esatus® AG
Enforcing Information Security

How to Launch Self-Sovereign Identity Technology for Corporate IT Access

With Self-Sovereign Identity (SSI) technology, users are in full control over their personal data. No longer are they dependent on central service providers who collect usage information on them and who might turn off access to important services at will. By using an SSI wallet app, users themselves control every connection to information issuers and information consumers and can decide which information to share with whom. Now, with all required technology being available, esatus AG has equipped all of its employees with its own wallet app and meaningful credentials rooted in the Sovrin Network. They can now use them to access in-house IT applications.

Dr. André Kudra, Executive Board Member / CIO
Sebastian Weidenbach, Head of Technical Consulting & Solutions, CISSP

Jumpstarting practical SSI application

Using SSI technology for Identity & Access (I&A) allows a privacy-preserving way of interaction between services and users. Furthermore, deriving access from flexible combinations of trustworthy real-world facts without being bound to contextual borders is easy to accomplish.

In this document the reader will learn how esatus AG designed their credential-based IT access concept as well as the technology which had to be set up. It also provides insight into the migration process and how employees reacted to the new technology. It covers *Hyperledger Indy* and *Hyperledger Aries* technology as well as the use of the *DLT (distributed ledger technology) Sovrin Network* [1] – the productive net, or MainNet – for establishing the required level of trust between the systems. More information about SSI basics as well as thoughts on technology and network selection can be found in the dotmagazine article *Self-Sovereign Identity Technology meets Identity & Access Management* [2] and in the esatus German language whitepaper *Identity & Access Management (IAM) – Realisiert mit Self-Sovereign Identity (SSI)* [3].

Eat your own SSI I&A

Already during the development phase of its credential-based Identity & Access solution *SeLF*, esatus AG wanted its first productive application in-house. The expected outcome is that the organization gains a better understanding of the product readiness and insights into the user acceptance. As for relevant applications to be access-controlled with SSI first, it became evident that the internal wiki system, in frequent use by most employees, was suitable. The underlying commercial product, *Atlassian Confluence*, is compatible with various authentication and authorization sources, of which many are supported by SeLF. SeLF itself uses *SSI-native* access control. It was created with full Hyperledger Aries standards compatibility in mind. Hence, an Aries-compatible wallet app is required on the end user side. In 2019, esatus AG was one of the partners in the project *Let's initiate self-sovereign identity* – in brief *Lissi* – of Main Incubator GmbH, which very successfully delivered on its goal of developing a mobile app. The outcome was leveraged as foundation for the I&A-specialized esatus wallet app.

Being well aware of the trade-off between privacy-preserving, secure solutions and ease of use, a key design goal for the esatus wallet app was making its use as comfortable as possible. SSI-based I&A will only have a chance if its usability is at least compa-

rable to other solutions already in existence. For the user, most of the interaction happens in the mobile app. Hence, in practical beta-testing, each screen and every single click was considered carefully. The SeLF development team quickly realized that equipping the wallet app with selected comfort enhancements was vital for usability. The esatus wallet app is optimized for frequent interactions, which happen particularly in authentication use cases. Furthermore, the *revocation* functionality – critical in I&A use cases – was implemented in the wallet app. Key features now available in the esatus wallet app:

- Auto-fill proof request form
- Auto-respond for trusted connections (multiple strategies available to the user)
- Enhanced notification appearance
- Full revocation support

Going productive with SSI I&A meant switching to the Sovrin MainNet from the esatus test network. Some features of the app, like the Sovrin *Transaction Author Agreement (TAA)* processing, had to be made configurable because they would only become functional after a network update. Additionally, the public DID for esatus `Ar1YzNwcM74M2Z4XKUWXMW` had to be provided with the endorser status so that schemas, credential definitions and revocation registries could be written to the ledger.

Tailoring an SSI access concept

In most organizations, *access concepts* including workflows for access request and approval are modeled around organizational structures. Based on these, different abstraction clusters for access decisions – a person's access to a system, system function or data set is enabled or not – can be derived:

- Being employed at a company
- Being a member of a certain department
- Being a member of a certain project
- Combinations of the above

Also in the esatus case, the SSI access concept is tailored to the organizational structure of the company. Thanks to SSI, this is easy and straightforward to implement. The esatus department structures are quite straightforward as well, as depicted in the organizational chart visualization in figure 1.

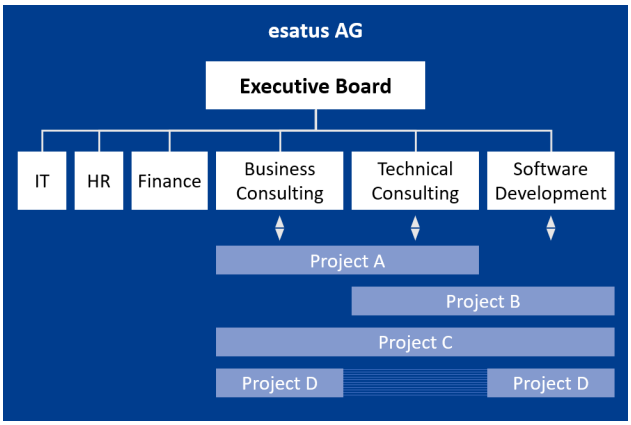

Figure 1: Organizational chart

As an example, an access decision could be solely based on the organizational membership: Everyone being able to prove that she is an employee of esatus AG may access the company intranet. To achieve this in an SSI access concept, the organizational structure has to be modeled with *schemas* which include relevant *attributes* for access decisions. Schemas are required for creating *credential definitions* which in turn can be used for issuing *credentials* to an individual's wallet. How this is achieved with the esatus SSI access concept will be covered in full detail in the next chapters.

From org structure to schema

Schemas define the attributes that can be included in a credential. While some organizations with own credential requirements will need to write their own schemas, as usage grows many will find an existing schema that suits their use case, thus will not need to write their own. As there were no existing schemas available which could be used, the esatus team had to create new schemas. Leveraging its extensive enterprise I&A experience for the schema

definition, esatus could depict its own org structure while making a general access model available for any other company to use. An expiry date and an issuing date was included in every schema to reference when the user obtained this credential and for how long it is valid directly in its content. This is useful even if a revocation of a credential is possible. As every employee needs to be able to prove she belongs to the company, an employment schema was created. The department information was intentionally omitted in this schema, as departments regularly change more often than the general employment data. The same applies to project memberships. Therefore, the created employment schema should be generic and applicable to many companies:

TxID: `Ar1YzNwcM74M2Z4XKUWXMW:2:EmploymentSchema:1.0`, Seqno: 54019

- employer
- employeeID
- userID
- email
- employmentType
- issuingDate
- expiryDate

Consequentially, there was a need to depict the department membership of an employee. The position within the department also matters, hence it was included in the schema. Companies can be located in different places, a location attribute was incorporated. In the esatus case, Hamburg, Munich and Langen are possible.

TxID: `Ar1YzNwcM74M2Z4XKUWXMW:2:DepartmentMemberSchema:1.0`, Seqno: 54020

- position
- location
- departmentID
- department
- issuingDate
- expiryDate

As many employees are involved in several projects, either internally or for third parties, i.e. clients, a separate schema reflecting project memberships was required. Typically, a project has a name and an identifier and an employee has a specific role in it. This resulted in the following schema:

TxID: `Ar1YzNwcM74M2Z4XKUWXMW:2:ProjectMemberSchema:1.0`, Seqno: 54021

- role
- projectID
- project
- issuingDate
- expiryDate

Employee-specific data was split up in a personal data and a postal address data schema. Personal data rarely changes, the postal address may more frequently, i.e. when an employee moves. This resulted in the following schemas:

TxID: `Ar1YzNwcM74M2Z4XKUWXMW:2:PersonalDataSchema:1.0`, Seqno: 54022

- title
- firstname
- lastname
- nationality
- dateOfBirth
- gender
- issuingDate
- expiryDate

TxID: `Ar1YzNwcM74M2Z4XKUWXMW:2:PostalAddressDataSchema:1.0`, Seqno: 54023

- street
- postOfficeBox
- postalCode
- locality
- region
- country
- issuingDate
- expiryDate

The usage of the esatus schemas is under monitoring to see if and how they offer value for the SSI community. esatus is open for comments and suggestions to evolve and improve its schemas.

From schema to credential definition

The deployed schemas define a broadly usable namespace for credentials. To use them as an issuer, implementing a separate instance of them is required, i.e. a credential definition. These restrict credential issuing to their creator. Consequentially, users receive credentials from a specific issuing authority. Two or more companies can use the same schema but they have to create their specific credential definition to issue their own credentials, which can be indistinctly attributed to them.
If revocation of issued credentials is required, credential definitions have to be implemented with revocation support. This is usually the case for all I&A scenarios – if an access is enabled via an attribute from a credential, the issuer would most likely want be able to revoke it by invalidating a credential. Revocation support for a credential means that a *revocation registry* (which is based on a cryptographic accumulator) has to be deployed together with the credential definition. With the accumulator persisted on the Sovrin ledger, it is possible to check if a credential is revoked (or not) at any given time. The initial credential definitions created for esatus are created as non-revocable to gain experiences with this setup first. Hence current credentials are issued with only a short period of validity time. In the meantime, a revocation registry was added, i.e. the next credentials issued will be revocable.

Access compliance considerations

Based on best practices, several compliance policies are enforced. This begins with the access concept for SeLF. It is fully SSI credential based, i.e. SSI-native without any "classic" access control mechanisms in-between. The following principles apply:

- The executive board (CEO and CIO) can issue any credential, including department heads and C-level positions.
- HR department members can issue credentials for employment, department membership (excluding department heads and C-level position), personal data and postal address.
- Every department head can issue any project member credential.
- The IT infrastructure department can implement and change rules for their applications.

A person can never issue credentials for himself, a different person must issue those. Also, an issuer can only issue credentials which are on a lower level of privilege than their own level. E.g. the HR head cannot issue other head credentials, only the executive board can.
A general policy is that all credentials become invalid after one year, for privileged users after six months. Upon expiration, they have to be reissued by the responsible person.
The issuer is responsible for false credentials issued, he is mandated to correct them in case of mistakes. Upon reception, a credential receiver is required to check the contents, i.e. attributes, for correctness. If an error is noted, they have to decline the credential and demand a corrected one.
For privileged access, the user needs two different, linked credentials. Dual control is enforced, these credentials can only be issued by two different persons. One issuer cannot compromise this logic. Only if the privileged user can present linked credentials, access is granted. SeLF administrative access is considered privileged and always monitored. The SeLF audit trail permanently logs which rules have been implemented or modified and when this was done.
In an employment termination case, the employment credential is revoked. Without this credential, no further access is granted in any connected system.

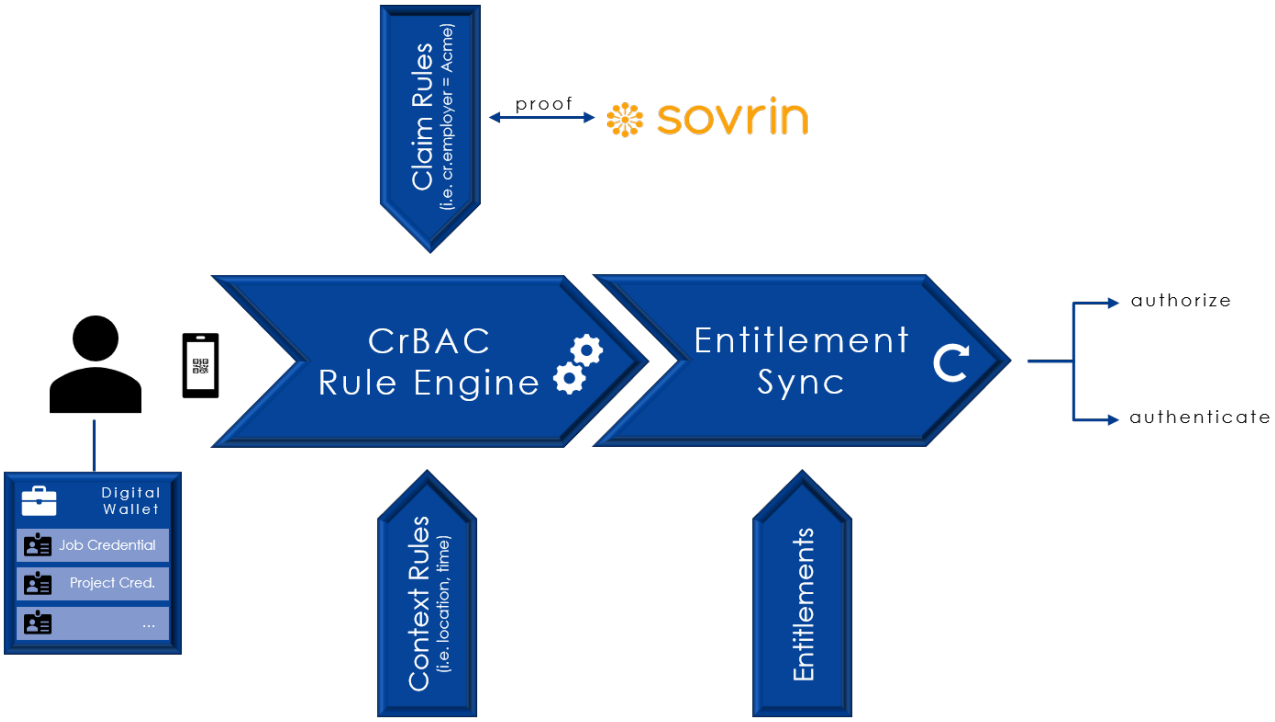

Figure 2: SSI-based CrBAC with SeLF Rule Engine

Modeling access rules

The heart of the esatus access concept is a set of *access rules* which determine who has access to what. The authentication and authorization is powered by *credential-based access control (CrBAC)*. The SeLF *rule engine* is populated with *credential rules*, e.g. "employer must equal esatus AG", and *context rules*, e.g. location and time. Credential rules fully leverage the attributes made available in issued credentials, based on respective schemas and credential definitions. The rules ultimately manage the access rights for connected applications, all via the SeLF rule engine. The rule processing mechanics are depicted in figure 2.

In the initial esatus implementation for the internal wiki system, a simple access rule was created:

All employees can access the wiki system.

The results of the rule processing lead to synchronizing the underlying entitlements in the wiki system, making authentication and authorization possible. To build the bridge between SSI and the Atlassian Confluence system, supported access control mechanisms are "fueled" by SSI credentials. The easy choice was to utilize *SAML (Security Assertion Markup Language)* authentication and *LDAP (Lightweight Directory Access Protocol)* authorization. Both are supported by SeLF, i.e. can be provisioned by the SeLF agent, and Confluence requires the *re:solution* plugin to enable SAML. By building the connection between the SeLF agent and Confluence via configuration settings and access rules, users are able perform their wiki login with their credentials.

Once done, every employee can access the wiki system: For signing in, she needs to scan a *QR (Quick Response)* code on the login page with the esatus wallet app on the smartphone. A *proof request* will be issued by the SeLF agent to her wallet app. She has to use her credentials to fulfill the proof request, i.e. confirm transmission of the required attributes to the SeLF agent. SeLF checks validity and rule match to decide if access can be granted or not.

Conducting the staged roll-out

The esatus SSI roll-out was conducted in several phases, starting with an initial phase for the SeLF team, which included access to the wiki system to test all functions in advance. After successful test completion, the roll-out was extended to all employees. To do this, the wallet app was made available to all employees for installation on their iOS smartphone devices (a feature-identical Android version of the wallet app is also available). After individual deployment each employee scheduled a session with the HR representative to receive their credentials. The HR representative was trained in advance to be able to answer questions regarding the roll-out process, the handling of the wallet app and the login procedure. During the whole procedure no noteworthy problems occurred.

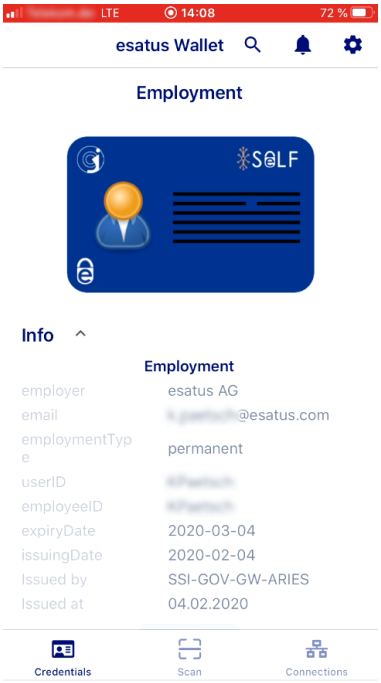

Figure 3: Employment credential

A real-life access example

Sam is new at esatus AG. On behalf of the company, Hans is running through the onboarding steps. He creates Sam's HR identity as he was not active in esatus realm before. Via SeLF, Hans establishes a connection between Sam's wallet app. This connection is protected by cryptographic keys known only to Sam's wallet app and the connected SeLF agent. Hans then issues Sam's credentials. Hans sends the new credential to Sam which he has to validate and accept. It is then securely stored in his wallet app (see figure 3).

Figure 4: Wiki Login QR code

As the internal esatus wiki login leverages SSI credentials, Sam is immediately able to log in. He can initiate the wiki login by scanning a QR code displayed on the wiki login page (see figure 4) with his wallet app. A SAML request is channeled to SeLF which initiates a request for several attributes to Sam's wallet app (see figure 5).

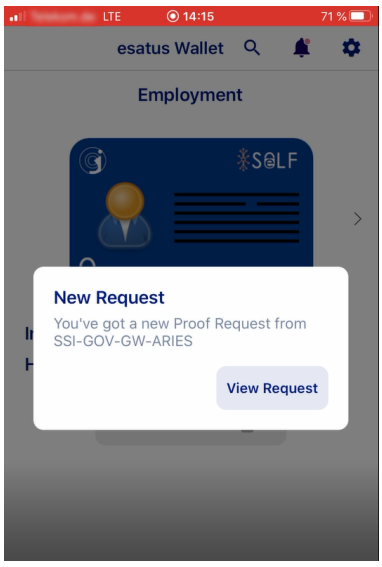

Figure 5: Wallet app request

For the wiki login, the attributes expiryDate, employer and userID have to be revealed as a proof to the SeLF agent to conduct the access decision (see figure 6). After Sam confirms sending of the attributes, SeLF checks validity on the Sovrin ledger. As the credential was just issued and is not revoked, he will gain access to the wiki system as it was proven he is an employee of esatus AG and the credential is not expired yet.

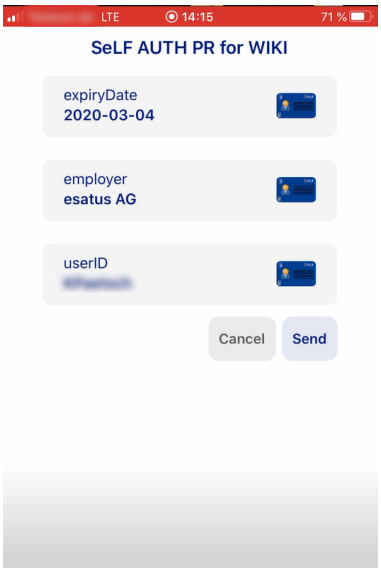

Figure 6: Wallet app authentication

Figure 7: SeLF SSI enablement matrix

A good start, what's next?

This was only a first step of rolling out SSI-based I&A at esatus AG. The solution will be applied to further internal applications, e.g. work time booking, file stores, IT ticketing, agile work planning, code repository and possibly even to the Windows login. More complex access rules will certainly be necessary to adequately reflect the organizational structures in the system domain. Making full use of the available credentials and pertaining attributes is a natural consequence. The revocation mechanisms will be comprehensively applied for all newly issued credentials. Applied SSI is not just an internal benefit for esatus AG. Many popular cloud applications can be SSI-enabled easily, figure 7 shows examples.

Concluding remarks

All aspects considered, the roll-out was a great success and one of many steps towards a broad usage of Self-Sovereign Identity. The planning, design, concept and development work in advance was clearly the major part. Especially SeLF as the enterprise I&A suite and the I&A-specialized wallet app were important milestones for success. Now, esatus is gathering all feedback from its employees, who are quite pleased to be SSI users but are advised to be cautious observers for bugs and improvement potentials. Everyone is excited and looking forward to further activities. esatus AG is one of the pioneers in implementing and using SSI solutions for I&A in a corporate context, hence in a prime position for evolving SSI and driving its global success.

[1] Sovrin Foundation: Sovrin™: A Protocol and Token for Self-Sovereign Identity and Decentralized Trust, A White Paper from the Sovrin Foundation, January 2018

[2] Sebastian Weidenbach: Self-Sovereign Identity Technology meets Identity & Access Management, September 2019

[3] esatus AG: Identity & Access Management (IAM) – Realisiert mit Self-Sovereign Identity (SSI), August 2019

esatus AG is a medium-sized IT consulting company. True to its mission "Enforcing Information Security" **esatus** AG is the qualified, experienced and flexible partner for any information security project. For its demanding customers, **esatus** AG implements optimal, tailor-made solutions addressing challenges in Identity & Access (I&A), IT Security, and Governance, Risk und Compliance (GRC). The customer satisfaction is the guideline, on which all of the company's actions are orientated.

Published by **esatus** AG in February 2020.