

A Federation of Web Services for Danish Health Care

Esben Dalsgaard
Chair, SOSI steering committee
Digital Health Denmark (SDSD)
Rugaardsvej 15
DK-5000 Odense C, Denmark
ead@sdsd.dk

Kåre Kjelstrøm
Solution Architect
Silverbullet A/S
Skovsgaardsvaenget 21
DK-8362 Hoerning, Denmark
(+45) 2092 8244
kkj@silverbullet.dk

Jan Riis
Solution Architect / Project Manager
Lakeside A/S
Aabogade 15
DK-8200 Aarhus N, Denmark
(+45) 2160 7252
jri@lakeside.dk

ABSTRACT

Having relevant, up-to-date information about a patient's health care history is often crucial for providing the appropriate treatment. In Denmark, IT systems have been built to support different work flows in the health sector, but the systems are rarely connected and have become islands of data.

To remedy this situation, a service-oriented architecture based on web services for online exchange of health care data between the vast array of heterogeneous IT systems in the sector is being built.

The architecture forms a federation of web services and enables secure and reliable authentication of end-users and systems in the Danish health sector. The architecture is based on national and international standards and specifications. Yet it defines its own profile for secure interchange of data due to a lack of available international profiles that could handle the special needs of the health sector at the time of project inception.

The architecture has evolved through a pilot project from mid 2005 to the end of 2007, and is being tested in a small scale 1st quarter 2008. This paper aims to convey experiences from the project, so rich in benefits that the architecture has been accepted and standardized as the foundation for the future of system integration in the health sector in Denmark.

Categories and Subject Descriptors

C.2.4 [Distributed Systems]: Distributed applications

D.2.11 [Software Architectures]

D.2.12 [Interoperability]: Distributed Objects

D.2.13 [Reusable Software]: Reusable Libraries

General Terms

Performance, Design, Reliability, Experimentation, Security, Human Factors, Standardization, Legal Aspects, Verification.

Keywords

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
IDTrust '08, March 4–6, 2008, Gaithersburg, MD.
Copyright 2008 ACM 978-1-60558-066-1...\$5.00

Federated Identity Management, Web Services, SOA, SAML, WS-Trust, Single sign on, X509 Certificates, Digital Signatures, SOAP, Security Token Service, Health Care, Electronic Patient Records.

1. INTRODUCTION

The IT system landscape in the Danish health care sector contains a plethora of different systems targeting various needs: patient administration, general practicing, specialized care, electronic health recording, citizen access through web based health portals, etc.

The systems fall more or less uniformly into three classes:

- 1) Off-the-shelf systems typically obtained by privately held companies (e.g. health centers)
- 2) Tender based regional systems (e.g. for hospitals) and
- 3) National systems, typically tender based systems hosted by health care related departments.

Some of these systems are integrated today, but typically integration has been done locally, with the aim to reduce information redundancy. The real benefit in terms of quality of patient treatment and care, however, lies in a deeper integration of health care systems across organizational boundaries, such that *all relevant* information for treatment and care is made directly available in the systems that the health care professionals use in their daily work.

Founded in the strategic vision to strive for better quality in patient treatment, better systems for health care professionals, and the optimization of resources, the health care sector in Denmark has started the work on a national health care integration architecture that supports this vision.

The quest for universal availability of relevant and up-to-date information has been *the* most important force in shaping the architecture. There are, however, many other premises that have governed this work, for instance the fact that in this domain, the "business" is never closed even if some or all of its IT systems become unavailable: People will still need treatment and care.

In 2005, the Danish health care sector launched an initiative with the purpose of analyzing, profiling and testing a combination of national and international standards in order to standardize service based integration mechanisms including measures for strong authentication of principals based on PKI.

The initiative was coined the SOSI project for “Service Oriented System Integration”. It was initiated by the Capital Region of Denmark, The Region of South Denmark, and the Danish Medicines Agency. Present in the steering committee was also the Danish Ministry of Science, Technology and Innovation. The project was funded by Danish Regions and is now governed by the Danish National eHealth initiative: Digital Health Denmark [4].

1.1 Life in the Health Sector

The Danish health sector is financed by the Danish government via taxes and treatment is otherwise free. As opposed to other countries such as e.g. the USA this means that there is only one major health care provider, and that many facilities including hospitals are owned by the public sector.

The health sector employs a wide array of different professionals, most notably hospital doctors and nurses, general practitioners (GP), and caregivers to the elderly and disabled.

In contrast to most of the other organizations in the health sector, GPs are private and often times small businesses employing only one or a few doctors as well as a secretary. The use of electronic health record systems (EHR) is widespread in this part of the sector: Today doctors receive patients in their consulting room from behind a computer screen, running an EHR system.

From time to time, the secretary will act on behalf of the GP in taking blood samples, screening patients over the phone while checking health records on his own computer, etc.

GPs will prescribe medications for the elderly or disabled, submit patients to hospitals, receive patients from hospitals for outpatient treatment, and more.

For hospital doctors and nurses, life is somewhat different from that of their GP colleagues.

Sometimes a doctor will work with patients in a ward all day together with a nurse in a situation akin to that of the GP. Wards are often equipped with a single computer that must be shared between the two professionals. At other times, a doctor must take ward rounds and share computers with local nurses and other doctors.

Doctors typically use IT systems in a read-only fashion while employing a dictation machine to take notes and make adjustments in patient treatment. Information is then entered into the system by a secretary who acts on behalf of the doctor.

Where GPs typically use a single system for all purposes, hospital doctors and nurses will use a set of systems during the day.

In another part of the sector, caregivers follow the directions of GPs in administering medicine for the elderly and disabled. Sometimes those receiving care will live in a nursing home, sometimes in private homes.

Because of the geographical distribution of clients, caregivers have a need for mobile access to medical information. In Denmark, this is usually realized through portable computers in the shape of PDAs. In nursing homes, a number of computers are shared by different caregivers in a situation akin to the one in hospitals.

Caregivers cannot prescribe and have only limited access to health care information, but can report the administration of medicines.

Danish law protects the rights of patients through “The Danish Act on Processing of Personal Data” [20] and “The Danish Health

Law” [22]. These laws govern who can access what kinds of information and why, and points out that citizens have a right to know to whom information has been disclosed.

In 2004, a web based national health portal, Sundhed.dk, [23] was launched with the purpose of providing a single point of access to health information for citizens. One vision of the portal is to make it possible for a citizen to learn what health care information is registered about her, and who has viewed it in compliance with the laws.

The health portal hence has a need to pull information from GPs, hospitals, laboratories and more, and to act as a medium between GPs and citizens e.g. for electronic consultation.

In summary, health care professionals across the sector need to exchange information about patients that pass from one practitioner to the next in order to provide the best possible care. At the same time, citizens have the right to gain insight into their own health care records.

These needs call for an IT-infrastructure that provides up to date health care information about patients across the sector and across the country in a timely fashion.

1.2 Privacy vs. Safety

Health care records often contain sensitive data, which could potentially harm a person’s reputation or private life, should it be exposed to unauthorized people. More seriously, though, these records are the basis on which a patient receives care, and errors caused by negligence, malicious intent, or the like can potentially cause physical harm.

For these reasons, health care records are surrounded by security measures.

Ensuring the confidentiality of information while in transit from one practitioner to the next, and while being stored, is imperative to avoid eavesdropping by unauthorized individuals.

Organizations that handle sensitive data are bound by Danish law to ensure that only authorized staff gains access. In order to comply with the privacy acts then, a practitioner should only be able to access information that she is authorized to, and which is of relevance with respect to her current treatment of a patient. Identifying health care personnel with a high degree of certainty, and performing authorization checks are hence prerequisites for exposing information.

Yet even with all the locks and latches of the world, information will eventually be spilled to unauthorized people, typically through authorized personnel. When such a breach is detected, it is imperative to be able to trace the identity of the malefactor for forensic purposes.

The need for privacy is complicated by the fact that access to information is sometimes a matter of life and death. While citizens have the right to privacy, safety has a higher priority in Danish health care. In other words, the life of a patient takes precedence over the unconfirmed exposure of sensitive information to authorized health care staff.

In the event of unconfirmed exposure, tracking the identity and following up on such an event is a necessity in ensuring privacy.

Hospital doctors and nurses typically use not just one, but several IT systems during the day. In the worst case scenario, a user has different identities and uses different credentials with each system.

Logging in and out of systems can therefore be a time consuming task if care is not taken.

The first step towards providing a faster and simpler authentication mechanism is hence to create a single identity with single credentials for access to all systems. The second step is to provide a single step of authentication to all systems, so called single sign on (SSO).

1.3 Availability

The existence of health care IT systems is justified only by promises of improvements in the overall quality and efficiency of patient treatment.

When work routines based on IT systems replace manual, paper based ones, health care professionals will begin to plan their daily schedule accordingly. This means that once a certain quality of service (QOS) has been established for day-to-day operations, this QOS has to be maintained.

In a complex of systems that exchange clinical information, any participant *must* hence minimize the impact caused when other systems fail for instance by switching to emergency states where only locally cached information is available.

The longer such external information systems are unavailable the higher the risk of inefficiency or errors in patient treatment, and hence the higher the risk of physical harm, adverse effects or permanent maladies. As a safeguard, applications, power sources, communication lines, etc. must therefore be highly redundant, and built with robustness in mind to ensure continued operations even when external systems are down.

2. THE SOSI SOLUTION

The Danish National board of Health is responsible for an overall IT-strategy with an ambitious goal: to provide a connected health care sector in which health professionals have access to all relevant EHR data regardless of where citizens seek treatment and no matter where or when this information was registered [5]. It is nonetheless important to stress that the motivation for the SOSI solution is primarily rooted in user requirements.

Most health care professionals wish to provide the best possible quality possible in their work, and hence base their decisions on all accessible and relevant information. There are therefore examples of health professionals starting up at least five applications every morning: some local, some regional, and some national. Single sign on as well as infrequent sign on are hence examples of real world requirements.

The SOSI project aimed at evaluating standards and technologies that could provide value to users by standardizing authentication mechanisms, standardizing the way service providers should expose services and by providing tools that could lower the threshold for both service providers and service consumers to be part of this new game.

Given the environment of disparate IT systems scattered across the country with a need for common information, it was decided to realize this goal through a national Service-Oriented Architecture (SOA) via SOAP based web services over HTTP.

The architecture had to address the availability and security issues identified earlier and build on existing infrastructure to reduce costs, while adhering to national and international standards in order to ensure maintainability.

There exists within the context of SOAP based web services a profusion of specifications aimed at solving various well-known issues from the world of computing: security, reliability, messaging, addressing, transactions, etc.

Each specification adds levels of complexity and typically provides not just one, but multiple ways of achieving the same overall goal. Add to this the fact that often times, specifications from different bodies compete to become the de-facto standard, each attacking the problem at hand in slightly different ways: There's the recipe for non-interoperability.

The solution to this problem comes in the shape of profiles that cut through the stack of specifications, paving a narrow path of design choices for specific usage scenarios.

In the world of federated single sign on over the Internet, a number of profiles and specifications exist. OASIS defines the SAML specification [14], which is implemented by the Internet2 initiative Shibboleth [6]. A large group of non-Microsoft companies drive The Liberty Alliance Project [21], whose specifications extend SAML. IBM and Microsoft push the WS-Federation [7] specification and implement support in a range of products.

The Ministry of Science, Technology and Innovation (MVTU) drives much of the standardization effort in the Danish public sector. It does so in part by evaluating international profiles and specifications and classifying them in an interoperability framework [25]. For federated identity management, SAML 2.0 is classified as the preferred framework of choice. Any Danish SSO architecture should hence build on SAML, which was therefore chosen for SOSI as well.

2.1 Security Architecture

An IT system that participates in a service-oriented architecture must weigh the risks associated with revealing information to unauthorized people against available security measures.

In the health sector, risks generally include unauthorized disclosure of sensitive information and in some cases physical harm. While some types of information e.g. classifications of diseases may be harmless if disclosed, others such as patient records are often not.

Because in Denmark it is legally the responsibility of the data owning organization to prevent unauthorized disclosure, every web service provider must perform a risk analysis and potentially strengthen security measures before exposing information via the federation.

That said, it still makes sense to define a set of basic security properties that are always in place, and to lay out a simple set of security choices that can be implemented depending on identified risks.

A cornerstone in the security architecture of SOSI, the CIA triad addresses the aspects of Confidentiality, Integrity and Availability [19]:

When in transit, data will pass over a number of networks and confidentiality is ensured through the use of encryption techniques to avoid eavesdropping, no matter the content.

Communicating parties also need a guarantee that data has not been altered during transit. The integrity property is also guaranteed via cryptographic techniques.

Finally, services must be available to be of any value to users, guaranteed by redundancy of critical components, communication lines, and enforced through policies and agreements.

Single sign on is achieved through the use of a trusted third party, who will verify credentials on behalf of all parties in the federation. This is a delegation of trust model, which reduces the burden of federation participants: instead of knowing all possible users, it is enough to be able to verify claims from the trusted third party.

SAML assertions are useful for propagating claims about digital identities that can be used in e.g. authorization checks. To be trusted by a third party, though, credentials must additionally be supplied for external verification.

Credentials come in many shapes and sizes with different security properties. While passwords may be simpler to manage, they are susceptible to eavesdropping attacks and may be easy to crack if not chosen carefully and changed often. X509 certificates offer stronger properties, non-repudiation and confidentiality, but require equally careful handling of private keys to be trusted.

The SOSI federation employs strong credentials based on X509 certificates, which provide a high level of certainty in the identification of the sender.

2.2 Architectural Building Blocks

When designing how exactly confidentiality and integrity should be realized through cryptography, it came down to reusing an existing national VPN based health care network or to employ “end-to-end” message encryption/signing. Although end-to-end encryption/signing may seem captivating because messages can then pass freely over potentially any network, the benefits were deemed smaller than the burden of encrypting and signing streams of very large messages.

A large part of the health sector organizations in Denmark are already connected to the above mentioned VPN network known as “SDN” [8]. The network was originally planned for teleconferencing, exchanging large amounts of data e.g. x-ray images, and accessing web based applications in a secure manner.

Any organization that wants access to services on SDN is evaluated for relevancy and must sign a mutual agreement per system-to-system connection. Although cumbersome, this procedure provides a certain degree of certainty that the network is primarily made up of organizations with legal business in the health sector.

By supplying an integrity and confidentiality protected transport mechanism, which is immune to known security attacks, and which has many of the relevant organizations connected already, SDN is useful for web services as well.

Part of the Danish it-strategy is the mandated use of digital signatures for secure identification of health care personnel. An important precondition in the design of a solution for SOSI would therefore be to leverage the Danish national certificate initiative, OCES.

OCES provides all the features of a nationally implemented X509 based PKI. What makes OCES even more interesting is the Open Source components and commercial products that surround the initiative which can also be used in WS integration. Specifically a Signature Server [1] that enables secure centralization of private keys in the client environment and the OpenOCES [18]

components that enable Java access to the Windows crypto API have been used and evaluated in the SOSI project.

Last but not least the current OCES operator has several web services that support the OCES initiative, e.g. services for converting certificate subject serial number to the national Danish person identification number.

In mid 2005, when the SOSI project was initiated, none of the existing Single sign on projects gave good solutions to the particular needs of the project. Although there was a SOAP binding for SAML, no profile existed that laid out a complete protocol stack for exchanging SOAP messages with SAML assertions, while achieving Single sign on to web services.

There was and still is a heavy bias towards providing SSO for browser-based clients, with specifications relying on facilities such as HTTP redirect and cookies.

However, most health care applications in Denmark are non-browser based. In most cases users need specialized and highly supportive systems, something which until very recently was not feasible to build with web browser technology.

Many of the existing SSO projects include services or components that increase system dependencies instead of reducing them, thereby introducing potential single points of failure. One of the best examples is that most SSO profiles mandate service provider initiated user authentication, typically done by communicating with an authentication service.

In browser-based applications, where connectivity is a precondition, service provider initiated authentication is natural and probably the only viable solution, in part because client systems are very thin and mostly session based applications. If the central authentication service is unavailable, service providers cannot be called either. .

However, in a pure web service integration architecture, where clients more often than not are servers themselves, it is possible to build more fault tolerant architectures.

Client-initiated user authentication can for instance be mandated, and issued security tokens be cached in client system for later use. In a model where security tokens are additionally off-line verifiable by service providers, only users that are not already authenticated will be affected if the authentication service is unavailable.

So, although many of the existing profiles had elements that could be reused, the use-cases and interaction schemes were not. A basic SAML and WS-Trust based profile [9] was therefore created based on the following principles:

1. A user should be able to authenticate with the federation once and then be able to use any service for which she has authorization for as long as she can present a valid federated security token. The design should, in other words, help reduce the number of sign-ons to the federation.
2. Basic information security should be provided by the existing security infrastructure.
3. Using a client initiated authentication scheme, a WS client (WSC) system should be responsible for logging the user into the federation before starting to interact with any WS provider (WSP).

4. Inspired by current work on short-lived PKI certificates [16][17] the security token must have a limited lifetime and hence eliminate the need for revocation checks by WSPs.
5. Security tokens must be off-line verifiable by WSCs and WSPs, i.e. without having to communicate with any third party.
6. Security tokens should be able to carry basic end-user and client-system attributes that most WSPs use for authorization and/or logging. The design should support trust re-use, such that when the credentials within the security token have been verified, the embedded attributes can also be trusted. In effect this reduces the effort that WSPs must put into implementing web services. It also stabilizes the entire architecture by reducing system dependencies to a minimum.
7. Security tokens must not be subject to theft, i.e. measures must be put in place to hamper hostile token takeover.

The proposed technical solution consists of:

- A trusted federation Security Token Service (STS) with a maximum validity of 24 hours.
- Security tokens as digitally signed SAML Assertions
- Client initiated authentication that results in STS signed SAML assertions
- Core attributes embedded in the SAML security token
- Message integrity through digital signing of SAML assertions combined with web service body data.
- Confidentiality of transport, but not of message data in a trusted circle of participants.

Figure 1 shows a simple interaction between a WSC, an STS, and a number of WSPs. Please note the WSC initiated authentication scheme and that the WSPs are not depending on access to the STS to verify security tokens and basic attributes:

- Step 1. The user authenticates with the federation either just-in-time before calling a service or as part of the local log-on to the WSC system. The WSC builds a SAML assertion with core attributes and user credentials, in this case a digital signature.
- Step 2. The STS checks that
 - a. the digital signature of the WSC system is valid
 - b. the WSP system certificate is valid and not revoked
 - c. the WSP is on the white-list of systems that are allowed to enter the federation
 - d. the user's digital signature is valid
 - e. the user's certificate is valid and not revoked
- Step 3. The STS now seeks to verify that the client-specified core attributes are valid by using backend attribute services. Some of these verified attributes are cached for a short period for optimization purposes.
- Step 4. If everything is OK, the security token is digitally signed by the STS and returned to the WSC.
- Step 5. The security token can now be used in interactions with different WSPs until it expires.

- Step 6. Upon receipt, the WSPs validate the security token by verifying the STS digital signature and leverage the embedded attributes for logging and authorization.
- Step 7. Finally a result, i.e. business information or an error is returned.

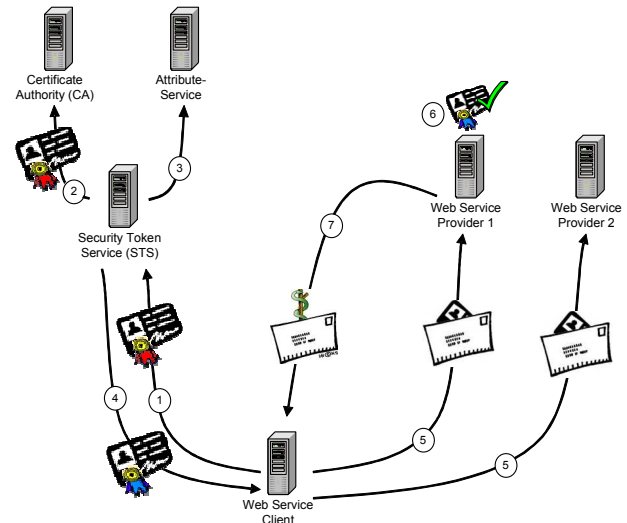


Figure 1: a simple WSC/WSP interaction

It is important to note the temporal flexibility between steps 1-4 and steps 5-7: The authentication request for the STS could be executed as part of the user's log-on to the WSC system. They could even be performed asynchronously and would only become blocking if the user entered a step in a workflow where entrance to the federation was needed, for instance in order to gather information from outside the system.

What is actually happening in the STS is that the user's long lived credentials are converted to short lived credentials combined with core attributes. In the SOSI project these short-lived security tokens were coined "virtual health professional identity cards" (or ID cards for short). The STS issues digitally signed ID cards that by PKI properties are verifiable by service providers without on-line access to the STS or attribute services.

In effect the federation is established by the sole STS certificate, which also means that the STS certificate must be protected viciously. In the SOSI project it was discussed whether a security breach on the STS should be handled by policies and emergency procedures (e.g. by phoning all service providers and having them shut down their services) or having all service providers check the STS certificate for revocation from time to time.

The latter seems to be the only manageable solution and was decided upon. The solution hence yields a system dependency from WSPs to the CA revocation service, but these calls are done infrequently and independently of the high volume business calls. The risk of compromising the federation certificate is comparable to the risk of compromising the CA root certificate, which is considered to be very low.

The maximum validity of ID cards is 24 hours in the SOSI architecture. However, the amount of trust a WSP can put into the security token depends on how "fresh" the token is. In other words the level of trust degenerates over time.

If the token is 5 minutes old when received by a WSP, the WSP can be pretty confident that the same user is still operating the console. The SOSI proposal opens up for the possibility that a WSP can choose to reject security tokens that are “not fresh enough” at its own discretion. In effect this means that users are, within 24 hours, *only* challenged to authenticate themselves when they use a service that needs authentication proof, which is more “fresh” than the security token the user currently holds.

It is worth noting that this mechanism is in direct opposition to the “single-sign-on” requirement: If all WSPs reject ID cards that are more than 5 minutes old, the user will be forced to re-login to the federation every 5 minutes, effectively disabling SSO.

This kind of time-out requirement should, however, only happen for service-operations, which provide or receive very sensitive information and hence demand very rapid security token time-outs, which means they are more likely to require a real digital signature and hence entail the end-user to provide credentials for activating the private key.

The decision, of which credential strength, security token time-out level and which verifiable authorization attributes the specific service provider should require, must be based on a thorough risk analysis.

The time-out level is a true measure of “trust”, and hence a scheme where WSPs’ have different time-out requirements for different (types of) client systems can be considered. In a national service infrastructure this can also be used for governing client systems towards federation compliance.

For the STS and WSPs’ to be able to distinguish which client system an ID card pertains to, the request needs to carry information about the “system-principal” that is performing the request on behalf of the user.

The current profile mandates that any request will carry two digital signatures: One from the user on the SAML assertion and one from the requesting system on the entire message, which provides a combination of system identity and message authentication.

The STS will check both signatures as well as both certificates and will acknowledge successful verification by embedding references to the original certificates, and sign the ID card itself. By including secure certificate hashes into the issued ID card, WSPs will be able to verify message authentication signatures without checking the client-systems’ certificate for validity or revocation since the revocation checks have been performed “recently” by the STS.

2.3 A New Profile

SOSI defines a web service profile where every request and response message will carry an assertion that identifies the sender, and every assertion will contain credentials that allow the receiver to verify the identity of the sender.

Credentials come in the shape of digital signatures over the XML elements that make up the SAML assertion stored in compliance with the XML-Signature specification [26]. A receiver can use the unique certificate identifier to lookup the person or company via trusted web services.

Routing information is embedded in the SOAP messages to enable other transport mechanisms than HTTP. WS-Addressing [26], the de-facto standard for such information in web services, is leveraged and profiled. WS-Addressing contains a unique

message-id, which is required on all messages in the shape of a Universally Unique Identifier (UUID), used to prevent replay attacks.

All messages are integrity protected through an additional digital signature on the XML that makes up the SOAP body, the WS-Addressing headers, and the SAML token. This ties the sender’s identity to the supplied information and prevents identity theft: Without this signature, an eavesdropper would be able to create a new message and embed a stolen SAML assertion, effectively impersonating someone else.

Messages are confidentiality protected only when in transit via the underlying transport layer. The federation is made up only of well-known parties who have signed an agreement before being granted physical access. Further each party is bound by laws not to disclose sensitive information, and it was therefore decided that it was not necessary to employ message encryption.

The profile defines an SSO mechanism, which uses WS-Trust [15] messages to perform authentication via a trusted third party, the Security Token Server (STS). WS-Trust was chosen over its SAML equivalent because it seemed to have the most momentum with respect to actual implementations in products at the time.

The profile uses a request-response model and leverages the SAML specification’s SOAP binding to embed the assertions in SOAP headers. In this respect, the profile is completely in compliance with the SAML specification.

3. IMPLEMENTING SOSI

At the time of writing, the pilot project is being tested on a smaller scale, yet many relevant observations have been made not least in the process of realizing the architecture.

3.1 Participating Systems

From the outset of the SOSI project, two existing hospital systems were planned to implement the solution: One in the capital region of Denmark and one in the region of South Denmark.

More specifically it was planned to improve the quality of available patient related medicines information by connecting the medication modules of these systems with the national medication and prescription services hosted by the Danish Medicines Agency. The latter, then had to be enhanced as well to be able to participate in the federated solution. The fourth party in the system setup was the authentication service (STS).

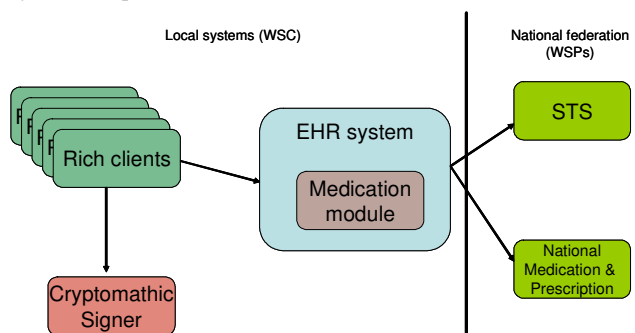


Figure 2: Participating systems overview

Since health care personnel moves around and uses the rich client part of the medication applications at different terminals, PKI private key handling is complicated somewhat. Fortunately this problem had already been identified and solved by the regions when the clinical workplaces were integrated with the Danish

eHealth Portal [23]. In both cases a regional “signing server” [1] was introduced that centrally handles and protects private keys for all end users (see figure 2).

When the EHR system needs to authenticate the user with the national federation, it challenges the rich client with parts of the STS request. Once the signed challenge returns to the EHR system, it can produce the final STS request and store the returned ID card locally for 24 hours. As the user then moves to another terminal, the same ID card can still be used in the national web service federation.

The requirements of the STS service were specified and quotes were requested from relevant vendors. The STS was required to handle the following number of requests for ID cards with the following response times for the pilot test period:

- Verification and signing of 12.000 ID cards in 24 hours
- A maximum continuous throughput (MCT) of 1500 ID cards per hour, with a peak of 10 simultaneous ID card requests.
- Mean response times < 2 seconds while upholding MCT
- 95% percentile < 5 seconds while upholding MCT
- 99% percentile < 10 seconds while upholding MCT

Of the two quotes received, one based was on an off-the-shelf solution, and one was custom development. The off-the-shelf solution was considerably more expensive than the custom solution and had some annoying limitations in the usage of SAML/WS-Trust. In addition, the modules that had to be developed to encompass the special health sector needs were proprietary, which made it difficult to move to another product without considerable expenses. The STS was optionally requested to be developed under a reciprocal Open Source license, which only the custom solution adhered to.

The custom solution was chosen for the pilot project and has now been developed and committed to the Public Danish Open Source initiative [12]. This fulfills the needs for the SOSI pilot project and a few soon to come smaller projects, but as the national federation develops, and the commercial STS market is maturing, the custom made STS will most probably be replaced with an off-the-shelf product.

At time of writing 3 of 4 systems have been developed/enhanced and a very slow scale-up has begun. Until now only a few doctors have been authorized to use the new facilities in their clinical systems and only a few dozens of ID cards have been retrieved from the STS. Nonetheless users have welcomed the solution and can see the benefits and perspectives right away.

The overall work flow has performance issues, which are currently being analyzed by the vendors. On the STS the biggest bottleneck is the OCES web service that resolves the Danish person ID from the PKI certificate. This STS back-end verification more often than not accounts for a 2 second “delay” which of course is unacceptable.

One of the problems in the current state of the project is that the systems are actually misleadingly slow because they were built for higher transaction volumes. In the current sparse user volume, caches and prepared database statements and connections are getting evicted between invocations, which results in unnecessary startup penalties almost every time the user is accessing the system.

When the (real) performance problems have been solved, more doctors will be allowed access to the system, and the false startup penalties should disappear. Since the new facilities can be activated on a per user basis, the systems can be scaled at a very fine grained level, enabling the monitoring of STS and WSP systems very carefully as more users get access.

3.2 Lowering the Threshold

The outlined architecture puts quite a burden on service providers and clients. In order to join, a party must be able to handle SAML, speak SOAP over HTTP, create and verify digital signatures, communicate with a trusted third party, check revocation lists, and more.

Because the architecture is based on standards it might be possible to find commercial off-the-shelf (COTS) products that would be able to handle these tasks. Each product would require some configuration, though, and might have a steep price tag on it. In some parts of the health sector, in particular at the GP’s, steep prices are not an option.

Further, the federation is built on existing software, which was not initially meant to communicate with other systems. Hence, any implementation of federation infrastructure will include some amounts of custom code.

It is crucial to the success of the health federation that all parties have the means to join in. As a remedy to the situation, it was decided early on to establish an open source organization, which would provide software that implemented federation infrastructure and that could support the vendors that were to use the Open Source products.

A set of tools would make it viable for even small companies to join in, lowering the threshold and enabling the federation.

Faced with a lack of product support due to a lack of profiles for SAML based web service interactions, it was clear early on that support for the SOSI profile would have to be implemented into every IT system in the federation in a custom manner.

While the SAML AttributeStatement although somewhat verbose in its syntax is not hard to implement, creating XML digital signatures is an entirely different story.

A programmer whose development platform does not support the XMLDSig [28] standard out-of-the-box will have to piece signing and verification functionality together e.g. from a crypto API. This includes creating secure hashes of data, implementing canonicalization algorithms, encrypting and decrypting, base 64 encoding and decoding, manipulating XML structures, and more.

As a remedy to this ailment it was decided early on by the open source organization to build a Java based library, “Seal.Java” [2] that would provide an abstraction, which would allow a developer to work with high-level primitives and not worry about envelope formats, digital signatures, or the darker secrets of the base-64 algorithm.

The EHR systems that entered into the SOSI project from the hospital side were mainly Java based, and while Seal.Java was relevant here, it could not be used with the EHR systems from the GP side that are mostly rich Win32 or .NET applications. This fact spawned Seal.NET [10], with the exact same purpose as its Java sibling.

Both projects have been constructed on an Open Source license and are available for general scrutiny via the web.

Third party software often suffers from the “not invented here syndrome”, a problem which the library projects sought to address by going to great lengths in testing, tuning, and publishing quality reports. When response times are high, multi-threading issues fixed, code coverage of the test suite well above 95%, and long term endurance testing of all API methods does not show any leaks; when the entire library is built from scratch, and all tests exercised on a nightly basis with fresh results published online in the morning [3], chances are that others will accept it as stable and useful as well.

This aggressive strategy for quality reporting has proven to be highly effective. Adoption of both libraries is high with most peers using them. This makes it very easy and low-cost to implement minor adjustments and optimizations to the SAML profile, because most vendors just need a new version of the library to be able to produce a new request or response XML.

Parallel to the developed Open Source components, a support organization was established. Vendors can contact a support mailing list for free and very rapidly get answers to general work flow or detailed coding questions, or solve problems or advice to workarounds with the libraries within hours. The response to this sort of support has been very positive and has had a very positive influence on the vendor / customer relationship.

3.3 The XML Schema Challenge

During development of the libraries, the idea surfaced that it would be useful to implement XML Schema validation for the XML, SAML, SOAP, WS-Trust, etc. that was passed around. Validation would improve overall quality and general faith that standards were followed.

Unfortunately that proved to be very difficult.

A profile that cuts across specifications is in effect limiting the number of possible choices a developer can make. Wouldn't it be great if it were possible to express the new set of limited choices in supporting schemas as well? It isn't! For instance, how do you express that it is a requirement to have an enveloped signature inside a SAML Assertion if the user authenticated using PKI?

Expressing such complex conditions is beyond and above what you can do with XML Schema. Even if it were possible, the problem of how to version a set of XML Schemas in concert arises: There is no great way today in which existing schemas can be narrowed under the same name space.

For development purposes, it was therefore decided to modify the original schemas, SAML, SOAP, etc. to allow only those elements that were mandated by the profile. While helpful for testing, these schemas would not be used for production because they were overly strict and hence not compatible with COTS products that will often attach extra non-critical SOAP headers, id's, etc.

Recently, a central test center for web services in the Danish health sector [11] has been launched. The test center is capable of emulating clients and servers for various concrete services to a certain point not including too much business logic. It is manned by staff that can monitor requests and responses, and aid in debugging. The center provides value in ensuring that all parties wishing to implement a service will get past syntactical obstacles with the profile as well as with the model of the service in question.

For reasons of maintainability, loose coupling, and reuse, web services should be designed in a contract-first manner, where the

service interface, the WSDL, including data models and service end-points, is defined independently of the code that implements it.

Unfortunately not that many off-the-shelf toolkits give good support to such a development paradigm. Now that tooling was already being implemented, it was decided to craft a contract-first WSDL tool that would allow for the easy creation of service interfaces as well.

Tooling is an important mechanism to help bridge the gap between specifications and products. Tools can make the difference as to whether a particular IT system will be able to participate in a certain scenario or not and without them, the SOSI project would not have been possible.

While providing tools and libraries to lower the threshold of integrating existing systems, there is also a risk associated with such a strategy: Source code, no matter how well written, will always have flaws, errors, or lack a feature for a given situation. Without an organization to maintain the code, it will eventually fail to be helpful.

On the other hand, it is actually possible to tune the profile over time or align it with coming standards, when all parties rely on a few infrastructure components. Given the volatility of the current specifications for federations of services, this might prove to be a crucial strength.

3.4 Federation Verification and Control

Because the federation is established by digital signatures from the authentication service (STS), all parties in the federated infrastructure are able to check federation security tokens (ID cards) by validating the STS signature and the STS certificate.

Checking the signature is a matter of working with XML-Signatures, something that enjoys library support in many programming languages. The STS certificate check is a little more involved as the entire chain up to the issuing authority must be validated including revocation checks. Fortunately it can be done rather infrequently due to the very low risk that STS credentials should get compromised. As a hardening measure, the SOSI production STS has been placed in the same production room as the Danish OCES CA services, hence the same policies for access, audit etc. are enforced for this system.

Having one key pair that establishes a federation makes it easy to establish test federations e.g. for pre-production, integration-test and other development stages. This has been employed in the SOSI project, where the OCES CA has issued two certificates – one for production and one for integration/pre-production.

However, the STS production certificate will expire some day, and various mechanisms have been put in place to make a smooth transition from one certificate to another. For instance the Seal.Java library has STS certificate checks incorporated that are resistant to STS certificate renewal.

While developing and testing the solutions, the vendors have had some trouble with the VPN based dedicated health network (SDN) through which all production services are accessed. None of the vendors could be authorized to access SDN directly as they are not public health related companies, and since the production servers are never on two networks at the same time, it was very cumbersome getting access to logs, debugging information, not to mention changing configurations or deploying new software versions.

As mitigation for this, the test federation was established on the public Internet, where developers can access a test STS from their development environments. Unfortunately some services e.g. back-end ones used by the STS for core attribute verification only exist on SDN and the semantics and performance aspects can therefore only be tested in the production environment. This is one of the issues that must be resolved in the coming national infrastructure.

3.5 Standardization

There is no profile without a specification, and the SOSI efforts have therefore been captured in a document known as The Apt Web Service (DGWS) [8].

Owned by the Danish Centre for Health Telematics (MedCom), who is responsible for standardizing the communication between parties in the health sector and operating SDN, the specification has now reached version 1.1.

While there might be a few adjustments to make on DGWS in the aftermath of the first pilot, this version is currently poised to become the de-facto standard for all web service communication in the Danish health sector.

To ensure compliance, MedCom staff mans the aforementioned test center [11] and is providing practical assistance in testing that web service clients and servers implement not only DGWS, but also the business data exchanged in DGWS SOAP envelopes.

With specifications, tools, and a certifying entity in place for free, there should be a viable chance of getting even the smaller vendors on board the federation.

4. LOOKING FORWARD

Federated identity management has evolved over the past few years, and there are now a couple of frameworks that might address the needs in the SOSI architecture. Most notably, the Liberty Alliance recently published version 2.0 of its Liberty ID-WSF, which defines interaction scenarios for web service clients with SAML via SOAP over HTTP. Future work will examine Liberty and alternatives in order to evaluate whether it would be feasible to align the SOSI project without critical impact.

Parallel to the initiatives in the health sector, MVTU is driving other pilot projects that address slightly different needs, but define similar architectures. The OIOSI [24] project for instance is being pushed for secure asynchronous business document exchange via the internet using PKI and web services.

The health sector specific infrastructure must to be aligned with a future national infrastructure for all of the public sector without violation of the identified design criteria.

While digital signatures are currently being touted in Denmark as *the* technology to identify citizens and professionals alike, it is loved more by engineers than by end users. A digital signature is cumbersome to deal with and certificate management is not mature from an end-user's perspective.

On the longer term, biometrics and RFID for near field identification could have a place as the identifying technology, e.g. to release the private key of a certificate instead of a password. The driving force for biometrics or RFID will, however, not be the increased security, but the fact that identification will become easier for end users.

At the time of writing multiple initiatives governed by Digital Health Denmark that extend the SOSI architecture are in the

crucible. Most notably a security gateway, SOSI-GW, is being developed that enables trusted domain cross-over. This vastly reduces the effort in implementing SOSI support for web service clients and will enable single-sign-on to the national federation across multiple client systems. The gateway becomes the single point of entry to the national web service federation from the trusted domain, and all sorts of common services can be centralized in this service.

Digital Health Denmark is also launching initiatives to analyze possibilities for limiting the impact when the back-end verification systems of the STS fail. One possibility is to allow the STS to issue partially verified ID cards as long as every attribute verification state is clearly stated. The STS may also cache verification states and skip re-verification if the cached verification has not decayed too much. This reduces the impact of failing verification services or decayed attributes to users who are using federation services that need these specific attributes.

Of interest is also the possibility to use "break the glass" solutions combined with logging and control mechanisms. If for instance a WSP requires a newly verified STS attribute, and the presented ID card contains a non-verified or decayed attribute, the WSP may choose to return a "break the glass" warning that informs the client system and subsequently the user that she will be subject to investigation if proceeding. This could be combined with asynchronous mechanisms that seek to resolve the unverified claims.

The SOSI project has not produced any results in the area of web service governance, but as the solution scales to multiple clients and providers this will become a very important issue. Digital Health Denmark is also launching initiatives to meet these challenges.

5. CONCLUSION

The proposed architecture and profile have been developed and tested in real life, and the results are very promising with respect to both the development process as well as the implementation effort.

At the time of writing end-user feedback has not been systematically gathered yet, but purely from a technical perspective the proposed architecture exhibits a set of nice qualities that support the special requirements for the health sector:

- **Single-Sign-On** to Web Services within the national federation / trust domain.
- **Authentication levels.** Users and systems can be authenticated with different degree of certainty, depending on the credentials that the principal presents. This is in accordance with the guidelines [13] from NIST on which MVTU has based their authentication guidelines.
- **Reduction of impact** of unavailability of services. If, for instance the STS is unavailable, only users without a security token or with an expired security token will be hindered in performing their treatment. All other users can continue to treat patients until their security token expires.
- **Reduction of the effort** that WSCs and WSPs must put into implementing web services. WSPs only have to trust/check *one* certificate (the federation certificate owned by the STS)

- **Maximum performance.** The number of requests/messages is minimized. When trust has been established and the user has logged in to the federation, the WSC and WSP communicate directly with no third party involved.
- **Transparency and flexibility** through the use of Open Source licensed tools and products.
- **Reuse of existing infrastructure.** The design reuses existing infrastructure for establishing secure channels that takes care of confidentiality and stream integrity and prevents known cryptographic attacks.

The positive experiences with the architecture and profile outweigh the downside of not yet having international standards that fit the requirements of the Danish health sector.

SOSI is currently acknowledged as the best solution to the integration challenge, and at the time of writing, multiple projects that implement modules and systems based on the SOSI design, its standards and the associated Open Source tools are in the making.

6. REFERENCES

- [1] Cryptomathic, Cryptomathic Signer, <http://www.cryptomathic.com/Default.aspx?ID=124>
- [2] Danish Regions, 2006-2007, SOSI Components, http://www.sosi.dk/twiki/bin/view/ProjectManagement/SOSI_Products
- [3] Danish Regions, 2006-2007, SOSI Seal Component, <http://www.sosi.dk/sosi/seal/>
- [4] Digital Health Denmark, 2007, <http://www.sdsd.dk/>
- [5] Digital Health Denmark, 2007, National IT strategy, http://www.sdsd.dk/arch/_img/9080664.pdf
- [6] Internet2/MACE, 2007, Shibboleth Project – Internet2 Middleware, <http://shibboleth.internet2.edu/>
- [7] Lockhart et al., 2007, Web Services Federation Language, <http://www.ibm.com/developerworks/library/specification/ws-fed/>
- [8] MedCom, 2003-2008, The Danish Health Network, <http://www.medcom.dk/wml110002>
- [9] MedCom, 2007, The Apt Web Service (DGWS), <http://sundcom.health-telematics.dk/svn/DGWS/>
- [10] MedCom, 2006-2007, Den Gode Webservice Tools, <http://www.medcom.dk/wml110344>
- [11] MedCom, 2007, Testcenter, <http://testcenter.medcom.dk/>
- [12] National IT and Telecom Agency, 2007, Software exchange: Forum for software development in the public sector, <http://www.softwareborsen.dk/>
- [13] NIST, Electronic Authentication Guideline, 2006, http://csrc.nist.gov/publications/nistpubs/800-63/SP800-63V1_0_2.pdf
- [14] OASIS, 2007, SAML 2.0, http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security#samlv20
- [15] OASIS, 2007, WS-Trust, <http://docs.oasis-open.org/ws-sx/ws-trust/v1.3/ws-trust.pdf>
- [16] PGP Corporation, 2006, PGP White Paper – Revocation Made Simpler, http://download.pgp.com/pdfs/whitepapers/Revocation-SLCS_060104_F.pdf
- [17] Profile for Short Lived Credential Services X.509 Public Key Certification Authorities with secured infrastructure. <http://www.tagpma.org/files/IGTF-AP-SLCS-20051115-1-1.pdf>
- [18] TDC, 2007, OpenOCES, <http://www.openoces.org/>
- [19] The CIA Triad, http://en.wikipedia.org/wiki/Information_security
- [20] The Danish Data Protection Agency, The Act on Processing of Personal Data, Datatilsynet, Law no. 429, May 31st, 2000, <https://www.retsinformation.dk/Forms/R0710.aspx?id=828>
- [21] The Liberty Alliance, 2007, Liberty Alliance Project, <http://www.projectliberty.org/>
- [22] The Ministry of Health and Prevention, The Health Law, Law no. 546, June 24th, 2005, <https://www.retsinformation.dk/Forms/R0710.aspx?id=10074>
- [23] The Ministry of Health and Prevention et al., 2007, Sundhed.dk, <http://www.sundhed.dk>
- [24] The Ministry of Science, Technology and Innovation, 2006, OIO Serviceorienteret Infrastruktur, <http://www.oio.dk/arkitektur/soa/infrastruktur>
- [25] The Ministry of Science, Technology and Innovation, 2006. The Interoperability Framework. <http://standarder.oio.dk/English/>
- [26] W3C, 2006, Web Services Addressing 1.0 – Core, <http://www.w3.org/TR/ws-addr-core/>
- [27] W3C, 2002, XML-Signature Syntax and Processing, <http://www.w3.org/TR/xmldsig-core/>
- [28] W3C, 2002, XML-Signature Syntax and Processing, Recommendation. <http://www.w3.org/TR/xmldsig-core/>