UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Computer Science Curriculum

Gediminas Milašius

Exploring integration complexity of different multi-national eID authentication solutions in the EU private sector

Master's Thesis (24 ECTS)

Supervisor(s): Arnis Paršovs, MSc

Tartu 2022

Exploring integration complexity of different multi-national eID authentication solutions in the EU private sector

Abstract:
Many interpreting program languages are dynamically typed, such as Visual Basic or Python. As a result, it is easy to write programs that crash due to mismatches of provided and expected data types. One possible solution to this problem is automatic type derivation during compilation. In this work, we consider study how to detect type errors in the WHITESPACE language by using fourth order logic formulae as annotations. The main result of this thesis is a new triple-exponential type inference algorithm for the fourth order logic formulae. This is a significant advancement as the question whether there exists such an algorithm was an open question. All previous attempts to solve the problem lead lead to logical inconsistencies or required tedious user interaction in terms of interpretative dance. Although the resulting algorithm is slightly inefficient, it can be used to detect obscure programming bugs in the WHITESPACE language. The latter significantly improves productivity. Our practical experiments showed that productivity is comparable to average Java programmer. From a theoretical viewpoint, the result is only a small advancement in rigorous treatment of higher order logic formulae. The results obtained by us do not generalise to formulae with the fifth or higher order.

Keywords:

List of keywords

CERCS:

CERCS code and name: `https://www.etis.ee/Portal/Classifiers/Details/d3717f7b-bec8-4cd9-8ea4-c89cd56ca46e`

Tüübituletus neljandat järku loogikavalemitele

Lühikokkuvõte:

One or two sentences providing a basic introduction to the field, comprehensible to a scientist in any discipline.

Two to three sentences of more detailed background, comprehensible to scientists in related disciplines.

One sentence clearly stating the general problem being addressed by this particular study.

One sentence summarising the main result (with the words “here we show´´ or their equivalent).

Two or three sentences explaining what the main result reveals in direct comparison to what was thought to be the case previously, or how the main result adds to previous knowledge.

One or two sentences to put the results into a more general context.

Two or three sentences to provide a broader perspective, readily comprehensible to a scientist in any discipline, may be included in the first paragraph if the editor considers that the accessibility of the paper is significantly enhanced by their inclusion.

Võtmesõnad:

List of keywords

CERCS:

CERCS kood ja nimetus: `https://www.etis.ee/Portal/Classifiers/Details/d3717f7b-bec8-4cd9-8ea4-c89cd56ca46e`

Contents

Unsolved issues

1 Introduction

1.1 Motivation

With the emergence of COVID-19, work from home has rapidly grown in popularity. It has been especially noticeable in the IT industry. This phenomenon has led some businesses to transition to operate fully remote [1], allowing for potential customers, clients, and employees to operate with the companies' IT systems from all around the globe.

Identity verification is a significant roadblock when establishing a remote work policy. In some managerial businesses, such as logistics, it is essential to assure the authenticity of persons signing in to perform their duties. This security requirement is essential for those dealing with contracts, where one input can cost thousands of euros in losses. Traditionally, as work was always on-premises, it was easy to verify the identity with the help of an identity document. With the constraints of operations being fully remote, companies no longer have the luxury to perform such a check.

Establishing identity online for potential employees and clients is not the only use case for digital identity. Organizations such as the British Council employ privacy undermining practices. As part of registration to the IELTS exam, they require their customers to submit a photocopy of their identity document for verification purposes [2]. This process is a significant privacy concern since anyone could replicate the uploaded document. Having no agency over their documents is of great concern for the end-users, making them reluctant to use the company services. Replacing the document upload with a digital signature check is a more secure and trustworthy way of performing business.

After the EU introduced the eIDAS regulation, an alternative method for identity verification became available. All EU member states are mandated to implement an eID solution in their country and recognize other countries' eID solutions [3]. Each eID solution guarantees some degree of authenticity, from substantial to high, allowing for verifying a persons' identity with trustworthy means.

Particular risks exist that businesses must be aware of before integrating an eID authentication service. There are no comprehensive resources outlining the obstacles and costs of implementing eID authentication in the private sector. Unknown risks are an excellent deterrent for innovation and make companies reluctant to use new technologies. Proper research into this subject may lead companies to take risks associated with implementing new technology and kickstart the mainstream adoption of eIDs in the private sector.

1.2 Research Problem

The main goal of the thesis is to investigate if the advantages provided by eIDs are sufficient enough to outweigh the risks associated with adopting new technology in the free market and shine a light on the costs associated with implementation. From this goal, the extracted research question is as follows:

What is the best eID authentication option available for an Estonian EU targeting enterprise for use in their Web-based Single Sign-On (SSO)?

Maybe it would be good to change to "is the market ready"

The research question can be refined further into additional sub-questions:

- What advantages do eIDs provide?
- How should an authentication scheme change to support eID?
- What privacy considerations must companies take when processing user data?
- What are the different eID authentication providers available to Estonia's private sector?
  - What risks does the eID provider transfer away from the consumer company?
  - What is the market reach (in countries) of a given solution?
  - Where are the weak points in the protocol used? How should a company assess them?
  - How expensive is the solution to operate?

With these questions answered, it should be possible for any company to understand the risks and benefits of any given solution eID solution.

1.3 Scope and goal

There will be some assumptions about the company wishing to implement eID authentication in the thesis.

Company already uses an HTTP-based SSO (in the cloud or on-premises) Non-HTTP-based SSOs or other esoteric authentication systems will not be considered.

Company is committed to getting some form of eID authentication system in place This means market research was done, and the management found that it is favorable to invest in eID authentication, as the users would adopt this authentication mechanism.

The eID provider must be accessible by an Estonian company Other countries may also provide eID solutions. However, for the scope of the thesis, only solutions available to Estonian citizens and companies will be considered.

1.4 Contribution

The thesis aims to fill the research gap for the use of eID in the private sector. There is some research about connecting eIDAS nodes. However, the focus was on connecting eIDAS nodes of other countries and not connecting customers to the eIDAS infrastructure, which is mainly off-limits.

The thesis contains the following contributions:

1. enumeration of eID service providers in Estonia;
2. analysis of personal data storage under GDPR for use on authentication;
3. comparison of the different flavors of eID service providers;
4. assessment of various data transfer protocols eID service providers use;
5. display of example on how a company could implement an eID service into an SSO;
6. disclose a vulnerability in Dokobit's identity gateway;
7. feedback on the usability complexity for eeID project;

Structure of work The document will consist of the following main chapters:

ISO standard to pick trust level for if companies even need it

What are different options in eIDAS, e.g. what is a QESSD

What are the differences between primary services and middlemen, advantages disadvantages

What are the weakpoints in the company structrue

What is the research model?

Findings about ID Card, Dokobit, and eeID

Non web-based SSO?

Ponder about the advantages of middleware pseudominization? Say instead of personal code you get some arbitrary ID that matters only in the system

2 Background

2.1 Electronic identity eID

In Estonia, digital identity has been around for over 20 years [4]. The Estonian government issues ID cards loaded with certificates that enable cardholders to identify themselves digitally. Estonia's early adoption of eID, the political focus on digital government, has led to over 89% of internet users accessing the e-government, making it the leading country in the EU [5]. This adoption rate is in stark contrast to another EU country - Romania, where the first easy access to eIDs came in the form of new chip ID cards in August of 2021 [6]. In that country, only 16% of internet users access government services online. The adoption of eIDs in other countries lands somewhere between these two extremes.

It is clear that if a company wishes to implement eID authentication in a given country, it must realize the differences between the adoption rates. For some countries' citizens, the button "sign in with eID" will instinctively make them draw their smart card or other eID capable device, where it would confuse and repulse others. Depending on the country's market a company would like to access, eID sign-in may attract or repel potential clients. Early adopters must be aware of the widespread adoption of the eID infrastructure.

eID adoption in Estonia and Lithuania On the surface, Estonia and Lithuania have the exact eID solutions: Bank Link, ID card, Mobile-ID, and Smart-ID. However, even with the same infrastructure, we see many inconsistencies even in the case of just these two countries.

Consider Lithuania. It is possible to connect from a centralized website `https://epaslaugos.lt` to access the public sector services [7]. Here it is possible to sign in via bank link, ID card, and Mobile-ID. Smart-ID is not an option. The omission of Smart-ID is strange, as most banks support sign-in via three major eID providers, including Smart-ID, and actively encourage switching to it.

Some banks listed on the public sector gateway portal, like PaySera, provide significant security concerns. With that bank, it is possible to access the e-government services with only email, password, and a 2FA code sent to the registered person's phone number . For similar security concerns, Estonia's Information System Authority has taken steps to deprecate bank link [8] from use in TARA, Estonia's gateway to e-government.

source: I did it myself 02-27

In Estonia, all three major authentication options, ID card, Mobile-ID, and Smart-ID, can be used to access the e-government.

2.2 eIDAS

The eIDAS regulation [3] provided the groundwork for recognizing the signatures issued by other EU countries by imposing strict liability and mutual-recognition requirements. The regulation introduced the concept of a Trust Service Provider (TSP), which allowed relying parties to have a trust anchor. Before, with digital signatures, it was possible to verify that the person signed documents with a certificate and it was valid. However, there was no good way of ascertaining if a trusted authority issued the signing certificate. The role of establishing trust is the purpose of TSPs. Each member state maintains a list of their TSPs, where each TSP is certified to perform specific tasks, such as timestamping or issuing signing certificates.

The regulation also requires member states to establish eID systems and integrate them into a federal plan if they haven't already. This regulation was also the basis for creating the eIDAS node network [9]. These nodes connect across country borders, allowing users to authenticate with the eID of their home (eID issuer) country in the host (current residence) country. The eIDAS authentication protocol redirects the authentication requests to the appropriate country, federating the identification process. For the institutions trying to target the EU market, this provides a significant advantage since access to one node would mean access to all nodes in the EU.

The main issue private companies encounter when accessing the network is the highly exclusive access. The eIDAS network is only concerned about connecting countries. How and who can access an eIDAS node is up to the member states to decide.

eIDAS notifications in Estonia and Lithuania For countries to communicate through the eIDAS node network, the governments must first notify the European Commission about what eID authentication methods they will provide [3]. Other countries can then use these methods to authenticate foreign citizens into their public services.

In the case of Estonia, the country has notified the European Commission about its smart card and Mobile-ID authentication methods [10]. Smart-ID is not a permitted method of authentication in the context of eIDAS. In Lithuania's case, only the smart card solution is allowed - no mobile sign-in methods have been notified [10].

The governments of these two countries have revealed a gap of what they consider to be a secure and trusted source of eID and what they are willing to be held liable for in the context of the eIDAS network.

2.3 eID providers in Estonia

Applied Cyber Security Group of the University of Tartu maintains a list of e-services [11] that uses at least one eID authentication method in Estonia. The following authentication methods were listed: Bank Link, ID-card, Mobile-ID, Smart-ID, TARA, and HarID.

While the list does not count Dokobit and eeID as primary eID providers, they list them as consumers of at least of schemes mentioned before. Their business model is to act as an intermediary for ease of integration.

2.3.1 Bank link

Banks have initially created this authentication method to provide close integration with e-commerce providers to receive risk-free payments [12]. Over time it saw an additional use case - secure and trustworthy authentication method for the public and private services [13].

Researchers found that the bank link protocol, while applicable, was "extremely insecure" [14]. From March of 2021, RIA has disabled the use of bank link to access public services [8], which accounted for only 1 percent of all authentications, also revealing its popularity in public.

Due to the lack of security auditing required to satisfy eIDAS, poor market reach, and no support from the government, this authentication method will not be discussed in the scope of this thesis.

2.3.2 Smart card (ID card)

ID cards are the most popular way to access their eID in Estonia, primarily due to the legal requirement of having one. The Identity Documents Act [4] states that all EU, not only Estonian, citizens residing in Estonia must hold an ID card, with which they could access public services online. Interestingly, this requirement caused the government to issue more ID cards than there are people in Estonia [15, 16]. Smart card functionality touches not only ID cards but residence permits and other government-issued cards as well [10].

There are no variable costs to allow a person to log in to websites with their smart card. For this authentication method, no per-transaction fees exist, as the certificate validity service (OCSP) [17] can be queried for free.

An end user's computer can extract an authentication certificate from their smart card with the help of special software, generally distributed by the government agency responsible for maintaining the cards [15]. This certificate, once on the computer, can be sent to the private company's authorization server with Client Certificate TLS option [18] natively or with the use of specialized helper library [19], using standard REST calls.

Qualified trust service provider for Qualified Certificates for e-signatures installs the certificates in ID-cards [20], which ensures a high degree of certainty about the identity of the person authenticating.

A significant advantage of using a decentralized eID infrastructure, such as ID card authentication, is that there is no intermediary service in the process, allowing companies to skip going into expensive contracts with an eID service provider.

2.3.3 Mobile-ID

Five years after SK ID Solutions introduced ID cards for use in Estonia, they have developed a mobile phone-friendly way to access the users' eID for use in Estonia and Lithuania [21]. SK achieved it by extending the functionality of phone SIM cards by adding new applications on the microchip.

The price of using Mobile-ID for the service provider varies based on usage, starting from 10 euro per month (10ct per request) to costing over 5 000 euro, where the effective cost is under 1ct for request [22]. For the end-user, mobile operators can charge an additional fee for the use of this service [23].

Accepting Mobile-ID would allow companies to access the markets of two countries: Estonia, and Lithuania, as the technical implementation is identical.

Qualified trust service provider for Qualified Certificates for e-signatures installs the certificates in a particular variety of SIM cards, capable of supporting Mobile-ID [20], which ensures a high degree of certainty about the identity of person authenticating.

2.3.4 Smart-ID

Smart-ID is the latest and fastest-growing way of accessing citizens' eID, working in all 3 of the Baltic States [24]. The protocol utilizes mobile phones as authentication, similar to Mobile-ID. Unlike Mobile-ID, it does not require specialized external hardware [25]. The authentication process is handled by combining the eID server and the end user's smartphone. Despite that, it still passed the eIDAS compliance audit for the requirement of ensuring signature private key is "with a high level of confidence under sole control" of its owner [26]. After passing the audit, Smart-ID was recognized as a QSCD, allowing it to create QES in 2018 [27].

The price of using Smart-ID for service providers, much like Mobile-ID, varies based on usage, starting from 50 euros per month (10ct per request) to over 20 000 euros, where the effective cost is under 1ct for request, based on the total amount of transactions performed within a month [28]. For users, unlike Mobile-ID [23], there are no telecommunication operators involved, and there are no additional costs.

Implementation of Smart-ID would allow users to access the markets of three countries: Estonia, Latvia, and Lithuania.

Qualified trust service provider for Qualified Certificates for e-signatures users their data centers to hold part of the private key and certificate used to authenticate users [20], which ensures a high degree of certainty about the identity of person authenticating.

2.3.5 TARA

TARA is Estonia's primary gateway for authentication to public services [29]. TARA provides the ability for users to sign in with any of the three primary eID methods of

Estonia and with the eID schemes of other EU member states. The ability to authenticate with the systems of other countries is of particular interest, as it also doubles up as the official eIDAS node of Estonia [29].

Estonian Information System Authority intends to limit the use of TARA to public services only [30].

Technical implementation for the consumer, unlike Mobile-ID and Smart-ID, will be much easier to implement, as it uses the well-adopted protocol of OpenID Connect [31, 32].

It is worth mentioning while the underlying authentication methods have received proper eIDAS auditing and are backed by a qualified trust service, this and all of the following authentication methods have not been audited in compliance with eIDAS, or the auditing was not made public.

TARA only acts as an authentication service. It would not be able to provide means of signing documents [31]. If the business is considering expanding to allow for online digital signing, an infrastructure like TARA will unlikely be a great choice.

2.3.6 eeID

Estonian Internet Foundation created eeID service for the exclusive purpose of bringing eID authentication to the private sector [33]. It is a clone of TARA without it being Estonia's gateway for the eIDAS node network. The similarities mean that all points outlined to TARA apply to this service too.

The service is new, does not have pricing tiers, and currently asks for 9ct per successful authentication request [34].

The vision of the said service is to allow users to access the markets of all EU countries. Currently, there are only fourteen countries with notified eID authentication methods [10]: Estonia, Germany, Italy, Spain, Belgium, Luxembourg, Croatia, Portugal, Latvia, Lithuania, Netherlands, Czech Republic, Slovakia, and Denmark.

2.3.7 HarID

Estonian Ministry of Education and Research created this service for the youth of Estonia to access different educational institutions across Estonia [35]. ID cards are only legally required to be held by citizens over the age of 15 [4], so everyone under would have been unable to access their school system. HarID accepts TARA authentication methods with the addition of username & password. This authentication method is held exclusively for the education sector and will be skipped over in this thesis.

2.3.8 Dokobit

In the initial list of services using eID in Estonia [11], one service stands out - Dokobit [36]. They provide services comparable to eeID in that they aggregate different eID methods of Estonia (ID-card, Mobile-ID, and Smart-ID) and other countries. The primary difference between the two authentication providers is how they want to achieve their multi-national implementation goal. Dokobit relies on integrating each country's system individually. In contrast, eeID depends on using the framework of the eIDAS infrastructure [33].

Pricing for Dokobit varies drastically, and the provided prices for the Baltic States [37] start at 50 euros per month (7.1ct per request), going down to 4.2ct per request at 500 euros per month.

Dokobit supports 11 countries: Estonia, Italy, Spain, Belgium, Latvia, Lithuania, Finland, Norway, Iceland, Poland, and Portugal [36].

UAB Dokobit is a trust service provider for Qualified validation of qualified e-signature. It means the service itself does not provide Digital Signature certificates, but eIDAS considers the results of validation of signatures trustworthy [20].

2.4 Authentication and QSCD

The requirements for Estonian ID cards make a clear distinction between "ADF AWP" and "ADF QSCD" applications. Both software applications are loaded onto the smart card; however, only the QSCD application, guarded by PIN2, can create QES. Implication here is that for authentication with an ID card, a QSCD is not used.

The fact that smart cards do not use a QSCD for authentication does not mean that the certificate used was not audited under eIDAS. In practice, the same QTSP CA issues both the authentication and the QSCD certificates (see figure 1). "Since the authentication certificate is issued under the same CA and the same Terms and Conditions, the certificate policy documents and other CA statements are applicable to both certificates" [38]. While eIDAS does not outline any requirements for authentication modules in the QSCDs, it is reasonable to assume that the QTSP took the exact organizational and technical measures to protect the authentication key on the same physical device.

For different eID authentication schemes, companies have to assess the scheme's security on a case-by-case basis. For all of the methods analyzed in the thesis (Lithuanian and Estonian ID cards, Mobile-ID, and Smart-ID), the same QTSP signed both certificates - for authentication and signing. There seems to be no discernable benefit to issuing an authentication certificate with a different CA, so if a device offers both authentication and qualified signature creation functionality, it is highly likely that the authentication certificate is as trustworthy as the eIDAS compliant QSCD.

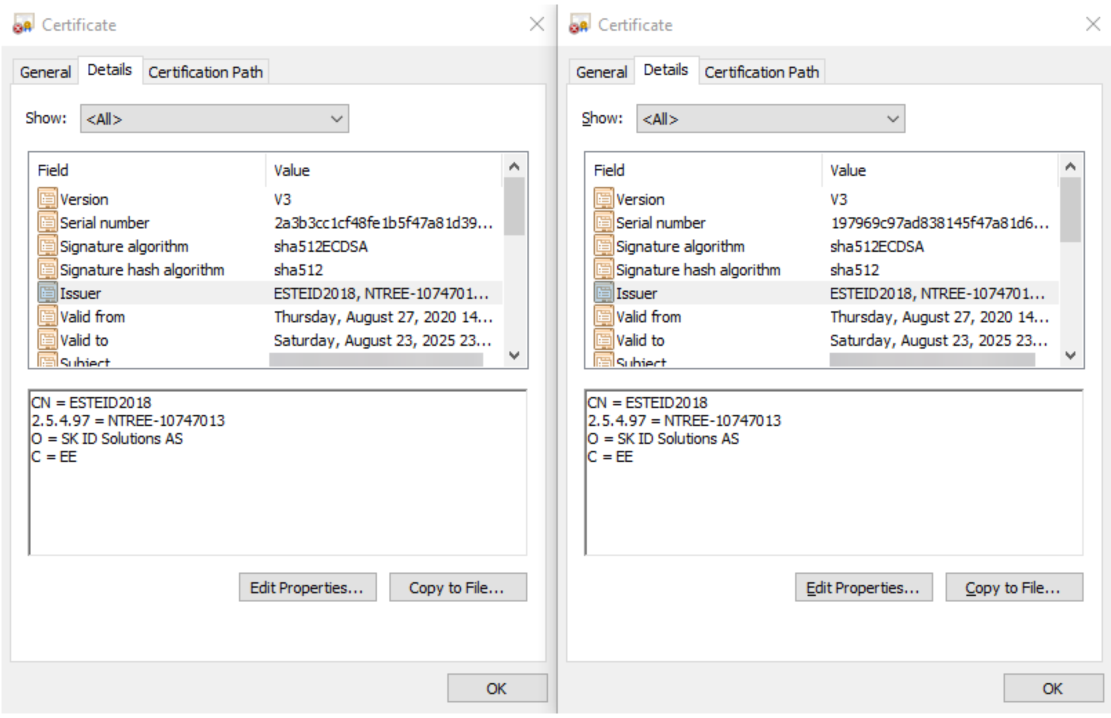

Figure 1. Comparison of authentication (left) and signing (right) certificate issuer field

Special case: third-party providers When using an intermediary such as TARA (eeID) or Dokobit, the new eID provider acts as a new trust anchor. All rigorous eIDAS and QTSP auditing become irrelevant when using intermediary services, as systems are as secure as their weakest links. Companies must first look at the security auditing and the certifications of the intermediary; only then consider the security of authentication options its delegates.

2.5 Levels of assurance

The eIDAS regulation uses three assurance levels: low, substantial, and high. These levels refer to the degree of confidence about the claimed or asserted identity of a person [3]. These assurance levels map to levels 2-4 of ISO 29115 [39].

Finish this section

2.6 GDPR

When dealing with eID, sensitive personal data processing is required. Two years after the EU enacted the eIDAS legislation, the parliament issued new legislation to consolidate all previous privacy laws [40].

In GDPR, companies must be aware of key terminology. Interested parties can find a complete list of definitions in article 4 of the regulation. Below you can see a list of the most critical keywords in the context of eID authentication.

Personal data The eID solutions in scope all provide the following personal information: full name, serial number (national id), and the country of origin.

Processing The minimal amount required to process information is to store uniquely identifying information that would link an eID to an internal id code. At the very least, this action involves collecting, storing, and retrieving personal data.

Processor and Controller In the minimal case, no third parties are involved, and the controller, the processor, and recipient are the same entity.

2.7 Threats

There has to be a source with the extensive list, all protocols I checked look the same, but I can't seem to find a post saying yes, you need to do this.

In the sequence diagram (see figure 2), we see a high-level overview of any eID authentication solution. In the authentication protocol, the relying party (company implementing eID sign-in) entirely depends on the QSCD interface, which acts as a trust anchor. A communication channel exists between the relying party and the interface. We can categorize the channel type into two groups:

1. trusted, where the sender can reasonably guarantee that adversaries are unable to change the data in transit (Smart-ID, TARA, Dokobit);

2. untrusted, where there is a reasonable possibility for data to be tampered with in transit (Web eID).

Trusted communications generally operate using an encrypted backchannel, where untrusted ones require the client to send data from the QSCD interface. Because the client sends the data, this becomes an untrusted channel, and the relying party must perform additional certificate verification.

Failure to encrypt and establish the authenticity of the interface in a trusted backchannel or not validating data in an untrusted channel allows adversaries to perform man-in-the-middle attacks.

Limitations: For the scope of the thesis, only two security aspects will be analyzed: the network communication between the natural person and the interface and the certificate authenticity guarantees. Security analysis of QSCD and its communications with the interface will not be analyzed.

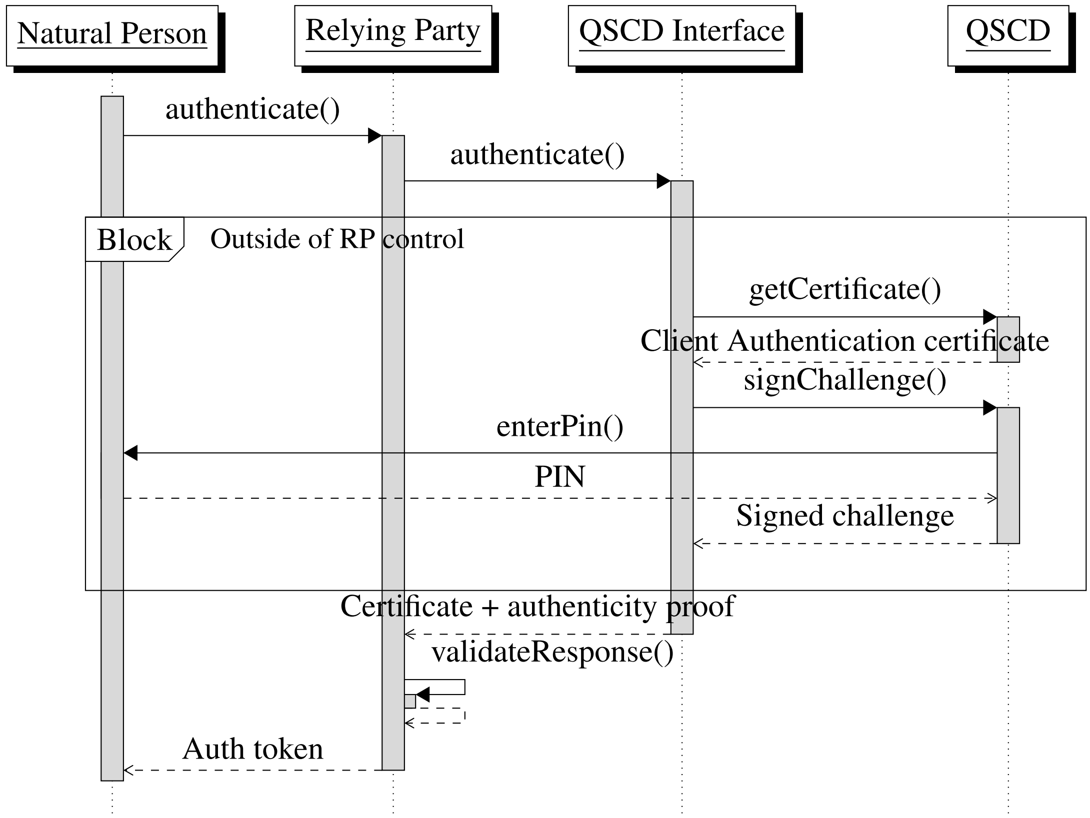

Figure 2. High-level sequence diagram of a eID authentication flow

Further research? Limitations: proper analysis of what companies would like to see in a solution for more accessible bullet points

3 Related Work

3.1 National e-ID card schemes: A European overview

In 2008, researcher Siddhartha Arora investigated different uses of eID in Europe [41].

The technical report was published at the time when the eID technology was still in its infancy, and the concept of eID was tied to it being linked to a physical ID card.

Paper references that eID cards offer three forms of information security functionality, each with an increasing level of security provisions: identification, authentication, and signature (see table 1). In this table, A is prover and B is verifier.

Table 1. Forms of information security functionality provided by eID [41, 42]

Identification (I)	A can prove to B that he is A, but someone else can not prove to B that he is A.
Authentication (A)	A can prove to B that he is A, but B can not prove to someone else that he is A.
Signature (S)	A can prove to B that he is A, but B can not prove to himself that he is A.

The idea of splitting functionality into identification, authentication, and signature can be verified today in Estonian [43], and Lithuanian [44] ID cards. In these cards, there are two certificates — one for client authentication and the second for digital signature.

These authentication and identification certificates are not encrypted, and can anyone with the correct tools can read them from the ID card. Signed documents also have a copy of the certificate attached. These certificates identify a person, but due to ease of replication, the recipient should not trust the sender's certificate because there are no guarantees that the certificate belongs to the sender.

The authentication and signing certificates require their respective keys to perform asymmetric cryptographic operations. In theory, it is possible to sign documents with the authentication certificate; however, the verification software will reject such signatures because the certificate's purposes would not include digital signature.

Another thing the paper touches on is the possibility of having multiple eID schemes. The author spotlights Austria as they wanted to have various sources of eID, not limit themselves only to one card. Having multiple authentication methods was a novel concept at the time. Many countries followed suit, and in Estonia, there are three primary sources of eID. In France, a source of eID doesn't even come from an ID card [45].

The paper's conclusion emphasizes the importance of the eID itself, not ID cards. The EU took this path when implementing the legislation for eIDAS, which allowed easier integration of infrastructure member states already had in place.

3.2 The Austrian eID ecosystem in the public cloud: How to obtain privacy while preserving practicality

This paper explores what information the Austrian government stores on the issued identity documents and what operations the documents can perform [46]. Researchers identified four types of functionality: identification and authentication of Austrian citizens, qualified electronic signature creation, encryption and decryption, data storage. This functionality seems widely adopted as it matches Estonia's ID card.

Paper presented an interesting legal issue - Austria does not allow a person identifying code (CRR number) to be "used directly in e-Government applications due to legal data protection restrictions." The solution required Austria to create SourcePIN, a framework to develop different personal identifying numbers for each service trying to access it while hiding the original code [46, 47].

A big concern authors express is that everything goes through one single source of trust, which does not scale well. If many people wanted to use the system, it would quickly become a bottleneck. Moving many essential components to the public cloud can alleviate the problem.

The paper's main contribution to this thesis is to remind that even though technological barriers are crumbling, there might still be legal obstacles to overcome. Austria is currently not part of the eIDAS node network, and it would be an excellent further research topic to investigate how Austria's eIDAS node operates.

3.3 Secure cross-cloud single sign-on (SSO) using eIDs

Researchers explore the possibility of users using an SSO system to log in via their eID instead of the traditional username/password authentication method [48]. As means of doing so, they explore the capabilities of the STORK framework and other frameworks seen in previously mentioned related literature. The STORK framework is the predecessor to eIDAS [49].

The idea of the STORK framework is that any EU citizen should be able to use their eID issued by their home country to authenticate with services in other countries. An example of an activity would be opening a bank with an Italian ID card. The paper suggests extending the framework to support federation so private business identity providers can use the security options provided by eIDs and not store weak passwords.

The paper shows a proof of concept prototype usage for bringing STORK to support SSO. Emphasis was given on the backward compatibility, not to require any breaking changes to an existing STORK protocol.

Researchers found that one SAML protocol, however similar they may be, is not compatible with one another. The consumer company wishing to implement the proposed protocol must develop an adapter application to integrate different identity providers, such as STORK, Facebook, and Google. Before a protocol sees widespread mainstream adoption, facades will be required.

3.4 EID @ Cloud: integración de la identificación electrónica en plataformas europeas en la nube de acuerdo con el reglamento eIDAS.

This paper talks about integrating a new eIDAS node with the private sector in mind [50]. The eID@Cloud research initiative has proven it possible to allow private citizens to integrate this system to authenticate persons. Possible does not mean ready for use and outlines some issues that need addressing.

Even though the eIDAS node infrastructure brings apparent benefits to the citizens, the public, private entities, and the service vendors, there are still caveats that slow the final integration of the EU digital identity platform. The project eiD@cloud shines light upon these barriers:

1. There are still some differences between the national schemes and the integrations of said national schemes in a unique and interoperable net that must be the eIDAS in the context of the EU.
2. The deployment of each eIDAS node of each member state happens at different speeds, creating mistakes and a lack of availability to complete the eIDAS project.

The interoperability testing consisted of accessing each partner's cloud platforms to verify the identities that belonged to the citizens of the other partners' countries. Norway's eIDAS node did not work with other countries' eID - the protocol executes correctly, but the user incorrectly received an error message asking for Norway identification. It shows that some parts of the system were not stable at the research time, but the whole infrastructure continued to run.

The eID@Cloud was a great project testing the implementation and readiness for public and private sectors, which provided excellent feedback for the EU Commission. The most important finding is that it found a way for private entities to connect to the mesh.

3.5 LEPS - Leveraging eID in the private sector

This final research [51] was performed at a similar time to the eID@cloud [50], but in different countries. LEPS researchers have implemented an eIDAS node for private customers. However, they also provided market analysis.

The market analysis targeted four main categories of e-service providers:

1. Organizations that need or want to migrate from the existing identity and access management (IAM) solution. This could apply to organizations that have scaled out their internal or tailor-made IAM solutions or organizations that already use partially external or third-party e-identification or authentication services but are looking for a higher level of assurance (LoA).

2. Organizations that use low assurance third-party eID providers such as a social login want to elevate the overall level of security and decrease identity theft and fraud by integrating eIDAS eID services.

3. Organizations that are already acting or could be acting as eID brokers.

4. Organizations that want to open new service delivery channels through mobile phone and are interested in mobile ID solutions that work across borders.

In the case of the thesis, the targeted e-service providers are of the first category - organizations wishing to improve IAM solutions to include a higher level of assurance.

The researchers recommend using an approach like LEPS to integrate eID authentication rather than creating an eIDAS node. The primary reason for avoiding node creation would be the cost-effectiveness of implementation, as these adopters "are unlikely to have the know-how, resources, and capacity to implement eIDAS connectivity." "Many organizations do not have resources for eID service implementation and operation internally was already exploited by social networks." The targeted benefit is the "easy way to integrate highly scalable, yet low assurance, eID services."

LEPS is a similar service to Estonia's eeID.

3.6 Federated Identity Architecture of the European eID System

The authors of this paper describe the current situation in the identity management landscape [52]. The researchers provide all the necessary background information to understand the implementation details of any eID authentication system design.

Does this belong in related work? This paper is de-facto required background

3.6.1 Authentication methods

The first significant contribution relates to explaining different ways of authenticating persons.

Any authentication method is based on something user knows (password, pin code, answer to security question), is (biometrics - eyes, fingerprints), or has (physical device - key card, USB device) [53]. Any other method would leave the person without agency over the authentication process.

An emphasis is put on the importance of mixing and matching these authentication schemes to increase the system's security.

3.6.2 Authentication Paradigms and Models

The second helpful point of the paper is the description of different identity management paradigms and models [54]. Paradigms refer to the implementation and deployment, whereas models refer to the data storage and roles.

The three main paradigms as network, service, or user-centric. The network-centric approach gathers all identities into one place, usually known as a "Domain Controller." The service-centric method would create a new identity for each service, leading to high duplication. The user-centric paradigm makes the user prove their own identity. Europe's eID solution does not favor any of the paradigms allowing identity providers to innovate [4, 33, 36].

There are also three authentication models: isolated, centralized, and federated. Unlike paradigms, where any of them is fair game, Europe's identity providers can use only the federated one. The isolated model requires all services to hold a copy of every identity in the EU. The centralized model is suitable for having a central place for looking up identity in a country, and it is an excellent solution for high-profile agencies. The federated system has the advantage of scaling well horizontally and not requiring to keep an index of all citizens it would like to serve, which is a tremendous advantage in the area of GDPR.

3.6.3 Authentication protocols and services

Researchers have allocated a good portion of the paper to provide an overview of potential protocols and implementations. The list is massive and in-depth; however, it becomes clear that SAML [55], OAuth2.0 [56], and OpenID Connect [32] protocols are by far the most popular protocols to choose for implementation. The engineers behind the eIDAS network implementation decided to settle on the SAML protocol.

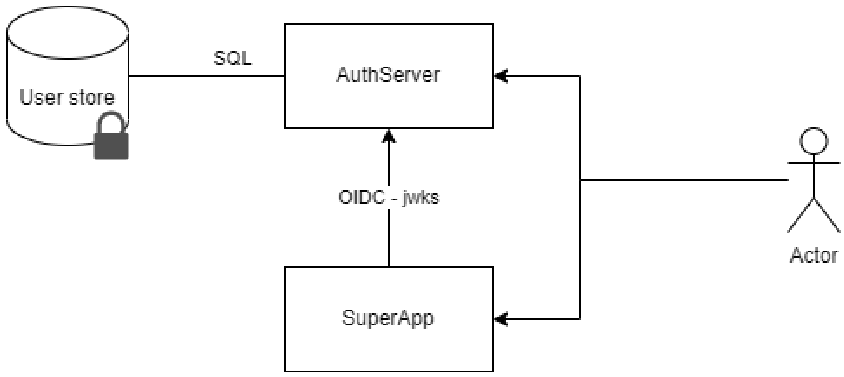

Figure 3. Initial system architecture

4 Research

4.1 System architecture

4.1.1 System overview

Initial state In figure 3 we see a high-level overview of a system we are trying to integrate eID authentication. This system consists of the following components:

1. AuthServer - the company's SSO; acts as a central authority for identity. Issued OIDC id tokens, which contain user ID, their roles, and claims.

2. SuperApp - a resource server with access control enabled. It uses id tokens issued by AuthServer and verifies them using asymmetric cryptography.

3. User store - a data store containing user login information - usernames, password hashes, other PII.

4. Actor - a physical person accessing the resources in the system.

Desired state The company wishes to implement eID authentication. Since the authentication is not done locally but is delegated to some remote service or device, the protocol can be treated as an external federated sign-in. Frameworks such as ASP.NET Identity have special helpful tools to handle external identity providers.

With the inclusion of an external eID provider, we can see the new system architecture in figure 4.

We can see two significant additions:

1. eID provider - a gateway to obtain someone's eID. It can be any eID source like Dokobit, TARA, Smart-ID, ID card.

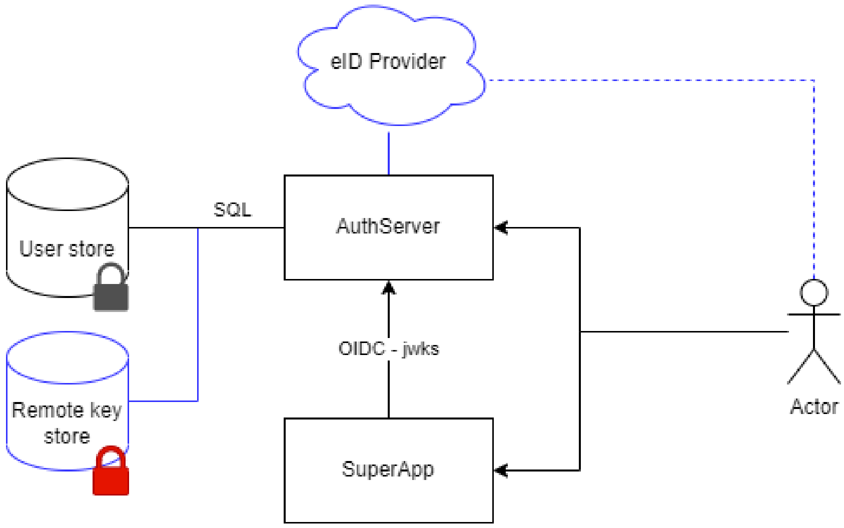

Figure 4. System architecture after the inclusion of an eID provider

2. Remote key store - it is a storage for unique identifiers provided by the eID provider.

The primary purpose of the remote key store is to link the user ID used in the internal system with the unique identifier provided by the eID provider. Because the unique identifier can change or the same physical person can have multiple eIDs [57], it is required to allow numerous eIDs to map to a single internal ID.

The key store is represented with the addition of a red lock. Depending on the country, the data stored there may be subject to strict privacy regulations. Companies should consider implementing strict access control for this part of the infrastructure.

Final state The end goal for the scope of this thesis is to implement three eID providers into the architecture. For normal companies, it would make sense to implement multiple in case they would like to get more coverage. Additionally, they could register non eID providers, such as Google or Microsoft social logins. The final high-level overview of the system can be seen in figure 5.

4.1.2 Process overview

For validation of the architecture, we will consider two use cases.

The first one (see figure 6) is concerned about accessing a protected resource with a token issued by the AuthServer. This use case validates the base state of the system.

The second use case (see figure 7) is also concerned about accessing a protected resource, but only those, who authenticate with a higher level of assurance, like eID, can

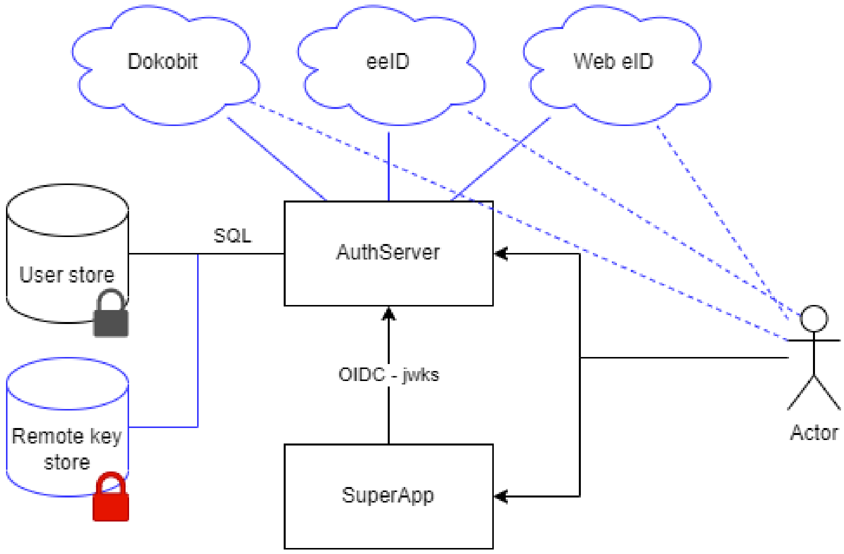

Figure 5. System architecture after the inclusion of all eID providers in scope

access it. This use case validates the successful implementation of eID authentication and access control.

4.1.3 Linking eID to an internal user ID

There will be a need to uniquely link an identity to an internal account in the company SSO. The security requirement, in this case, is not to allow other users to access the same account. For this goal, companies must use one or more person-identifying properties.

When using a passport as a reference, it has the following identifiers: (issuer) country code, document number, surname, given name, personal code, citizenship, date of birth, date of issue, date of expiry, and authority. In these cases, it is easy to use the personal code for identifying a person as it is unlikely to change - people change names, documents expire; authorities and date of birth do not narrow it down nearly enough.

Implementers will hit a roadblock when checking a passport of Ireland - additionally, it has a place of birth but, more importantly, no personal identification code. In this case, the next best unique identifier would be to use the document number and update the account with a new number when the document eventually expires and is replaced.

In the world of digital identity, the eIDAS node network must provide a unique identifier for all requests [57]. Having a standardized way of obtaining an identifier is good news. All countries who wish to connect to the eIDAS network would have to expose some code to identify a person uniquely, removing the burden from the software architects to analyze what identifier they should use.

In the eIDAS node network, the unique identifier remains "unchanged for the lifetime

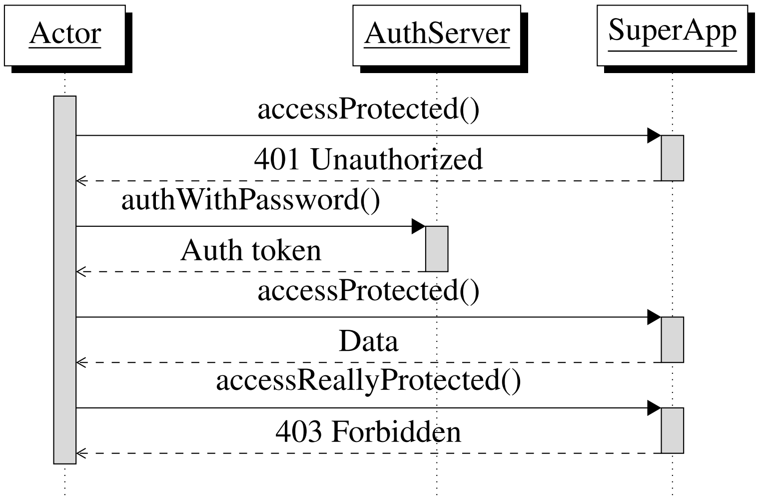

Figure 6. System behavior when authenticated with a username + password scheme

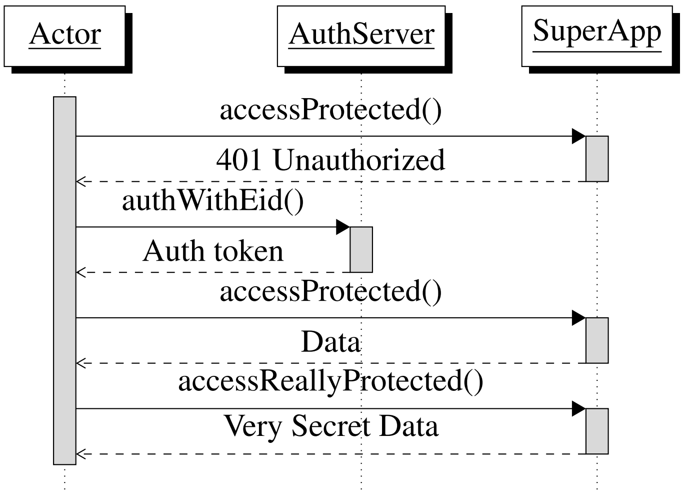

Figure 7. System behavior when authenticated with an eID scheme

of the account" [57]. Unfortunately, it does not mean that identifiers cannot change; the account associated with that identifier cannot change. If an identifier were to change when "the user's digital identity is replaced or repaired," relying parties should treat the newly obtained identifier as a completely new identity.

eIDAS Unique Identifier Structure In eIDAS SAML Attribute Profile, an identifier code is defined as:

1. The first part is the Nationality Code of the identifier. This is one of the ISO 3166-1 alpha-2 codes, followed by a slash ("/")).

2. The second part is the Nationality Code of the destination country or international organization. This is one of the ISO 3166-1 alpha-2 codes, followed by a slash ("/").

3. The third part is a combination of readable characters. This uniquely identifies the identity asserted in the country of origin but does not necessarily reveal any discernible correspondence with the subject's actual identifier (for example, username, fiscal number etc).

Example: ES/AT/02635542Y (Spanish eIDNumber for an Austrian SP).

Summary Using eIDAS Unique Identifier structure as a base, we can see that it is enough to uniquely identify a digital identity in eIDAS with the country of origin, country of destination, and a set of characters to identify that person in the origin country. The destination country will always be the same in our company's case. To uniquely identify a person, we will only need its origin country and a unique identifier a member state must provide.

For this thesis, identifiers will be marked as "{ISO 3166-1 alpha-2}/{Code provided by country}". Unique identifier examples: EE/38001085718, LT/49003111045, SE/870314-2391.

4.1.4 Privacy Policy

The company wishing to implement eID authentication will have to deal with personal information as described by GDPR [40]. Before going live with an eID solution, companies must first consult a legal professional for advice.

A privacy policy is a legal document and is way outside of the scope of a technical implementation thesis. However, it is still important to understand the basics. For this goal, two privacy policies will be analyzed: Web eID [58] and Dokobit [59].

From the cursory analysis of the two policies, there are three fundamental aspects our company needs to address: what data is processed, with whom the company shares the data, and what is the retention policy.

Based on the privacy policies of Web eID and Dokobit, we constructed a rudimentary privacy policy for the use of the test application environment. A copy of the text can be found in the thesis' appendices.

Further research can be done to outline better the GDPR requirements needed to process a person's eID.

4.2 Weaknesses in the architecture

All systems come with points of compromise. It is best to be aware of these weak points and harden against them early on rather than after a cyber incident. The weaknesses can be grouped into two parts: eID provider dependent and local.

eID provider dependant weaknesses These threats come from one common issue: trusting an eID service provider when a relying party should not have. There are four primary causes:

1. relying party trusts an input from an insecure channel;
2. relying party does not perform the necessary validation;
3. weakness in the architecture of transport protocol;
4. eID provider itself gets compromised;

These points will be addressed on a case-by-case basis for each of the eID providers in the study.

Local weaknesses In figure 4, four components are not linked to a particular eID: AuthServer, SuperApp, application store, user store, and remote key store. We can identify weaknesses for those parts here. Each weakness was analyzed through the lens of CIA security analysis.

4.2.1 SuperApp

These two assets have similar weaknesses, with the only difference being the way users authenticate themselves. More in-depth analysis on authentication for the AuthServer can be seen on the

[CI] Users can see and or edit data normally forbidden to access This issue is usually created by one of two causes:

1. access control measures are disabled or do not protect resources;
2. OIDC token validation is disabled or misconfigured;

The first cause has a trivial fix - developers have likely forgotten to enable the data access guard on the endpoint. It is unlikely that all authorization rules break.

In the case of the second cause, some corners were likely cut in the ID token validation process. When validating a token, the process has to match the one described in the OIDC spec [32] exactly. This process consists of three major parts:

1. check if the token's crypto algorithm is as expected;
2. validate token signature;
3. validate claims - issuer, audience, timestamp, nonce;

Developers usually need not worry about the process, as most frameworks have adopted the OpenID Connect protocol or have well-maintained libraries.

[A] Service becomes unavailable This threat is caused by one of the following:

1. server is offline or overloaded;
2. OIDC token validation is misconfigured to have an incorrect authority;

The first case is a common availability issue, meaning the server is suffering a denial of service attack. We will not cover the mitigation of this form of availability threat in the scope of the thesis.

A likely cause for this issue is the manually configured OIDC properties on the relying party. If possible, developers should never configure the properties manually and use the well-known metadata endpoint instead. A metadata endpoint usually looks like this - `https://auth.mycompany.org/.well-known/openid-configuration`.

4.2.2 AuthServer

All of the points that apply to SuperApp also apply to AuthServer. However, there is a critical use case that is worth mentioning explicitly.

[CI] Users can see used eID schemes and add new ones with unsafe sign-in As per requirements, users must assign multiple identity providers to their accounts. When adding a new external scheme, the currently signed-in user must have the same privilege level as the scheme they are trying to add. This countermeasure is for the event an adversary gains access to their account; they wouldn't be able to elevate their privileges by adding their eID scheme and signing in with it.

With this requirement in place, registration with an email and password becomes less valuable, as those accounts could never add an eID afterward. An exception to this rule can occur if a user has no eID schemes registered.

This problem is solved by having a company policy to verify the user's first added eID manually. Without this verification, an adversary can register a new account with their real eID and access the data anyway.

4.2.3 Data stores

All three data stores have the exact issues between them. For the sake of brevity, they will be grouped under the umbrella of *data store*.

[CI] Users have a less secure way of accessing the data store The system is as secure as its weakest link. If users have direct access to the database while bypassing the eID authentication check, the security and assurance guarantees are worthless.

If there are alternative ways of accessing the database, companies must implement proper access rules that would be as secure as the one's eIDs provide. How that can be achieved depends on what data storage option is being used. The best option would be to close down external access to the data stores completely.

[CI] Man-in-the-middle attacks While uncommon, some data stores are vulnerable to MitM attacks [60]. Although the best course of action would be to move the data storage server away from the general internet and make it accessible only on the local network or VPN, that is not always enough. For maximum security, companies must implement the recommended MitM prevention techniques [61].

[A] Data is destroyed or lost Ransomware attacks and accidental database corruptions can happen, so offline remote backups are a must.

One must not forget to satisfy the requirement of not allowing for easier access to data even here. These backups must be protected from unauthorized access.

One approach to solving this would be to use encryption; keys should only be available after authenticating with an eID scheme. One equally secure alternative would be to use something like the encryption and decryption functionality of Estonia's ID cards.

4.3 Case Study: Dokobit

4.4 Case Study: eeID

4.5 Case Study: Web eID

4.5.1 About

Released in the Summer of 2021 [19] and having undergone significant changes in January of 2022, this eID framework allows users to authenticate and sign documents using their smart cards.

Functionally this framework is split into three parts: software the user needs to install on their computer, a javascript library that acts as a data intermediary, and the certificate validation library for the back-end.

The software users need to install is similar to the one various countries' governments issue. The significant difference is that this software supports more than one countries' eID solutions. Supported countries include Estonia, Latvia, Lithuania, and Finland [19].

4.5.2 Data Flow

Figure 8 displays the high-level overview of the complete flow of data within the Web eID framework. A detailed explanation of the steps can be found on the technical specification page [62]. Companies implementing the framework should only consider the browser and the server application (steps 1-3 and 13-17).

4.5.3 Security Analysis

Researcher Arnis Paršovs published a security analysis of the protocol v1 in October of 2021 [63]. Developers behind the Web eID framework acknowledged the weaknesses and addressed them in v2 [62], which will be used in the scope of the thesis. At the time of writing, independent researchers and auditors have not yet performed security analysis for this version.

Actors The actors in the figure 2 assume the roles of: QSCD Interface - web-eid.js [64], web-eid-app [65]; QSCD - Smart cards of Estonia, Latvia, Lithuania, and Finland [19].

Even though distributed by the same website, id.ee, this interface is separate from the official id.ee software Estonian citizens use to sign and verify documents. "In the future, the final version of Web eID will be added into the ID-software installation package, available for the users the website on www.id.ee" [19]. Owners of other countries' smart cards will still have to download the special software from id.ee.

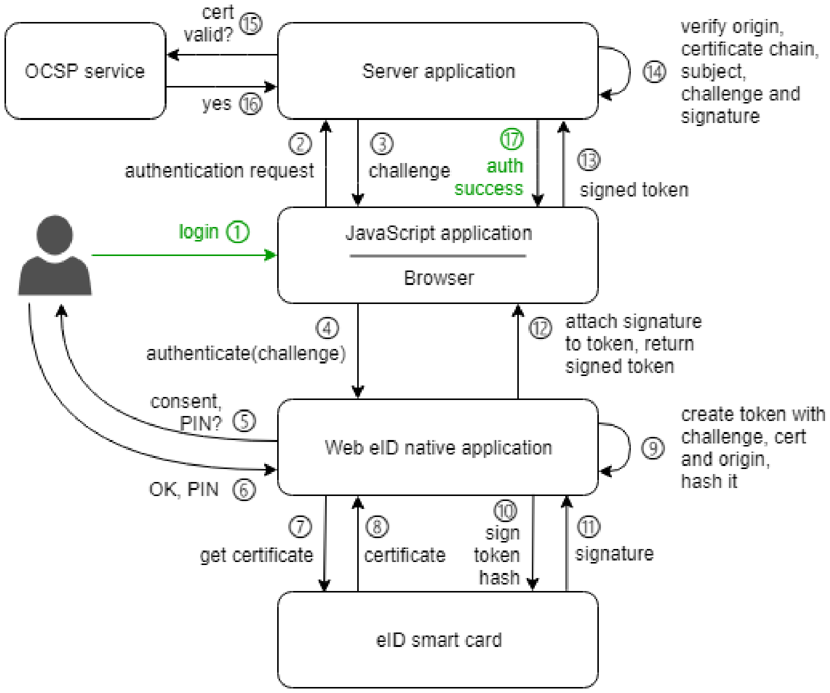

Figure 8. Web eID Authentication flow [62]

Threat protection The Web eID framework uses an insecure channel for communications, so developers must take caution and verify received data when implementing the framework.

Unlike in the cases of Dokobit and eeID, the risk of impersonation is not transferred to the eID service provider.

Discussion: suggest how the certificates are way too challenging to obtain for a casual company and may lead to additional vulnerabilities

Discussion: phishing attacks if the company establishes policy to add certificates if they cannot sign in with their card

4.5.4 Trust Anchor

Unlike Dokobit and eeID, Web eID does not provide any guarantees about the trustworthiness of a certificate. It is, however, not out of malice and reminds developers by sending the certificate in a field called "unverifiedCertificate" [66].

The relying party must verify the certificate and challenge themselves by checking the origin, certificate expiry, trust chain, OCSP response, and the challenge. This validation structure makes the trust anchor technological and highly dependent on the implementation correctness by the developers.

Discuss how we can never trust developers without proper supervision

4.5.5 Pricing

The Web eID authentication service is free of charge, as the only external validation, OCSP [17] requests are free to use. When creating digital signatures, the timestamping service may require payment [66].

4.5.6 Implementation

For each protocol implementation step, developers will have to fulfill certain guarantees before the system goes into production.

Steps 1-3 Building the challenge nonce. The goal of these steps is to create the challenge the user will have to sign with their private key. There are a couple of guarantees the application must provide:

1. Generated challenge nonce must be between 32 and 96 bytes (inclusive) in length [67];

2. "It must be guaranteed that the authentication token is received from the same browser to which the corresponding challenge nonce was issued" [66]. The framework creators suggest attaching it to the user session.

3. "Cache must be used for protection against replay attacks by guaranteeing that each authentication token can be used exactly once" [66].

4. "Cookie-based authentication must be protected against cross-site request forgery (CSRF) attacks and extra measures must be taken to secure the cookies by serving them only over HTTPS and setting the HttpOnly, Secure and SameSite attributes" [66].

In the implementation example, these measures were addressed by:

1. a 64 byte cryptographically secure randomly generated nonce is created (see listing 1);

2. challenge nonce is set in the user's session, which adversaries cannot tamper;

3. the generated nonce is stored into local memory cache for later use; nonce expires after 5 minutes;

4. an input field is rendered on the page with a unique CSRF validation token, which prevents cross-site request forgery attacks (see listing 2);

```
private TimeSpan ChallengeLifetime { get; } = TimeSpan.FromMinutes(5)
    ;

private readonly IMemoryCache _cache; // Injected

[HttpGet("challenge")]
public IActionResult GetChallenge()
{
    var nonce = RandomNumberGenerator.GetBytes(64);

    _cache.Set(Convert.ToBase64String(nonce), true, ChallengeLifetime
        );
    HttpContext.Session.Set("eid.challenge", nonce);

    return Ok(new { nonce });
}
```

Listing 1. Web eID Challenge Endpoint

```
@inject Microsoft.AspNetCore.Antiforgery.IAntiforgery _csrf
@{ var csrfToken = _csrf.GetAndStoreTokens(HttpContext); }

<!  Button used to sign in  >
<a role="button" class="btn btn secondary" id="webeid auth button">
   Web eID</a>

<input id="csrfToken" type="hidden" value="@csrfToken.RequestToken"/>

<script>
    ...

    const authTokenResponse = await fetch("/signin id/login", {
        method: "POST",
        headers: {
            "Content Type": "application/json",
            "RequestVerificationToken": document.getElementById("
                csrfToken").value
        },
        body: JSON.stringify(...)
    });

    ...
</script>
```

Listing 2. Web eID UI excerpt

Steps 13-17 After the user signs the nonce challenge and sends their certificate, the server must verify its authenticity. The application must perform all of the following before allowing the user to sign in:

1. verify the CSRF token from earlier steps [66];
2. verify the challenge nonce came from the original user and has not expired, was not consumed;
3. verify the certificate validity and check if nonce was signed by the associated private key (see below);
4. issue an authentication token with the fields from the certificate's subject;

In the implementation example, these measures were addressed by:

1. the back end endpoint for login is decorated with ValidateAntiForgeryToken Attribute. This attribute instructs the ASP.NET API to ignore requests not containing a CSRF token [68]. A JavaScript application can only access the protected endpoints by providing RequestVerificationToken header (see listing 2);
2. the application tries to extract the challenge nonce from the browsing session. The process would succeed if the session cookie were not modified. After the extraction, the application checks the nonce cache to verify if the challenge is still active. Cache hit means the nonce has not expired, and no previous authentication attempt was performed. Remove the challenge nonce from all stores.
3. The API calls a standalone validation service to verify the nonce and certificate (see below).
4. Application populates the ASP.NET identity management system with the fields from the certificate: serial number, given name, surname, country. An identity session cookie is sent to the client.

```
[HttpPost("login")]
[ValidateAntiForgeryToken]
public async Task<IActionResult> Login([FromBody]
    WebIdAuthTokenResponse token)
{
    // Obtain the challenge from session
    if (!HttpContext.Session.TryGetValue(ChallengeNonceKey, out var
        nonce) && nonce == null)
         return Unauthorized();

    // Check if token was not used before or expired
    var challenge = Convert.ToBase64String(nonce);
    if (!_cache.TryGetValue(challenge, out _))
         return Unauthorized();
```

```
        _cache.Remove(challenge);
        HttpContext.Session.Remove(ChallengeNonceKey);

        // Validate the certificate and signed challenge
        var validationResult = await _webEidValidationService.GetResult(
            new WebEidValidationRequest(token, nonce));
        if (!validationResult.Success)
             return Forbid();

        // Certificate is valid. Sign in the user

        await HttpContext.SignInAsync(BuildUser(new X509Certificate2(
            Convert.FromBase64String(token.UnverifiedCertificate)).Subject
            ));

        return Ok();
}
```

Listing 3. Web eID Login Endpoint

Certificate and nonce verification This step is the most complicated in the entire validation process. To prevent any issues with incorrect implementation, the framework maintainers recommend using their library for validation [66]. Libraries can come with security vulnerabilities, and developers are reluctant to update their used version; however, it is still more favorable to creating vulnerabilities from misconfiguration [69].

The eu.webeid.security Java package performs most of the certificate validation: expiry, purpose, policy, OCSP [66]. Developers will only have to configure the CA and host validation. Configuration is handled by providing a set of trusted CA certificates for trust chain verification and the hostname for challenge nonces (see listing 4).

This library does favor EE cards a lot more, even having hardcoded some Mobile ID policies. Not sure if it is worth mentioning.

```
public class AuthTokenValidatorService {

    @Bean
    public AuthTokenValidator validator() {
        try {
            return new AuthTokenValidatorBuilder()
                .withSiteOrigin(URI.create(System.getenv("ORIGIN_URL")))
                .withTrustedCertificateAuthorities(
                    loadTrustedCACertificatesFromCerFiles())
                .build();
        } catch (JceException e) {
            throw new RuntimeException("Error building the Web eID auth
                token validator.", e);
        }
    }
```

```
  private X509Certificate[] loadTrustedCACertificatesFromCerFiles() {
    List<X509Certificate> caCertificates = new ArrayList<>();

    try {
      CertificateFactory certFactory = CertificateFactory.getInstance
          ("X.509");

      File[] files = new File("/certs").listFiles((f, n) > n.
          endsWith(".cer"));
      if (files != null) {
        for (File file : files) {
          try (InputStream stream = new FileInputStream(file)) {
            X509Certificate caCertificate = (X509Certificate)
                certFactory.generateCertificate(stream);
            caCertificates.add(caCertificate);
          }
        }
      }
    } catch (CertificateException | IOException e) {
      throw new RuntimeException("Error initializing trusted CA
          certificates.", e);
    }

    return caCertificates.toArray(new X509Certificate[0]);
  }
}
```

Listing 4. Web eID Login Endpoint

The token validation service described in listing 4 requires the maintainers to set the origin URL in the form of an environment variable and to populate the folder /certs with trusted CA certificates.

Origin URL can be obtained by checking the window.origin JavaScript variable in the page containing the sign-in button.

For the CA certificate set, the company can get an up-to-date list of trusted certificates at the EU Trust Services Dashboard [20]. The issue with this list is that it contains all trust certificates for various scopes. In our case, we should limit the search to the extent of QCert for ESig. In the case of Estonia and Lithuania, only three entities are certified to issue certificates for QSCD (see figure 9). It is in stark contrast to Spain's 31 [20]. It is possible to further narrow down to only certificate generation services for qualified certificates (CA/QC).

In the case of Estonia's single TSP, we can see that only 3 CA are currently operational (see figure 10). Unfortunately, there is no standardized way of narrowing down which certificates could be used for authentication.

An alternative way to obtain certificates would be to go to the government authority of each country responsible for the distribution of certificates. This action requires prior

Figure 9. List of EU Trust service providers of Estonia and Lithuania capable of creating qualified certificates for e-signatures

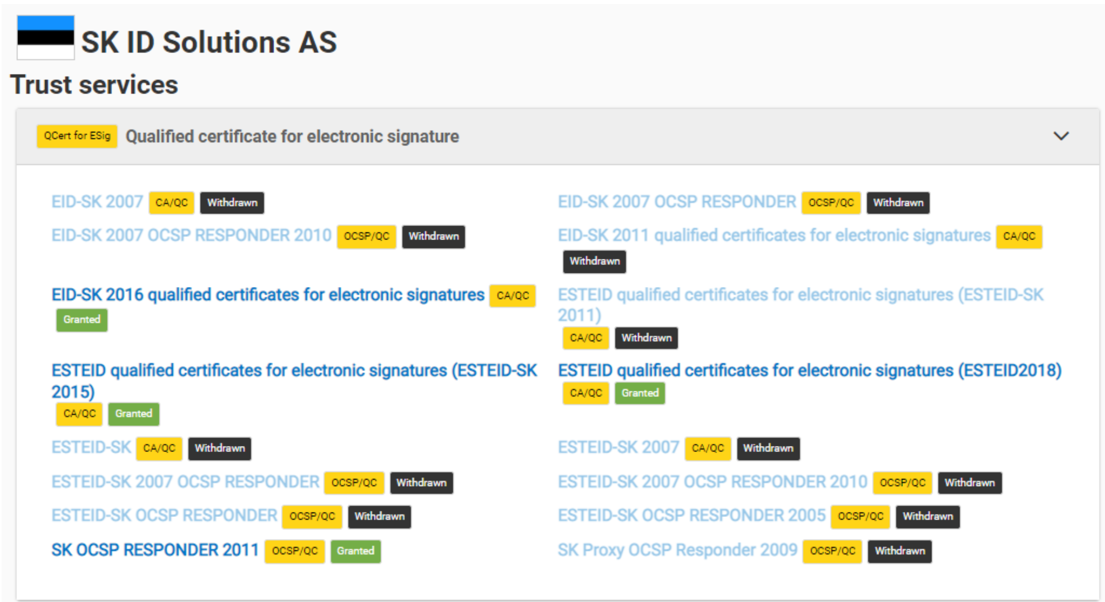

Figure 10. List of certificates issued to SK ID Solutions AS for the purposes of Qualified certificate for electronic signature

knowledge of who is responsible for issuing certificates and their purposes.

In Lithuania's case, it is the Ministry of the Interior [70] who issue two certificates (A and B) every couple of years. As of early 2022, four certificates are active, and all will be added to the trusted CA list.

In Estonia's case, SK ID Solutions manages the CA certificates [71]. Of the three certificates found on the EU Trust Services Dashboard, only two are relevant to us, the 2015 and 2018 ones, as the 2016 one has its purpose for use in Smart-ID, which the Web eID framework does not support.

The final list of certificates to support Lithuania and Estonia include four certificates from Lithuania's Ministry of the Interior and two certificates issued by SK ID Solutions for a total of six. It is essential to keep track of these certificates as each one of them can act as a point of compromise and must be monitored in the event they are revoked for security [72] or other issues.

Exposing the service With the certificate validation service configured, it is now required to link it to the Web API. If the company orients around using microservices, this service can be just that. All that the validation service requires is to expose an endpoint that accepts a nonce and a token from the javascript library and returns a validation result.

Companies must take proper measures to protect such service from adversaries as it acts as a fundamental trust anchor. Developers should take steps outlined in assume breach to mitigate the risk of misuse.

citation missing

Related research about legal person documents?

High-level overview of the system

Points of compromise?

Complete alternative, using your own issued certificates?

5 Discussion

Talk about which of the 3 case studies would work best in a company in early 2022

Should notify how eeID and Web eID are really in their infancy and further research should repeat this study in a couple of years

References

[1] Adam Ozimek. The future of remote work. *Available at SSRN 3638597*, 2020.

[2] British Council. IELTS - how to register. `https://www.ielts.org/for-test-takers/how-to-register`, 2022. Online; accessed 26-Feb-2022.

[3] THE EUROPEAN PARLIAMENT and THE COUNCIL OF THE EUROPEAN UNION. Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC. *Official Journal of the European Union*, L 257/73, 2014.

[4] Riigikogu. Identity documents act. 2021.

[5] European Comission. Digital economy and society index (desi) 2021. 2021.

[6] Răzvan Dan. Cât plătesc oamenii pentru buletinul cu cip care a început să fie eliberat deja populației. `https://stirileprotv.ro/stiri/actualitate/buletinul-cu-cip-eliberat-deja-in-cluj-napoca-oamenii-platesc-70-de-lei-pentru-`
`html`, 2021. Online; accessed 27-Feb-2022.

[7] Informacinės visuomenės plėtros komitetas. E-government gateway. `https://www.epaslaugos.lt/portal/nlogin`, 2022. Online; accessed 27-Feb-2022.

[8] Estonian Information System Authority. As of 1 march, it will no longer be possible to access certain public e-services via a bank link. `https://www.ria.ee/en/news/1-march-it-will-no-longer-be-possible-access-certain-public-e-services-bank-lin`
`html`, 2021. Online; accessed 27-Feb-2022.

[9] Jesus Carretero, Guillermo Izquierdo-Moreno, Mario Vasile-Cabezas, and Javier Garcia-Blas. Federated identity architecture of the european eid system. *IEEE Access*, 6:75302–75326, 2018.

[10] European Comission. Electronic identification schemes notified pursuant to Article 9(1) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market. *Official Journal of the European Union*, C 432/7, 2020.

[11] Applied CyberSecurity Group. Estonian e-services using eid. `https://acs.cs.ut.ee/services_using_eid`, 2021. Online; accessed 21-Nov-2021.

[12] Katri Kerem. *Internet banking in Estonia*, volume 7. PRAXIS, 2003.

[13] AS SEB Pank. Bank link and authentication service. https://www.seb.ee/en/bank-link, 2021. Online; accessed 21-Nov-2021.

[14] Arnis Paršsovs. Security analysis of internet bank authentication protocols and their implementations. Master's thesis, Tallinn University of Technology, 2012.

[15] Estonian Information System Authority. Home - id.ee. https://www.id.ee/en/, 2021. Online; accessed 20-Nov-2021.

[16] Statistics Estonia. Population figure. https://www.stat.ee/en/avasta-statistikat/valdkonnad/rahvastik/population-figure, 2021. Online; accessed 20-Nov-2021.

[17] Stefan Santesson, Michael Myers, Rich Ankney, Ambarish Malpani, Slava Galperin, and Dr. Carlisle Adams. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. RFC 6960, June 2013.

[18] Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, August 2018.

[19] Estonian Information System Authority. Web eid. https://www.id.ee/en/article/web-eid/, 2022. Online; accessed 28-Feb-2022.

[20] European Commission. Trust service providers. https://esignature.ec.europa.eu/efda/tl-browser/, 2022. Online; accessed 28-Feb-2022.

[21] SK ID Solutions. History - 2007. https://www.skidsolutions.eu/en/about/history/year-2007/, 2007. Online; accessed 21-Nov-2021.

[22] SK ID Solutions. Price list of Mobile-ID service. https://www.skidsolutions.eu/en/services/pricelist/mobile-id-service, 2021. Online; accessed 21-Nov-2021.

[23] Telia. Mobiil-ID. https://www.telia.ee/era/lisateenused/mobiil-id/, 2022. Online; accessed 28-Feb-2022.

[24] SK ID Solutions. History - 2017. https://www.skidsolutions.eu/en/about/history/year-2017/, 2017. Online; accessed 21-Nov-2021.

[25] SK ID Solutions. Smart-ID documentation. https://github.com/SK-EID/smart-id-documentation, 2021. Online; accessed 10-Feb-2022.

[26] Sylvie Lacroix, Olivier Delos, Evgenia Nikolouzou (ENISA), and Slawomir Gorniak (ENISA). Assessment of Standards related to eIDAS. Technical report, European Union Agency For Network and Information Security, 2018.

[27] SK ID Solutions. Digital signatures with Smart-ID. https://www.smart-id.com/e-service-providers/smart-id-as-a-qscd/, 2018. Online; accessed 28-Feb-2022.

[28] SK ID Solutions. Price list of Smart-ID service. https://www.skidsolutions.eu/en/services/pricelist/smart-id/, 2021. Online; accessed 21-Nov-2021.

[29] Estonian Information System Authority. The information system authority's authentication service TARA. https://www.ria.ee/en/state-information-system/eid/partners.html, 2022. Online; accessed 27-Feb-2022.

[30] Estonian Information System Authority. TARA business description. https://e-gov.github.io/TARA-Doku/Arikirjeldus, 2021. Online; accessed 21-Nov-2021.

[31] Estonian Information System Authority. TARA technical specification. https://e-gov.github.io/TARA-Doku/TechnicalSpecification, 2021. Online; accessed 21-Nov-2021.

[32] Natsuhiko Sakimura, John Bradley, Mike Jones, Breno De Medeiros, and Chuck Mortimore. Openid connect core 1.0. *The OpenID Foundation*, 2014.

[33] Estonian Internet Foundation. eeid service. https://www.internet.ee/eeid-service, 2021. Online; accessed 28-Feb-2022.

[34] Estonian Internet Foundation. Subscription agreement and terms of use of the estonian internet foundation electronic identification service. https://meedia.internet.ee/files/Pricing_eeID.pdf, 2021. Online; accessed 28-Feb-2022.

[35] Estonian Ministry of Education and Research. HarID. https://harid.ee/et/users/sign_in, 2022. Online; accessed 28-Feb-2022.

[36] Dokobit. Electronic signature API solutions. https://www.dokobit.com/solutions, 2022. Online; accessed 28-Feb-2022.

[37] Dokobit. Dokobit authentication solutions pricing. https://www.dokobit.com/docs/pricing/Dokobit-authentication-solutions-pricing.pdf, 2022. Online; accessed 28-Feb-2022.

[38] Arnis Paršovs. *Estonian Electronic Identity Card and its Security Challenges*. PhD thesis, University of Tartu, 2021.

[39] Information technology — Security techniques — Entity authentication assurance framework. Standard, International Organization for Standardization, Geneva, CH, 2013.

[40] THE EUROPEAN PARLIAMENT and THE COUNCIL OF THE EUROPEAN UNION. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC. *Official Journal of the European Union*, L 119/1, 2016.

[41] Siddhartha Arora. National e-id card schemes: A european overview. *Information Security Technical Report*, 13(2):46–53, 2008.

[42] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Conference on the theory and application of cryptographic techniques*, pages 186–194. Springer, 1986.

[43] Estonian Information System Authority. Id-card documentation. `https://www.id.ee/en/article/id-card-documentation/`, 2021. Online; accessed 01-Mar-2022.

[44] Lietuvos Respublikos vidaus reikalų ministerija. Sertifikatai įrašyti atk. `https://www.eid.lt/lt/apie-tapatybes-kortele/sertifikatai-ir-ju-valdymas`, 2020. Online; accessed 01-Mar-2022.

[45] The eIDAS Cooperation Network. Opinion no. 3/2021 of the cooperation network on the on the french eid scheme ("franceconnect+/the digital identity la poste"). `https://ec.europa.eu/digital-building-blocks/wikis/pages/viewpage.action?pageId=439222578`, 2021. Online; accessed 01-Mar-2022.

[46] Bernd Zwattendorfer and Daniel Slamanig. The austrian eid ecosystem in the public cloud: How to obtain privacy while preserving practicality. *Journal of Information Security and Applications*, 27-28:35–53, 2016. Special Issues on Security and Privacy in Cloud Computing.

[47] Herbert Leitold. eid and egovernment in austria. Graz University of Technology `https://online.tugraz.at/tug_online/voe_main2.getvolltext?pCurrPk=16600`, 2006. Online; accessed 01-Mar-2022.

[48] Bernd Zwattendorfer and Arne Tauber. Secure cross-cloud single sign-on (sso) using eids. In *2012 International Conference for Internet Technology and Secured Transactions*, pages 150–155, 2012.

[49] Ribeiro C., Leitold H., Esposito S., and Mitzam D. Stork: a real, heterogeneous, large-scale eid management system. *International Journal of Information Security*, 17(5):569 – 585, 2018.

[50] Vicente Guerola-Navarro, Raúl Francisco Oltra Badenes, Hermenegildo Gil Gómez, and Doina Stratu-Strelet. Eid@ cloud: integration of electronic identification in european platforms in the cloud in accordance with the eidas regulation. *3C Business, Researchóand critical thinking*, 8(3):64–87, 2019.

[51] Martín J.C., Aranda N.I., Carreras R.C., Pasic A., Baún J.C.P., Ksystra K., Triantafyllou N., Papadakis H., Torroglosa E., and Ortiz J. *LEPS - Leveraging eID in the private sector*. River Publishers, 2019.

[52] Jesus Carretero, Guillermo Izquierdo-Moreno, Mario Vasile-Cabezas, and Javier Garcia-Blas. Federated identity architecture of the european eid system. *IEEE Access*, 6:75302–75326, 2018.

[53] L. O'Gorman. Comparing passwords, tokens, and biometrics for user authentication. *Proceedings of the IEEE*, 91(12):2021–2040, 2003.

[54] Yuan Cao and Lin Yang. A survey of identity management technology. In *2010 IEEE International Conference on Information Theory and Information Security*, pages 287–293, 2010.

[55] Scott Cantor, Internet2 John Kemp, Nokia Rob Philpott, and E Maler. Assertions and protocols for the oasis security assertion markup language. *OASIS Standard (March 2005)*, pages 1–86, 2005.

[56] Dick Hardt. The OAuth 2.0 Authorization Framework. RFC 6749, October 2012.

[57] eIDAS eID Technical Subgroup. eidas saml attribute profile v1.2. Technical report, European Commission, 2019.

[58] Estonian Information System Authority. Data protection terms of the web eid website. `https://web-eid.eu/files/Web%20eID%20privacy%20policy.pdf`, 2021. Online; accessed 09-Mar-2022.

[59] Dokobit. Privacy policy. `https://www.dokobit.com/compliance/privacy-policy`, 2022. Online; accessed 09-Mar-2022.

[60] Adam Chester. Sql server authentication with metasploit and mitm. `https://blog.xpnsec.com/sql-server-auth-mitm/`, 2017. Online; accessed 09-Mar-2022.

[61] Microsoft Docs contributors. Enable encrypted connections to the database engine. `https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/enable-encrypted-connections-to-the-database-engine`, 2021. Online; accessed 09-Mar-2022.

[62] Estonian Information System Authority. Web eid: electronic identity cards on the web. https://github.com/web-eid/web-eid-system-architecture-doc, 2022. Online; accessed 05-Mar-2022.

[63] Arnis Paršovs. On the format of the authentication proof used by ria's web eid solution. Technical report, University of Tartu, 2021.

[64] Estonian Information System Authority. web-eid.js. https://github.com/web-eid/web-eid.js, 2021. Online; accessed 06-Mar-2022.

[65] Estonian Information System Authority. web-eid-app. https://github.com/web-eid/web-eid-app, 2021. Online; accessed 05-Mar-2022.

[66] Estonian Information System Authority. web-eid-app - authenticate.cpp. https://github.com/web-eid/web-eid-authtoken-validation-java, 2022. Online; accessed 05-Mar-2022.

[67] Estonian Information System Authority. web-eid-app. https://github.com/web-eid/web-eid-app/blob/8bedc0c57698c729012a0c16f590ce0907d27d20/src/controller/command-handlers/authenticate.cpp#L103, 2021. Online; accessed 05-Mar-2022.

[68] Fiyaz Hasan, Rick Anderson, and Steve Smith. Prevent cross-site request forgery (xsrf/csrf) attacks in asp.net core. https://docs.microsoft.com/en-us/aspnet/core/security/anti-request-forgery, 2022. Online; accessed 05-Mar-2022.

[69] Ying Wang, Bihuan Chen, Kaifeng Huang, Bowen Shi, Congying Xu, Xin Peng, Yijian Wu, and Yang Liu. An empirical study of usages, updates and risks of third-party libraries in java projects. In *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 35–45, 2020.

[70] Asmens dokumentų išrašymo centras. Asmens tapatybės kortelė ir elektroninis parašas. https://www.nsc.vrm.lt/downloads.htm, 2021. Online; accessed 05-Mar-2022.

[71] SK ID Solutions. Certificates. https://www.skidsolutions.eu/en/repository/certs/, 2022. Online; accessed 05-Mar-2022.

[72] Estonian Information System Authority. Roca vulnerability and eid: Lessons learned. Technical report, Estonian Information System Authority, 2018.

Appendix

I. Glossary

II. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Gediminas Milašius**,
(author's name)

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Exploring integration complexity of different multi-national eID authentication solutions in the EU private sector,
   (title of thesis)

   supervised by Arnis Paršovs.
   (supervisor's name)

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Gediminas Milašius
11.06.2022

III. Questionnaire

The interview's goal is to understand better the reasons for the poor adoption of eIDs in the private sector. This interview was conducted with the CTO of a multinational logistics company.

1. With electronic key cards, users can authenticate themselves using a piece of hardware, say a card or a USB stick. This authentication method is often more secure than the usual username and password approach. Are you and the company in general aware of this?

2. I would describe an eID scheme as something like your id card, but digitally. There are three main schemes in Estonia: ID cards, Mobile-ID, and Smart-ID. Are you familiar with at least one of them?

3. With the eIDAS regulation, these three eID schemes can create digital signatures, with the legal value of a handwritten signature. Do you have a place in the company where you print a document, sign it, scan it and upload it? Would you switch to a solution that would avoid this process?

4. Without disclosing the worth of transactions floating around the company, would the security benefits of the eID schemes benefit company enough for you to switch to using them?

5. Authentication and signing usually come hand in hand, but if you were to have the ability to choose, assuming authentication and signing both costs equally as much to implement, would you rather spend the resources on authentication or digital signing? SEB Bank used to or still allows for transactions under 50€ to be done without signatures, only authentication. Would you at that point no longer consider the authentication method entirely?

6. Your company deals a lot with automation. Would you be comfortable automating the use of digital signatures in your company's name, or would you rather still have a person at the end manually reviewing and signing documents?

7. Say a human mistake occurs: a person mistakenly signs a document they shouldn't have, and the company faces losses. It would be easy to track who made a mistake with digital signatures. What would your company do in that case?

8. I have four different authentication options a company can take. Assume you would have to pick one of the four and explain the main reasons for your choice.

   The first option uses the primary eIDAS network of Europe to authenticate themselves to any EU public sector service. For example, a Lithuanian citizen can use their eID to sign into Estonia's banks. This network's security is held to the highest standards. Some discrepancies appear because of the criteria, such as Estonians being unable to sign in via Smart-ID to foreign websites. It is significant as a lot of people use Smart-ID. Do you think it is an acceptable solution for you?

The second option would use a company in the middle whose sole responsibility would be to federate the sign-in process. Like the first authentication method, you can also sign in from many more European countries, but this time without using the eIDAS network. A clear advantage over the first one is the more lax security requirements, allowing other authentication methods such as Smart-ID. Keep in mind that this authentication method is still highly trustworthy. Would you consider the ability to reach a broader audience at the cost of not using the official infrastructure a risk worth taking?

The third option puts a lot more risk on the company and allows for only a narrow market band. I am talking about smart cards and how a company could accept one, but the server should never trust the certificate a card sends. This approach is challenging to integrate and susceptible to many attacks; however, its advantage is that it is free to operate. If we ignore the personnel costs for maintaining the trust certificates, that is. Would no operational fees be convincing enough to pick this option?

The last option is similar to the third about the challenging implementations and the narrow market band. This time you will not have the advantage of free operational costs. However, you will still benefit from not having an intermediary company. This option would be if you integrated with Smart-ID directly. Is having an intermediary company of concern to you?

9. What is an acceptable price for a single successful authentication? The business model of options 1, 2, and 4 is to charge an amount per authentication. Let's aim for around 10 000 authentications per month; how much do you think is acceptable to spend on such a number? Would 500€ per month be acceptable?

10. Options 2-4 also create digital signatures; the first cannot. Does your opinion change at all about which solution you would pick?

11. An alternative to using government-issued eID solutions, you can also issue them yourself at a highly reduced price and trust factor. This solution is still more secure than the username+password approach. If you were to change how the company performs authentication, would you switch to the internal system, eID scheme, or not switch at all, and why?

IV. Privacy Policy

1. This document explains which personal data is processed on the AuthServer website (auth.eid.gedas.dev) for use as part of master's thesis research.

2. Information we process:

   (a) User's eID authentication data, which can include: given name, surname, country of eID issue, unique identifier provided by eID, birth date;

3. Information we store:

   (a) registration email address as part of the account creation process;

   (b) user's country and unique identifier as provider by services used for external authentication;

4. Data retention policy:

   (a) all data is wiped from the application at 00:00, Estonia time;

   (b) users can manually remove their data by visiting `https://auth.eid.gedas.dev/Identity/Account/Manage/PersonalData`; effective immediately;

   (c) users can download their personal data on the same page as (b);

5. Data shared with third-parties:

   (a) information received from Dokobit service is subject to UAB Dokobit privacy policy;

   (b) information received from Web eID service is subject to RIA's privacy policy;

   (c) information received from eeID service is subject to internet.ee privacy policy;

   (d) when checking the validity of certificates, the issuer defined in the certificate received will receive a certificate identifying information to validate the revocation status;