EU Public Key Directory/Gateway (see 3.1.3.1 below).

The process and related procedures for the secure registration of public keys or certificates will be defined by the eHealth Network.

Health certificate issuance service

A service used by Country A for issuing health certificates and delivering them to certificate holders.

The service may be implemented as a patient-facing app, as a patient portal, as a healthcare professional portal, or it may be integrated to another national, regional or local system.

Every Country A shall implement at least one health certificate issuance service. Health certificates shall only be issued by entities with active authorisation according to the published List of authorised issuers.

Health certificate revocation list (HCRL)

A system used by Country A for publishing information about revoked health certificates.

Each Country A shall publish one and only one aggregate list of all revoked health certificates. Country A is responsible for putting its revoked certificates on the list and signing it using one of its signing keys controlled by the PHA.

Online verification (webapp, browser-based) – for future consideration

An online system (website/webapp, to be accessed using a browser) that may be used by verifiers for ascertaining the validity of health certificates presented by their holders.

Country A shall not make the use of the online verification webapp mandatory for the verification of health certificates.

Every Country A may make an online verification webapp available. A Country providing such a webapp should make exactly one online verification webapp available.

More detailed specifications are to be provided in the next revisions of this Trust Framework.

Online verification (endpoint, API) – for future consideration

An online system (such as a RESTful API) that may be used by verifiers for ascertaining the validity of health certificates presented by their holders.

Country A shall not make the use of the online verification endpoint mandatory for the verification of health certificates.

Every Country A may make an online verification endpoint available. Countries A providing an endpoint should make exactly one online verification endpoint available.

More detailed specifications are to be provided in the next revisions of this Trust Framework.

Online verification (phone service)

A phone service that may be established by Country A for enabling verifiers to check the validity of health certificates presented by their holders.

The service may be implemented through the national contact points for cross-border healthcare. The answer to a verification request should be provided within 2 working days.

Country B (country of verification)

The country of verification is responsible for verifying health certificates presented by their holders. The Country B shall accept valid health certificates that are issued following this Trust Framework.

List of certified verifiers

Specifications are to be provided in the next revisions of this Trust Framework.

Fetch and use

This is a process executed by Country B in order to retrieve information from the EU Public Key Directory/Gateway.

Trusted signing public keys, certificates or CAs

Signing keys are fetched by Country B from the EU Public Key Directory/Gateway and trusted by Country B.

All public keys and certificates marked as valid in the EU Public Key Directory/Gateway by Country A shall be trusted by Country B. If Country A has uploaded a public key certificate of a Certificate Authority (CA), all certificates issued by this CA shall be trusted by Country B.

Vaccination certificate verifier app

These are application(s) that are used by verifiers for ascertaining the validity of certificates presented by their holders.

In this version of the Trust Framework, only offline verification is supported. All verifier apps shall support offline verification.

Central services

The central services provide a process and a gateway for sharing trust anchors (public keys or certificates) between Countries A and B.

Before the gateway is implemented, Country B may request trust anchors directly from Country A through a mechanism ensuring the authenticity and integrity of this data, for example through the use of secure email or by downloading the information from the PHA's website of Country A.

After the implementation of the central services, the use of the direct trust establishment mechanism shall be discontinued.

EU Public Key Directory/Gateway

A directory that contains information about public keys or certificates published by Country A, as well as their metadata, and acts as a gateway used for providing trust information to national systems.

The directory shall be provided by a public sector body, such as the European Commission.

The directory shall be derived from the Lists of authorised issuers published by all Countries A. The contents shall be made publicly available. The list shall not contain personal information such as names of health professionals.

A flat structure could be foreseen for the PKD; further considerations are ongoing.

Country B shall ensure that the contents downloaded from the EU Public Key Directory (EU PKD) are regularly distributed to the verifier apps.

2.3.2 Legal basis

The trust framework described in this document is also subject to legal considerations.

As *some of the processing operations described involve personal data* (e.g. issuance and verification of certificates) such processing will fall under the scope of the General Data Protection Regulation (GDPR)footnote:[OJ L 119, 4.5.2016, p.1].

GDPR provides for obligations on controllers (entities determining the purposes and means of processing of personal data, here e.g. organisations issuing and verifying vaccination certificates), such as to have a *legal basis* for their processing operations, **document them**, implement **appropriate security measures**, and to *inform data subjects* (natural persons data relating to whom are processed). It also provides rights for data subjects, such as the right to access the data controllers hold about them and to have it corrected. Additionally, GDPR establishes rules for *transfers* of personal data outside the EU/EEA.

2.4 Data formats

.A proposal for data encoding and representation image::ehealth_figure4.jpg[]

2.4.1 UTF-8

UTF-8 will be used for character encoding.

2.4.2 FHIR

The Fast Healthcare Interoperability Resources (FHIR) standard data format is recommended to be used for expressing relevant health data. The data will be converted though appropriate mapping definitions to form the machine readable part of the certificate dataset using the JavaScript Object Notation (JSON) for data representation.

2.4.3 CBOR/COSE

The Concise Binary Object Representation (CBOR; RFC 8949) will be used for serializing the JSON data representation as binary data. The CBOR Object Signing and Encryption (COSE; RFC 8152) specification will be then used for digitally signing the machine readable certificate data.

2.5 Presentation formats

2.5.1 2D Barcode

Only 2D barcodes whose symbology is specified as an ISO standard SHALL be used. ISO standardized 2D barcodes symbologies include DataMatrix [ISO/IEC 16022], Aztec Codes [ISO/IEC 24778], and QR Codes [ISO/IEC 18004]. However, it is RECOMMENDED that the barcode is encoded as an Aztec code. Verifiers shall support all specified types of 2D barcodes.

2.5.2 W3C Verifiable Credentials

Decision about W3C Verifiable credentials to be made later.

2.6 Cryptography

2.6.1 Data signing

The CBOR Object Signing and Encryption (COSE) specifications will be used for digitally signing the machine readable certificate data.

To meet the timeline of this effort, and to ensure reliable and secure implementations of the technical specifications, the primary signing scheme for digital signatures supported by the

trust framework is EC-DSA (Elliptic-Curve Digital Signature Algorithm) for cross-border use where unlinkability does not apply. As a fallback, RSA is also supportedfootnote:[The RSA signing scheme should only be used if it is absolutely necessary, as it adds an around 50% size overhead to the resulting health certificate.].

To further address the development of a privacy preserving approach for the anticipated domestic use case, adding further cryptographic schemes such as CL or BBS+ will be supported outside of cross border scenarios.

2.6.2 Data encryption

Data encryption of the machine readable part of the certificates will not be used. Selected disclosure of information can be implemented using other mechanisms. Adding data encryption of individual fields would increase complexity associated with key management.

2.7 Verification protocols

2.7.1 Offline

Offline verification shall be supported. By the term offline we refer to the scenario where the verifier requires at the time of the verification needs no online access to external resources (such as a call centre or a webapp) to perform the verification. Instead, the digital signature included in the 2D barcode will be verified through dedicated verification software. Signature verification will include (1) the verification of its validity against the provided public key and (2) the check that the public key is on the list of trusted keys held by the verifier app. The list will be fetched periodically from the EU PKD, however in the first phases of deployment, direct exchanges of keys may be used, as described in Section 3.1.3. Once this digital signature has been verified, the verification software can decode the information in the 2D barcode and rely on its content.

2.7.2 Online

Online verification will rely on the UVCI and it will be incorporated in the next version of the specifications (V2).

3 PrivacyCred system

3.1 Requirements

The journey of a certificate is completed in 3 distinct steps:

1. the collection and registration of data about the citizen and her associated characteristics,
2. the issuance of certificate, and
3. the presentation of the certificate to a verifier for its verification.

A certificate should rely on a minimum dataset. Included in the minimum dataset is a Unique Certificate Identifier (UCI), which could be used as a link to the underlying data registry.

The verifier of a certificate should be able to establish that:

- The certificate has been issued by an authorised entity;
- The information presented on the certificate is authentic, valid, and has not been altered;
- The certificate can be linked to the holder of the certificate;

3.1.1 Main design principles and business requirements

The design of the trust framework for interoperable issuing of certificates and verification of their integrity and authenticity relies on key design principles listed below. The list is not prioritised. Instead, the trust framework that is specified later in the document attempts to optimise as many of the key design principles as possible.

Data protection (including data minimisation, purpose limitation, etc.) The trust framework should protect the data of the involved individual stakeholders (most importantly, certificate holders). This covers several data protection dimensions catered by the General Data Protection Regulation, including purpose limitation and data minimisation. In practice, only the bare minimum set of data that is required for the supported use cases should be processed (data minimisation) and the purpose of data collection should be checked against the use cases (purpose limitation).

Similarly, only the bare minimum set of data that is required for the supported use cases should be presented to a specific verifier (data minimisation) and the purpose of data presentation should be checked against the use cases (purpose limitation).

In order to achieve the latter, the trust framework may support different presentation datasets for different verifier scenarios. The data protection principle has a strong impact on the specification of the Minimum Dataset and the design of the use cases of the trust framework.

Data security and privacy by design and by default Abuse of data by actors (especially, the certificate verifiers and holders) and forgery should be prevented by any reasonable means. The trust framework should by design and default ensure the security and the privacy of data in the compliant implementations, ensuring both security and privacy.

Available tools should be used for restricting access to data and preventing malicious use of data, while the establishing of the authenticity of data and its link to the certificate holder should be ensured. The design should prevent the collection of identifiers or other similar data which might be crossreferenced with other data and re-used for tracking ('Unlinkability').

Inclusiveness (especially medium-neutrality) The trust framework should be inclusive especially towards the individual citizen ('no citizen left behind').

The design of the trust framework should attempt to maximize its support for diverse contexts (e.g., high resource vs low resource contexts).

To enable this, the trust framework should support a spectrum of certificate presentation media from plain paper certificate to augmented paper certificates (e.g., paper certificate with printed machinereadable parts such as barcodes, QR codes, Machine Readable Zones) and to purely digital certificates (e.g., in-app certificates).

The PrivacyCred system supports paper certificates, augmented paper certificates, QR codes and purely digital certificates.

In addition, contrary to many SSI Verifiable Credentials implementations, the PrivacyCred system is implemented as a PWA (Progressive Web App) that can be used simply in a mobile browser (or tablet/PC) without installing anything in the device or registering for anything.

However, the user can install the PWA in the device if the user so wishes to facilitate future uses. In any case, the system does not require any type of registration of the identity of the user.

Simplicity and user-friendliness It is very important that the trust framework is designed with simplicity and user-friendliness of the possible implementation of digital certificate systems in mind.

More formally, the trust framework should not have features or functionalities that would unnecessarily complicate the resulting implementation of a digital certificate system or make them unnecessarily difficult to use. Lack of simplicity could increase the time it takes to implement the compliant digital certificate systems, while lack of user friendliness could hinder the uptake of the resulting implementations. User-friendliness is relevant for quick and easy processing, specifically to certificate holders and to verifiers.

The PrivacyCred system follows the rule of **Occam's Razor** eliminating any feature or functionality which is not strictly required for the use case.

This not only provides simplicity and user-friendlyness but also provides a system easier to understand and maintain which is more secure and robust.

Implementation flexibility The trust framework specifications should provide implementers with a variety of options when developing digital certificate systems according to the trust framework specifications. This key design principle aims at reducing the implementation time and leveraging/reusing existing infrastructures in involved entities.

To satisfy this principle, the trust framework specifies, whenever possible, a list of alternative methods, flows, architectures and implementation options, for example alternative presentation media, verification options, implementation technologies, etc. whilst still guaranteeing the same level of trustworthiness.

Modularity and scalability This is strongly linked with the previous key design principle. The trust framework architecture should be modular and easily scalable, for instance, to additional usage scenarios, use cases and types of certificates.

The trust framework already supports different usage scenarios (e.g. alternative settings in which certificates may be requested or verification may take place).

Open standards The trust framework should rely for its implementations on open standards, to the extent that this is possible. This will greatly contribute to the interoperability of the

resulting implementations, in addition combined with open governance and open source implementations, it will instil trust in the involved stakeholders.

Cross-border interoperability Implementations of certificates that comply with the specifications of the trust framework should be interoperable, and not only at the national level.

This means that if Countries A and B implement the specifications, it should be possible for a verifier in Country B to verify a digital vaccination certificate that has been issued in Country A.

Cross-border interoperability should be ensured across EU and EEA countries. The Trust Framework should not prevent interoperability with the solutions designed on a global level.

3.1.2 ID binding and verification

An important parameter of the trust framework pertains to the identity of the subject of the certificate i.e., the person for whom the certificate is issued. The identity of this subject shall be bound to a certificate when the latter is issued (ID binding) and has to be verified when the certificate is being presented and verified (ID verification). These two processes (ID binding at the Issuance step and ID verification at the Presentation and Verification step) prevent possible impersonation attempts (i.e., a person fraudulently presenting a certificate that has been issued to someone else as if it were their own), and are in line with the data security and privacy by design and default principles of the trust framework.

The processes of ID binding and/or verification shall rely on (nationally and/or internationally) established methods for ID binding and verification. In other words, the trust framework does not specify in its architecture dedicated components or modalities for undertaking the ID binding and verification process.

The recommended methods for performing ID binding and verification are based on nationally issued identity proof documents, such as national IDs and passports, and regulated customer onboarding processes (in the case of private companies). The binding is performed at the time of issuance (ID binding) and verification (ID verification) of the certificate and therein personally identifying information held in the syetems of the entities involved should be compared against the information in the certificate.

Contrary to many SSI Verifiable Credentials implementations, the PrivacyCred system does not require any registration on the part of the user like registering her DID in the blockchain or any other repository, as the system relies in pre-existing identification processes (e.g., KYC for private companies).

The only personal information managed by the system is the one in the minimum dataset as specified in this document. The personal data elements are incorporated to the certificate and not used for any other thing or purpose. It is assumed that the minimal person identification data specified in this document can be used to perform the ID binding with a national ID, passport or any other suitable nationally issued identity document.

3.2 PrivacyCred: General description of the system

3.2.1 Main components

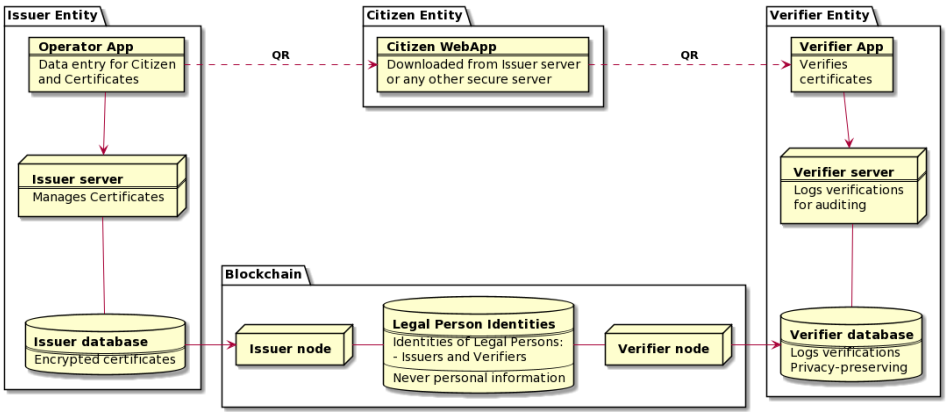

Fig. 3.1: Components of the system.

The main components are the following:

Issuer Entity The juridical person that digitally signs and issues a credential to the User. The Issuer Entity employs or subcontracts the actual people performing the process called Issuer Operator in the diagram. The Issuer Entity assumes full responsibility for the legal implications of the issueance process, especially GDPR compliance. The Issuer Entity acts as a Data Controller with respect to the Personal Information collected from the Citizen when the certification is issued.

Issuer Operator The natural person that is employed/subcontracted by the Issuer Entity to actually drive the process of issuing the credential on behalf of the Issuer Entity.

Issuer Operator App This is the application used by the natural person that drives the issuance of the credential. The application allows the operator to enter the details of the user and of the credential and issues the credential to the user on behalf of the Issuer Entity. It is the responsibility of the Issuer Entity to ensure that the Operator performs the process in the right way.

Citizen This is the natural person that receives a credential and may present it when needed.

Citizen WebApp This is the application used by the end user to manage the credentials. The reference implementation is not a native application but rather a PWA (Progressive Web App), which can be used either as a normal web app (without installation) or it can be installed and used in a very similar way to a native mobile app. The characteristics of this app are explained later.

Verifier Entity A juridical person that verifies the credential. In the process of verification, the Verifier Entity receives personal data from the Citizen. The Verifier Entity is responsible for compliance of all applicable regulations, including GDPR.

Verifier Operator A natural person that verifies the credential. It is important to distinguish between natural and juridical persons in the verification process because the flows may be different as the regulatory implications may be different. The diagram does not explicitly mention the Verifier Person, but it will be described in detail later in the document.

Verifier App The application used to verify the credential presented by the user. The reference application can be used either by an employee of a Verifier Entity or by an individual natural person, as explained later.

Blockchain This should be a Public-Permissioned blockchain network as a general-purpose blockchain network which is used to implement the Trust Framework allowing the efficient and secure verification of credentials. It is never used to store personal information. Personal information management is the responsibility of the legal entities Issuer Entity and Verifier Entity, and they are responsible for compliance to applicable regulations, especially GDPR. There may be more than one blockchain network, and the system is very interoperable across networks. The specific interoperability features are describer in a specific section later in this document.

3.2.2 Main credential flow

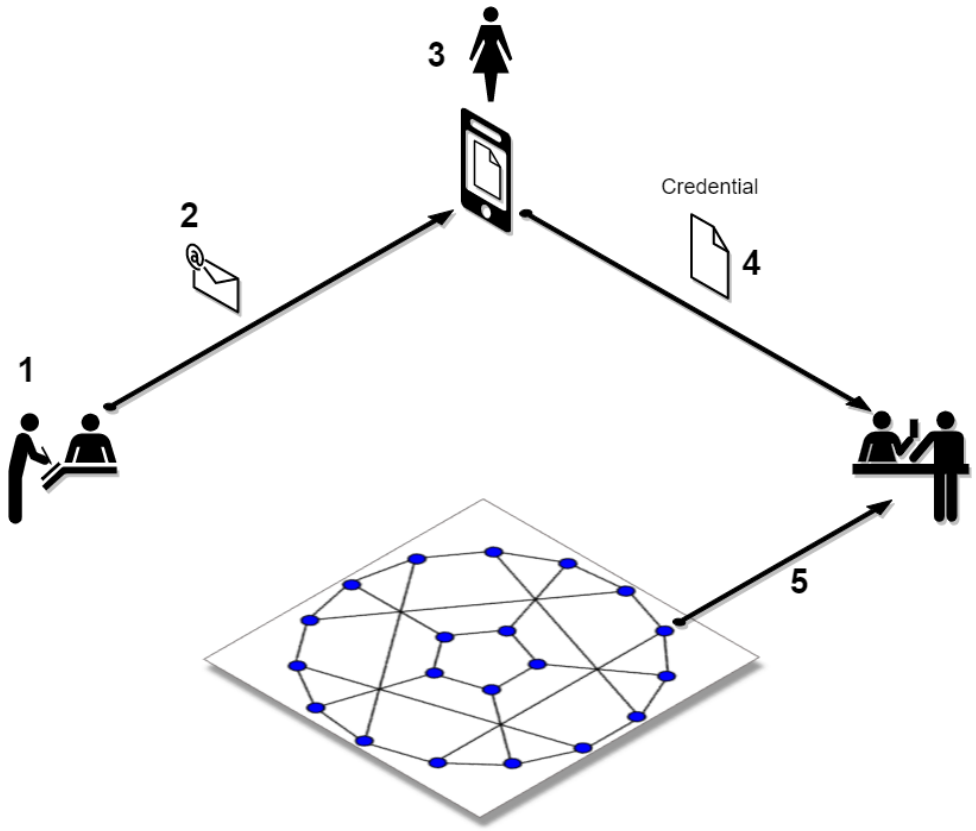

Fig. 3.2: Main credential flow.

1) **Verification of User and Credential issuance**

The Issuer Operator identifies the User (in the same way as an airline employee identifies passengers before boarding a plane) and uses her mobile app to enter the details of the User. In the initial implementation of the system the operator has also to enter manually the details of the Credential to be issued. It is the responsibility of the Issuer Operator (and ultimately of the Issuer Entity) to ensure the veracity of both the User details and the Credential details. This is a critical point in the system, as the level of trust in the credentials will depend on the level of trust of the issuance process.

2) **Sending the Credential to the Citizen**

The Credential is sent to the User. There are several possible flows, using different channels (email, QR, etc.). The main one is using QR codes and is the following:

1. The Issuer Operator displays the credential for the User in her mobile phone screen, in a QR format. More detaisl about the specific QR format later.
2. The User scans the QR using her mobile web app.
3. The mobile web app of the User gets the Credential and stores it in the storage of the mobile device.

3) **Store the Credential**

The Credential is stored in the mobile phone of the User. In the reference implementation it is stored in the IndexedDB local database. More than one credential can be stored in the mobile. A Citizen could for example store credentials of other persons of the family when traveling, or a history track of credentials received during a vacation. More details are given later in this document.

4) **Present the Credential**

When the Citizen has to prove something, she sends the Credential to the Verifier. As before, there are several possible flows, the main one using QR codes:

1. The User display the Credential in her mobile phone in QR format.
2. The Verifier scans the QR from the User mobile screen
3. The mobile app from the Verifier receives the Credential and verifies it.

5) **Verify the Credential using the Trust Framework in the blockchain**

The Verifier mobile app verifies formally the Credential with the signature, and then checks that the signature of the Credential corresponds to an authorized Issuer Entity registered in the Trust Framework in the blockchain. The verification process is essentially the one described in the W3C VC specifications.

3.3 The Trust Framework: bootstrapping the system

Before the issuance of credentials can take place, the system has to be bootstrapped and setup. There are two processes that have to be performed:

1. A One-time process at the beginning of the whole system: involves things like deploying Smart Contracts and initializing them with the parameters of the system.
2. A process for the onboarding of each new Issuer Entity and Verifier Entity. This process is basically generating and registering in the blockchain the Identity of the entity entering the system.

3.3.1 Public-Permissioned blockchain network

The system requires at least one Public-Permissioned blockchain network. The network should be trusted, efficient, publicly available and compliant with all applicable regulations.

The system is designed to be easily interoperable with other Public-Permissioned blockchain networks, like LACChain or EBSI. This is described in detail in the appropriate section of this document.

3.3.2 Information in the blockchain and Personal Identifiable Information (PII)

No personal information is ever recorded on the blockchain. The blockchain is only used to register the identities of the legal persons involved in the system. The information recorded for businesses and organizations includes:

- Public identification information of the legal person in the current regulatory environment, like VAT number, LEI (Legal Entity Identifier), or any legally accepted identification in the countries implementing the the system.
- Some commercial information, like the web site
- The public key used to verify the Verifiable Credentials digitally signed by the legal entity

The diagram below shows the registration of a new Issuer Entity in the blockchain. There are two types of legal persons registered in the blockchain:

1. **Issuer Entity**: a legal person has to be properly registered before it can issue any credential that can be verified by other actors in the system.
2. **Verifier Entity**: a legal person that receives and verifies credentials from natural persons has to be registered in the blockchain. When the legal person receives the credential (which includes personal data), this fact is registered in order to enhance auditability of the system later. This registration is performed in a privacy-preserving and scalable way. The process is described in detail later in this document. Natural persons can also verify credentials, but the verification process is different in order to avoid pre-registration of natural persons. This is described in detail later.

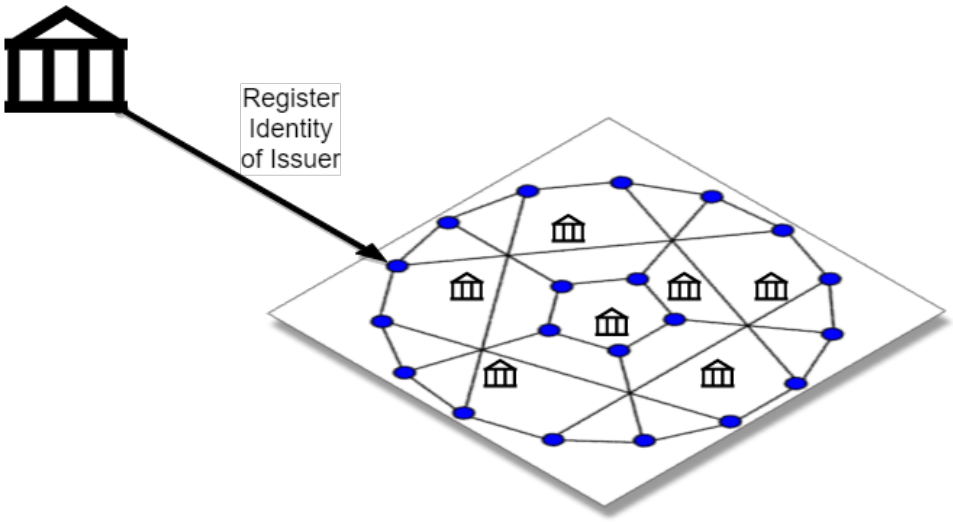

Fig. 3.3: Trusted Registry of Legal Entities in the blockchain.

3.3.3 Trust Framework: trusted registration process of legal entities

The trust framework is designed to be largely decentralised.

The identities of the legal persons involved in the ecosystem are registered in a common directory implemented in the blockchain following a hierarchical scheme very similar to the DNS (Domain Name Service) schema in the Internet. Once an entity is registered in the system, it is completely autonomous for adding other entities that are managed as child entities.

However, there is one centralised element: the root of trust at the top of the hierarchy should be a trusted entity in the ecosystem that is the one bootstraping the system. Typically it should be a regulatory body or a public administration.

The approach is described in the following figure.

Creating identities

A new identity can only be registered as a sub-node by an existing entity already registered in the system. The API used is `/api/did/v1/identifiers` and its definition is the following:

`POST /api/did/v1/identifiers`

Create an Identity anchored in the blockchain.

Request JSON Object

- **`DID`** (*`string`*) – the DID of the new identity, example: "did:elsi:VATES-B60645900"
- **`domain_name`** (*`string`*) – Domain name to assign in the hierarchy, example: "in2.ala"
- **`website`** (*`string`*) – Website of the entity, example: "www.in2.es"

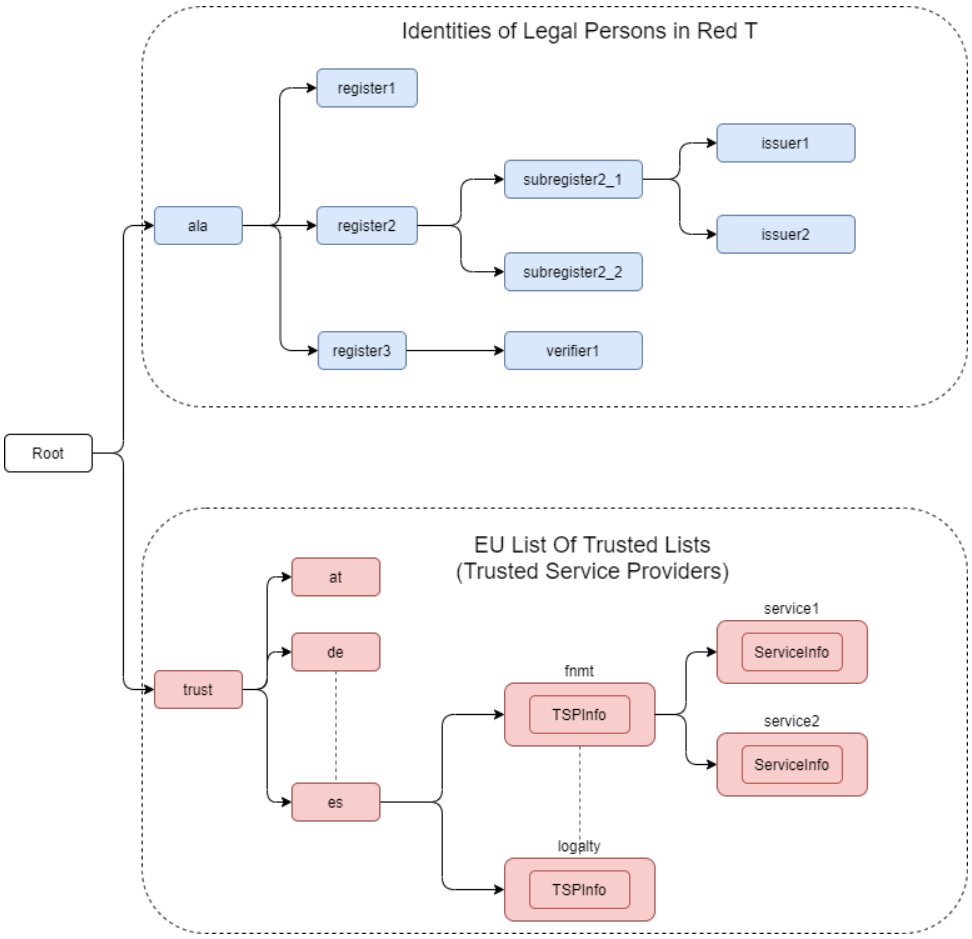

Fig. 3.4: The Trust Framework in the blockchain.

- **commercial_name** (*string*) – Commercial name, example: "IN2 Innovating 2gether"
- **new_privatekey** (*PrivatekeyJWK*) – The private key of the new entity
- **parent_privatekey** (*PrivatekeyJWK*) – The Private Key of caller (in this case the owner of "ala")

An example of the data in the request body:

```
{
    "DID": "did:elsi:VATES-B60645900",
    "domain_name": "in2.ala",
    "website": "www.in2.es",
    "commercial_name": "IN2 Innovating 2gether",
    "new_privatekey": {
        "kty": "EC",
        "crv": "secp256k1",
        "d": "Dqv3jmu8VNMKXWrHkppr5473sLMzWBczRhzdSdpxDfI",
        "x": "FTiW0a4r7S2SwjL7AlFlN1yJNWF--4_x3XTTxkFbJ9o",
        "y": "MmpxbQCOZ0L9U6rLLkD_U8LRGwYEHcoN-DPnEdlpt6A"
    },
    "parent_privatekey": {
        "kty": "EC",
        "crv": "secp256k1",
        "d": "Dqv3jmu8VNMKXWrHkppr5473sLMzWBczRhzdSdpxDfI",
        "x": "NKW_0Fs4iumEegzKoOH0Trwtje1sXsG9Z1949sA8Omo",
        "y": "g4B3EI0qIdlcXTn-2RpUxgVX-sxNFdqCQDD0aHztVkk"
    }
}
```

Response JSON Object

- **didDocument** (*DIDDocument*) – The DID document associated to the input DID

A more detailed explanation of each field follows:

DID is the DID of the new entity. We support ELSI DID method (ELSI_DID_Method) and AlastriaID. The DID has to be created before the call to the API with the appropriate method for the DID. In the case of ELSI this is trivial and described in the section mentioned above.

domain_name the domain name for the new entity in the Trust Framework. In the example it is *in2.ala* because it will be a sub-node of the Alastria one. The new identity will be created as a child node of the existing node owned by the entity controlling the `parent_privatekey`. If the parent domain name specified here is not owned by the entity controlling the `parent_privatekey`, an error is returned and no action is taken.

website the website address in the off-chain world, so other participants can look more information about the entity. This field is informational only. However, it can be used

by external appications to check that the entity in th ereal world corresponds to the one registered in th eblockchain.

`commercial_name` the name of the company as it appears in the official register of the country/region. For example, in the case of IN2 (a Spanish business), the name should be the one registered in the Business Registry of Spain.

`new_privatekey` is the Private Key of the new entity, in JWK format. In this case the new entity is IN2. Please make sure the server being called is highly trusted.

`parent_privatekey` is the Private Key of the entity owning/controlling the parent node in the domain name, in JWK format. In this case the parent node is *ala*, corresponding to Alastria. Please make sure the server being called is highly trusted. Ideally, the server has to be operated by the same entity calling the API.

3.4 Credential flows

3.4.1 Credential Issuance

The figure below describes the interaction flows between the Issuer and the Citizen. Here the term Issuer includes the mobile application of the Issuer Operator and the associated backend system of the Issuer Entity.

The main interaction consists on the transmission of the Verifiable Credential from the Issuer to the mobile of the Citizen. The transmission is initiated with a QR.

The flows and the APIs used are described in detail below.

The credential issuance process is the following:

Credential generation

- The diagram assumes that the Issuer Operator starts the process for th ecreation of the credential, but other initiation mechanisms could be used depending on the context.
- The system gathers existing data from the citizen from a previous identification process, like KYC.
- The system stores the information and generates a credential in the standard W3C Verifiable Credential format.
- The system then generates and displays a QR code that will be scanned by the Citizen to receive the Credential. The QR contains the URL in the Issuer's system where the credential can be retrieved.

Citizen receives the Credential

- The Citizen uses the webapp to scan the QR code displayed by the Issuer Operator
- The Citizen mobile webapp uses the URL in the QR to get the credential in JWT format, signed by the Issuer.

Citizen webapp verifies the credential and signature of Issuer

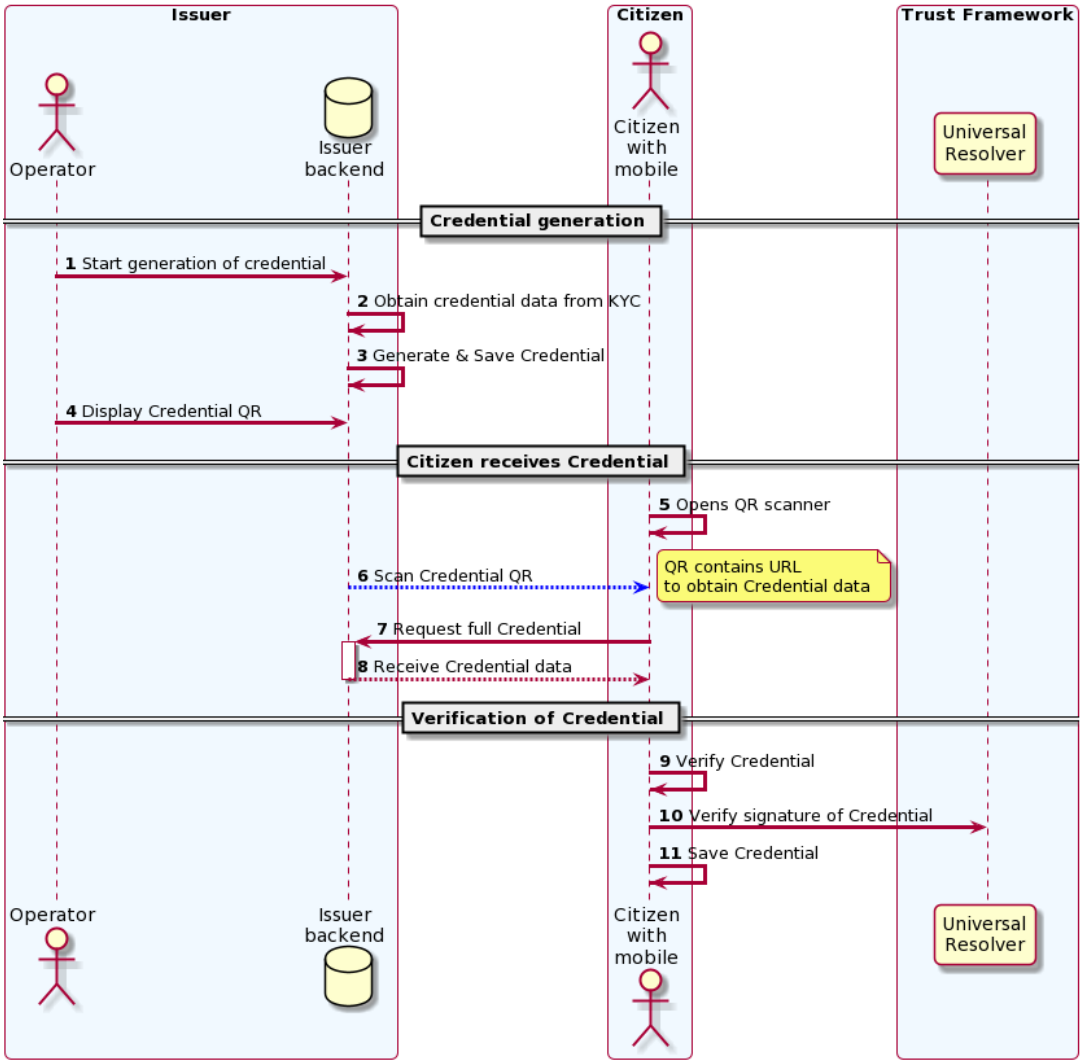

Fig. 3.5: Credential Issuance.

- The credential is verified as per the standard W3C Verifiable Credentials Implementation Guidelines.
- The verification includes resolving in the blockchain the identity of the Issuer Entity specified by the Issuer DID in the credential. The Issuer DID is registered in the blockchain and it includes the Public Key used by the Issuer Entity to digitally sign the credential.
- The Citizen mobile webapp uses a Universal Resolver to make this DID resolution and access the blockchain in read mode. The Universal Resolver is described in detail later in this document.
- After verification the credential is stored in the local storage of the Citizen mobile device. The user has also the option to store the credential in encrypted form in one or more of the personal cloud storage systems she has (Google Drive, MS Onedrive, Dorpbox, ...).

3.4.2 Credential Verification

The system supports the standard online verification process as is common in most implementations of an SSI system. But in addition it supports a special flow for on-person verification of credentials, for example when the credential has to be presented to a Verifier Operator in-person and it has to be verified by the Operator. This flow is useful when some process has to be performed in-person in the offices of the Verifier Entity, or even when for some reason it has to be performed out of the offices. In other words, when the citizen is not interacting directly with a web page of the Verifier Entity.

This is the flow represented in the following diagram.

3.5 ELSI: a DID Method for legal entities

The system supports several DID Methods using the Universal Resolver to resolve each DID into a corresponding DID Document. But the main DID Method used for legal persons, anchored into a Public-Permissioned blockchain, is *ELSI*: *did:elsi*.

3.5.1 ELSI DID syntax

The name ELSI stands for **E**TSI **L**egal person **S**emantics **I**dentifier, because it is based on the *Legal person semantic identifier* defined in the European Norm ETSI EN 319 412-1, related to digital signatures, peer entity authentication, data authentication as well as data confidentiality.

The ELSI DID Method refers only to legal persons, so we are using the *id-etsi-qcs-SemanticsId-Legal* definition described in Section 5.1 of ETSI EN 319 412-1.

Creating a DID is extremely simple and fully decentralized (does not require participation of any central authority), assuming that the legal person already exists. Its definition using ABNF syntax is:

```
did = "did:elsi:" id-etsi-qcs-SemanticsId-Legal
```

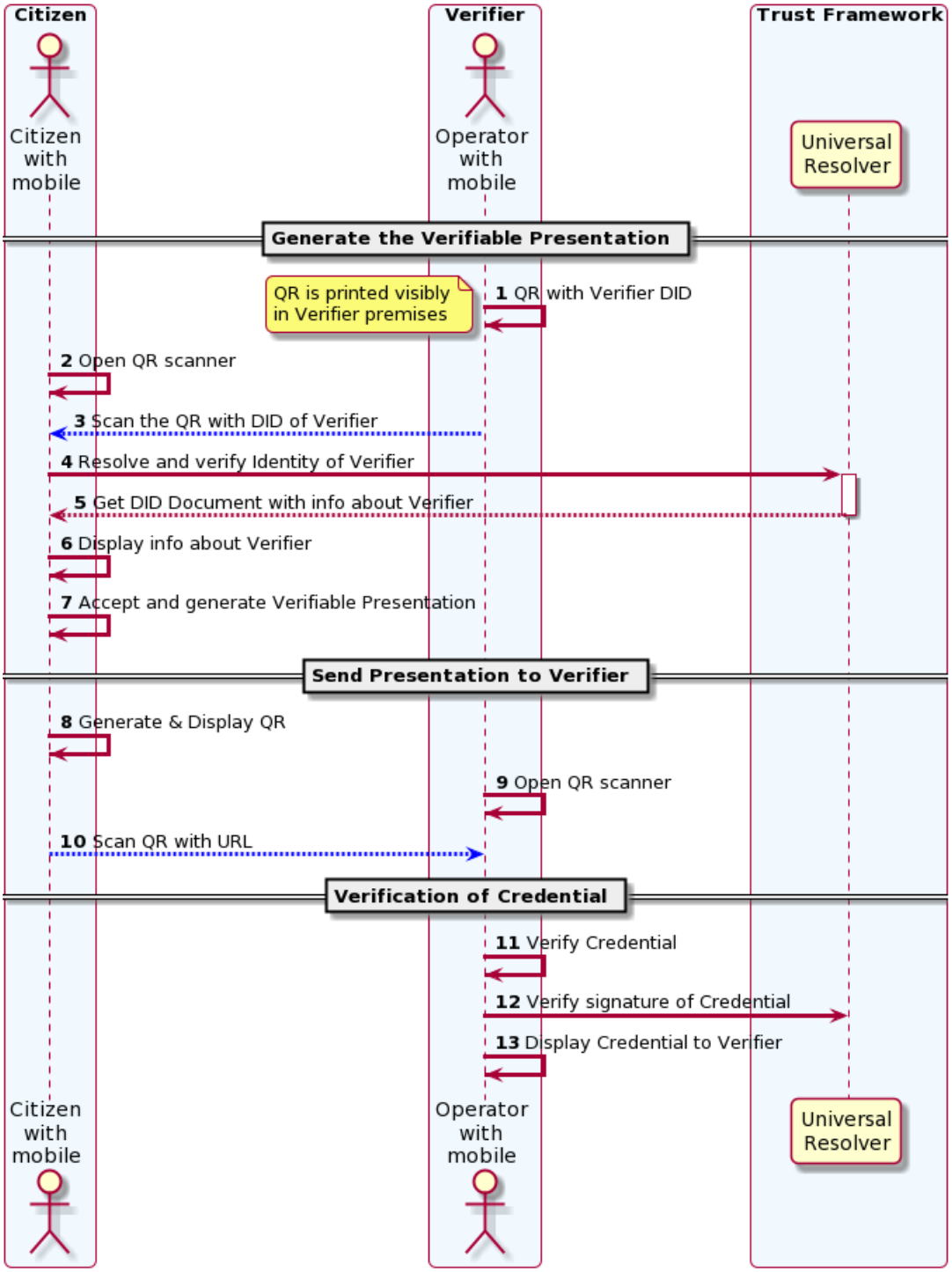

Fig. 3.6: Credential Verification

Which is the concatenation of the prefix *did:elsi:* with the legal person identifier defined in ETSI EN 319 412-1. For the full syntax, please refer to the standards document, but for the two most common basic identifiers (VAT and LEI) the identifier is composed of:

- 3 character legal person identity type reference, like *VAT* for identification based on a national value added tax identification number or *LEI* for the Legal Entity Identifier.
- 2 character ISO 3166 [2] country code;
- hyphen-minus "-" (0x2D (ASCII), U+002D (UTF-8)); and
- identifier (according to country and identity type reference).

Some examples of DIDs are the following:

Name	DID
ENDESA ENERGÍA (www.endesa.com)	**did:elsi:VATES-A81948077**
Ayuntamiento de Malaga (www.malaga.eu)	**did:elsi:VATES-P2906700F**
IN2 (www.ins.es)	**did:elsi:VATES-B60645900**
Inter-American Development Bank (www.iadb.org)	**did:elsi:LEIXG-VKU1UKDS9E7LYLMACP54**
DAA plc (Dublin Airport Authority) (www.daa.ie)	**did:elsi:LEIXG-635400HRFGVKXFHZ8O77**

3.5.2 ELSI DID Document

An example DID Document is the following:

```
{
"payload": {
    "@context": [
    "https://www.w3.org/ns/did/v1",
    "https://w3id.org/security/v1"
    ],
    "id": "did:elsi:VATES-B60645900",
    "verificationMethod": [
    {
        "id": "did:elsi:VATES-B60645900#key-verification",
        "type": "JwsVerificationKey2020",
        "controller": "did:elsi:VATES-B60645900",
        "publicKeyJwk": {
        "kid": "key-verification",
        "kty": "EC",
        "crv": "secp256k1",
        "x": "3K4iNuzPkcrHlEbhHE8vYXlF6K5xGZ2rdOrn3cQ-LnQ",
        "y": "9Z_l_hQLkq6aLuZz8gheq7R_o5ZUHUlxZ3IBGHsdzaA"
        }
    }
    ],
    "service": [
```

(continues on next page)

(continued from previous page)

```
    {
        "id": "did:elsi:VATES-B60645900#info",
        "type": "EntityCommercialInfo",
        "serviceEndpoint": "www.in2.es",
        "name": "IN2 Innovating 2gether"
    },
    {
        "id": "did:elsi:VATES-B60645900#sms",
        "type": "SecureMessagingService",
        "serviceEndpoint": "https://privatecred.hesusruiz.org/api"
    }
    ],
    "anchors": [
    {
        "id": "redt.alastria",
        "resolution": "UniversalResolver",
        "domain": "in2.ala",
        "ethereumAddress":
↪"0x8CDA8113567e633805e48c87747257E9FFAAdDF5"
    }
    ],
    "created": "2021-02-08T06:53:08Z",
    "updated": "2021-02-08T06:53:08Z"
}
}
```

3.6 PrivacyCred Verifiable Credentials

3.6.1 Data Model

The PrivacyCred credential uses the standard W3C Verifiable Credentials Data Model for its representation, with some extensions to fit the requirements of this use case.

The specific credential data is encoded in the credentialSubject field of the VC. The following two figures represent the complete VC, where it has been divided in two parts to facilitate visualization.

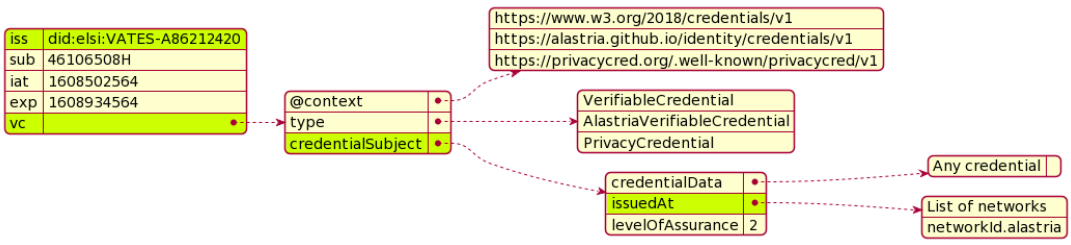

Fig. 3.7: W3C Verifiable Credential and extensions

The figure above represents the VC with standard fields and some extensions.

1. The iss field (issuer in VC terminology), uses the DID method `elsi`, specific for legal persons and explained in a section below.

2. There is an extension to specify the blockchain network (or networks) where the VC can be verified. More precisely, the `issuedAt` field of `credentialSubject` specifies the networks where the identity for the legal person that issued the credential can be verified.

   A legal person can have its *elsi* DID registered in one or more networks, and the same credential can be verified using any of those networks. The trust on the credential depends on the trust on the registration procedure of the identity of the signer. The Verifier entity can choose to verify the credential in whatever network is trusted to the Verifier.

   This mechanism provides a lot of flexibility in interoperability schemes across networks. More details are described in the section on interoperability.

3.6.2 Example of Verifiable Credential


```
{
    "exp": 1614770844,
    "iat": 1614252444,
    "iss": "did:elsi:VATES-X12345678X",
    "sub": "46106508H",
    "uuid": "829588b3162249d28f3eae5e84349777",
    "vc": {
            "@context": [
                "https://www.w3.org/2018/credentials/v1",
                "https://alastria.github.io/identity/credentials/v1
↪",
                "https://privacycred.org/.well-known/privacycred/v1"
            ],
            "type": [
                "VerifiableCredential",
                "AlastriaVerifiableCredential",
                "PrivacyCredential"
            ],
            "credentialSchema": {
                "id": "PrivacyCredential",
                "type": "JsonSchemaValidator2018"
            },
            "credentialSubject": {
                "privacyCredential": {
                        "citizen": {
                            "dob": "27-04-1982",
                            "idnumber": "46106508H",
                            "name": "COSTA/ALBERTO",
                            "type": "atRisk"
                        },
                        "comments": "These are some comments",
                },
```


(continues on next page)

(continued from previous page)

```
                "issuedAt": [
                        "redt.alastria"
                ],
                "levelOfAssurance": 2
            }
    }
}
```

3.7 Verification of the credentials

The system includes two APIs to help client applications with the verification of credentials received from other actors in the ecosystem. The choice of API depends on the trust level of the client application on the server implementing the APIs

GET /api/did/v1/identifiers/(**string:** *DID*)
Resolves a DID and returns the DID Document (JSON format), if it exists. It supports four DID methods: **ebsi**, **elsi**, **ala**, **peer**.

Only **PEER** and **ELSI** (*https://github.com/hesusruiz/SafeIsland#62-elsi-a-novel-did-method-for-legal-entities*) are directly implemented by this API. The others are delegated to be resolved by their respective implementations.

For example, for **EBSI** we call the corresponding Universal Resolver API, currently in testing and available at https://api.ebsi.xyz/did/v1/identifiers/{did}

Query Parameters

- **DID** (*string*) – The DID to resolve into a DID Document.

Response JSON Object

- **didDocument** (*payload*) – The DID document associated to the input DID

Status Codes

- 200 OK – no error
- 404 Not Found – error resolving the DID

Example request:

```
GET /api/did/v1/identifiers/did:elsi:VATES-B60645900 HTTP/1.1
Host: example.com
Accept: application/json
```

Example response:

```
HTTP/1.1 200 OK
Vary: Accept
Content-Type: text/javascript
```

(continues on next page)

(continued from previous page)


```
{
    "payload": {
        "@context": [
            "https://www.w3.org/ns/did/v1",
            "https://w3id.org/security/v1"
        ],
        "id": "did:elsi:VATES-B60645900",
        "verificationMethod": [
            {
                "id": "did:elsi:VATES-B60645900#key-verification
↪",
                "type": "JwsVerificationKey2020",
                "controller": "did:elsi:VATES-B60645900",
                "publicKeyJwk": {
                    "kid": "key-verification",
                    "kty": "EC",
                    "crv": "secp256k1",
                    "x":
↪"3K4iNuzPkcrHlEbhHE8vYXlF6K5xGZ2rdOrn3cQ-LnQ",
                    "y": "9Z_l_hQLkq6aLuZz8gheq7R_
↪o5ZUHUlxZ3IBGHsdzaA"
                }
            }
        ],
        "service": [
            {
                "id": "did:elsi:VATES-B60645900#info",
                "type": "EntityCommercialInfo",
                "serviceEndpoint": "www.in2.es",
                "name": "IN2 Innovating 2gether"
            },
            {
                "id": "did:elsi:VATES-B60645900#sms",
                "type": "SecureMessagingService",
                "serviceEndpoint": "https://privatecred.
↪hesusruiz.org/api"
            }
        ],
        "anchors": [
            {
                "id": "redt.alastria",
                "resolution": "UniversalResolver",
                "domain": "in2.ala",
                "ethereumAddress":
↪"0x8CDA8113567e633805e48c87747257E9FFAAdDF5"
            }
        ],
        "created": "2021-02-08T06:53:08Z",
```


(continues on next page)

(continued from previous page)

```
        "updated": "2021-02-08T06:53:08Z"
    }
}
```

In general, validating a credential involves the following:

1. Deserialize the JWT without verifying it (we do not yet have the public key).
2. Get the `kid` property from the header (the JOSE header of the JWT).
3. The `kid` has the format did#id where `did` is the DID of the issuer and `id` is the identifier of the key in the DIDDocument associated to the DID.
4. Perform resolution of the DID of the issuer with the Universal Resolver API.
5. Get the public key specified inside the DIDDocument.
6. Verify the JWT using the public key associated to the DID.
7. Verify that the DID in the `iss` field of the JWT payload is the same as the one that signed the JWT.

`POST /api/verifiable-credential/v1/verifiable-credential-validations`

Is the easiest one to use and the one requiring higher level of trust. The client app just passes the JWT in the JWS Compact Serialization format (RFC 7519) as the body of a POST request and the server verifies the credential and credential signature using internally the Universal Resolver API for resolving the DID of the Issuer and checking its digital signature.

Request JSON Object

- **`credential`** (*`JWT`*) – The credential in JWT format.

Response JSON Object

- **`claims`** (*`object`*) – The JSON object with the verified claims in the JWT. Otherwise, an error

Status Codes

- 200 OK – no error
- 404 Not Found – error resolving the DID

The easiest one to use is `/api/verifiable-credential/v1/verifiable-credential-validations`, and it is the one requiring higher level of trust. The client app just passes the JWT in the JWS Compact Serialization format (RFC 7519) as the body of a POST request and the server verifies the credential and credential signature using internally the Universal Resolver API for resolving the DID of the Issuer and checking its digital signature.

`/api/did/v1/identifiers/(string:DID)` is the Universal Resolver API. The client application will have to perform the validations that the server does in the previous call.

4 References