

---

>

---

---

# **SAML Profile for Federation in Danish Public Sector V2.0**

**(DK-SAML 2.0)**

***Høringsversion***

---

## Contents >

---

1	Introduction	5
1.1	Purpose	5
1.2	Background	5
1.3	Referenced documents	6
1.4	Terminology	6
1.5	Pre-requisites	7
2	Architectural Overview	9
2.1	Basic Service Access with Authentication	9
2.2	Service Access with Single Sign-On	11
2.3	Access via a Portal and Attribute Retrieval	12
2.4	Single Logout	14
2.5	Federation using Persistent Pseudonyms	15
2.6	Profiles supporting the scenarios	18
3	DK-SAML Profile Content	19
3.1	Profile Information	19
3.2	Governance and Management of Profile	19
3.3	Errata	20
4	Web Browser SSO Profile	21
4.1	User Agent accesses Resource	21
4.2	Service Provider Determines Identity Provider	22
4.3	Service Provider sends <AuthnRequest>	22
4.4	Identity Provider Authenticates Principal	23
4.5	Identity Provider sends <Response>	23
4.6	Service Provider grants or denies access	24
5	Identity Provider Discovery Profile	26
5.1	If Discovery Fails	26
6	Single Logout Profile	27
6.1	Local Logout Requirements	28
7	Authentication Assertion Profile	29
7.1	Generic Assertion Requirements	29
7.2	Attribute Encoding Rules	32
7.3	Core Attributes	32
7.4	Sector-specific attributes	35
8	OCES Attribute Profile	37
9	Persistent Pseudonym Attribute Profile	42
9.1	Rolling Migration	42
9.2	Profile Requirements	42
10	Attribute Service Profile	43
10.1	Profile Overview	43
10.2	Requirements for Request/Response Messages	43
10.3	Processing Rules	44
10.4	Attribute Naming and Encoding	45
10.5	Meta Data	45

---

10.6	Discovery	45
10.7	Binding	46
10.8	Privacy	46
10.9	Security	46
11	Profile Considerations	48
11.1	Naming and Identifiers	48
11.2	Assertion ID as Transaction Identifier	48
11.3	Meta Data	48
11.4	Privacy in a Danish Context	49
11.5	Security Considerations and Requirements	50
11.6	Error Handling	55
12	Guidance on determining product compliance	56
13	Architectural Decisions	60
13.1	Attribute Profile in Requests	60
13.2	Authentication Level in Requests	60
13.3	Signing of Meta Data	61
13.4	OCES Subject as Attribute	62
13.5	Binding for Single Logout Profile	63
13.6	Requirements for Identity Provider Discovery Profile66	
13.7	Name Identifier Management Profile	66
13.8	Attribute Encoding	68
13.9	Core User Attributes to include in Authentication Assertion	69
13.10	Include Certificate Issuer in OCES Attribute Profile70	
Appendix A:	Overview of Profile Changes	72
13.11	Experience from the e-Authentication initiative	72
13.12	Profile changes	72
Appendix B:	References	74

# Document History

Version	Date	Initials	Changes
2.0	17.08.07	SPN	Document history reset as document is ready for public hearing

---

# 1 Introduction

---

>

## 1.1 Purpose

This document contains a set of profiles of the OASIS SAML 2.0 standard for use within Danish public sector federations. It is named *SAML profile for federation in Danish public sector V2.0* or in short *DK-SAML 2.0*

The DK-SAML 2.0 profile is an umbrella for several profiles described in this document. DK-SAML 2.0 replaces the previous version 1.1 of Danish SAML profiles; see appendix A for an overview of the changes.

The SAML standard is an XML-based framework for describing and exchanging security information between on-line business partners. This security information is expressed in the form of portable SAML assertions that applications working across security domain boundaries can trust. The OASIS SAML standard defines precise syntax and rules for requesting, creating, communicating, and using these SAML assertions [SAMLTechOverv].

The profiles contained in this document tailor the generic SAML framework to the needs of the Danish public sector by:

- Specifying which SAML profiles that must be supported.
- Limiting choices and complexity by narrowing the generally wide set of options allowed by SAML, for example regarding bindings.
- Taking the Danish OCES standard (and other Danish standards) for digital signatures into account.
- Dealing with scenarios required by portals such as the Danish Citizen Portal (borger.dk).
- Extending SAML with local requirements e.g. for stating the level of authentication- in assertions and how to include sector-specific attributes.
- Including experience and best-practice from other countries including the American E-Authentication initiative and New Zealand's e-government programme.
- Aligning with Danish law and regulations including Persondataloven and Registerloven.

## 1.2 Background

The National IT and Telecom Agency in Denmark has for several years worked on an initiative aiming for a common approach to authentication and user management for E-Government in Denmark. In the process the initiative has adopted several elements from the E-Authentication initiative in USA (<http://www.cio.gov/eauthentication>).

A requirement for the Danish initiative is to enable government Service Providers to use external authentication services instead of developing their own, Single Sign-On (SSO) across disparate systems and establish a foundation for federated identity management.

Goals supporting innovative new public sector IT-solutions as well as cost-reductions through re-use of authentication services, faster development cycles for E-Government applications, consistent application of security technology, improved user experiences (via Single Sign-On) and reduced administration cost.

The IT and Telecom Agency produced a set of documents and published them for public hearing (ending September 2005). The base document [ITTArch] defined the overall architecture and scenarios for Single Sign-On (SSO) to be supported. The architecture was based on the concept of federation and was technology-agnostic such that it could be implemented using different underlying technologies

Late 2005 the first versions of Danish SAML profiles (V1.0 and V1.1) were written. They were based on SAML 2.0 and the proposed architecture [ITTArch] which contained some non-SAML constructs. These constructs were introduced by the American e-Authentication programme to cover some of the gaps in SAML 1.1 and included:

- An authentication portal which stores meta data about applications and login services in the federation, facilitates selection of applications and services by interacting with the user, centralized error handling and correlation of distributed transactions.
- Cookies and parameters for e.g. communicating a selected application or authentication service outside the SAML framework.

Experience from the deployments in USA and other countries have since brought the following facts to light:

- SAML 2.0 has increasingly been adopted by software vendors.
- The non-SAML components in the US E-Authentication Architecture were a significant source of cost and complexity for Service Providers to implement (e.g. required custom development). Most of these components can be replaced with equivalent SAML 2.0 functionality.
- Some SAML constructs are easier to deploy and operate than others (e.g. POST binding is simpler than artifact binding).
- Based on the above, the US E-Authentication program – from which the Danish initiative has adopted several elements – decided to take advantage of the SAML 2.0 standard and at the same time simplify their architecture.

These factors combined a desire to offer – as an option – different identifiers for a given user at different service providers have motivated updates of the Danish SAML profiles – hence this document. Appendix A provides an overview of the changes in the new version of the DK-SAML profile.

### 1.3 Referenced documents

All referenced documents are listed in Appendix B. Each reference has an identifier in square bracket, like [SAMLCore]. Document are reference in the text using this identifier.

### 1.4 Terminology

The following table defines the most important concepts and terms used in this document. For a more detailed presentation of relevant federation terminology, please refer to [Terms].

Term	Description
------	-------------

Identity Provider	<p>An Identity Provider (IdP) is a trusted entity in a federation that authenticates users and generates authentication assertions or other assertions that vouch for a user's (subject's) identity.</p> <p>An IdP may create, maintain, and manage identity information for Users – in which case it also can act as an <i>Attribute Authority</i>.</p> <p>An IdP may also create assertions for WS-Security messages, and may in that context act as a Security Token Service (STS)</p> <p>An Identity Providers is also known as “Credential Service” (US e-Auth term), ”Authentication Authority” or “Login Service”.</p>
Service Provider	<p>A Service Provider (SP) is an entity that relies on assertions from an Identity Provider (IdP) to authenticate or authorize subjects' actions on its resources.</p> <p>A Service Provider is also known as “Relying Party” (SAML 1.1 term) which now has been adopted by WS-* as: ”a Web application or service that consumes Security Tokens issued by a Security Token Service”.</p> <p>A Service Provider will usually provide application services to end users – and as a prerequisite require knowledge about the user's identity in order to grant access.</p>
User	<p>Users comprise persons, application entities such as web services, or named machines—thus, a user is anything identified on a system, or on the network, as a named, individual entity and challenged to present credentials authenticating its identity.</p> <p>A User is an entity that can acquire a federated identity, that is capable of making decisions, and to which authenticated actions are done on its behalf.</p> <p>Users are also known as “subjects” or “principals”.</p>
Assertion	<p>A piece of data produced by an Identity Provider (SAML authority) or similar regarding an act of authentication performed on a User, attribute information about the User, or authorization permissions applying to the User with respect to a specified resource.</p> <p>Assertion is similar to <i>Claim</i> used in WS-* terminology. The term Assertion will be used in general.</p>
Trust	<p>The willingness of a party to take action based on its relationship with another party.</p>

## 1.5 Pre-requisites

The DK-SAML profiles largely build on the following:

- OASIS SAML 2.0 standards and profiles [SAMLCore], [SAMLProf], [SAMLBind], [SAMLMeta], [SAMLConf]
- OCES, the Danish PKI [OCESPers], [OCESMedarb]

---

>

---

- OIO guide on core attributes [ITTAtrib]
- OIO guide on authentication levels [ITTAuthLevel]

---

## 2 Architectural Overview

>

---

This chapter briefly presents an overview of the architecture in order to provide the reader with the context in which the SAML profile is used<sup>1</sup>.

The architecture will be illustrated in the following sections by highlighting the interactions between entities in different scenarios. The main entities are:

- **Identity Provider** – provides authentication of users as a service to the federation and (optionally) hosts an attribute service where identity attributes can be queried.
- **Service Provider** – provides (web) application services to end-users which require authentication.
- **Portal** – is a (thin) portal which collects / aggregates application services from different Service Providers. Since this SAML profile deals exclusively with the web browser SSO scenario we will only consider browser-based integration from a portal to Service Providers. Web-service (i.e. SOAP) based integration is thus not considered (e.g. WSRP or native web service integration). *In all aspects relevant to this profile, the portal will be considered as a Service Provider.*
- **User** – for example a citizen or employee who wishes to access services and has credentials to prove his / her identity (e.g. an OCES certificate).

### 2.1 Basic Service Access with Authentication

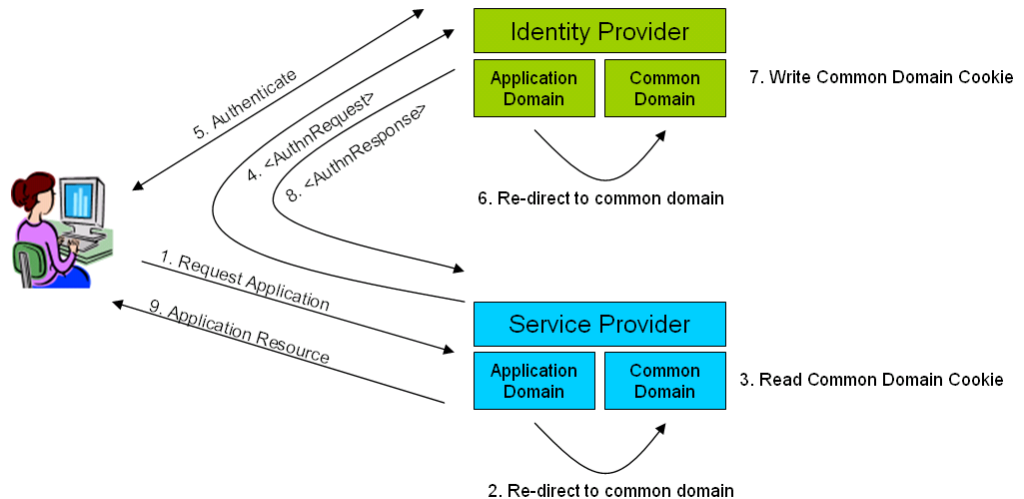
The first scenario shows the interaction where a user accesses a Service Provider directly (via her browser) to get a service with no prior session established. The Service Provider therefore redirects the user to the Identity Provider for authentication and session establishment.

The scenario shows the following profiles:

- Web Browser SSO Profile described in chapter 4
- Identity Provider Discovery Profile described in chapter 5
- Authentication Assertion Profile described in chapter 7

---

<sup>1</sup> Detailed functional and non-functional requirements for the Identity Provider beyond the SAML 2.0 requirements are out of scope for this document as they vary according to the different business requirements.



**Figure 1: Service Access with Authentication**

The steps are:

1. The user requests (via her browser) a web application resource from the Service Provider.
2. The Service Provider determines that the resource is protected and that the user has no current session. The Service Provider therefore re-directs the user to his common domain web server in order to discover the user's Identity Provider(s).
3. The Service Provider reads the common domain cookie to discover the user's Identity Provider(s) (via the SAML Identity Discovery Profile). The cookie will be empty in this scenario since the user has no current SSO session with an Identity Provider. The Service Provider will select its default Identity Provider. If the Service Provider supports multiple Identity Providers, he may prompt the user to select Identity Provider.
4. The Service Provider creates and signs an authentication request and re-directs the user to the Identity Provider with the request as a parameter.
5. The Identity Provider receives the authentication request, learns that the user has no current (IdP) session, and therefore initiates authentication of the user. The user authenticates with valid credentials (e.g. his OCES digital signature).
6. After successful authentication the Identity Provider establishes a session and re-directs the user's browser to his common domain server.
7. The Identity Provider stores his identifier in the common domain cookie. This will facilitate later discovery of the Identity Provider and re-use of the session (hence Single-Sign On).
8. The Identity Provider re-directs the user back to the Service Provider with a signed response containing a SAML assertion. The Service Provider validates

the assertion, creates a user session<sup>2</sup>, and performs an authorization check on the resource originally requested by the user.

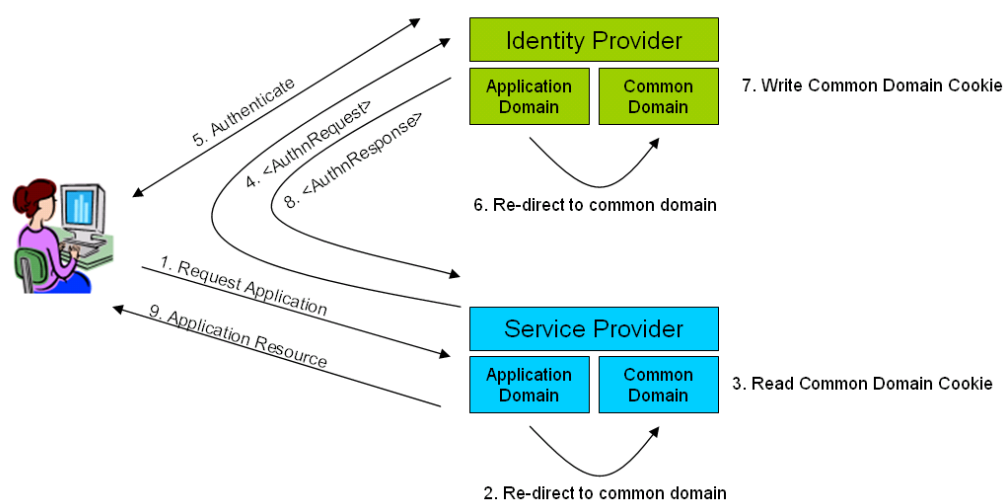
9. If the authorization check succeeds the requested application resource is returned to the user.

Note that subsequent requests to the same Service Provider can be authenticated via the user's Service Provider session and will thus not require interaction with the Identity Provider.

## 2.2 Service Access with Single Sign-On

The second scenario shows the interaction where a user accesses a Service Provider directly (via her browser) to get a service when an Identity Provider session has previously been established. The Service Provider still re-directs the user to the Identity Provider but here the previous session is re-used and no user authentication takes place.

The scenario uses the same profiles as the previous scenario.



**Figure 2: Service Access with Single Sign-On**

<sup>2</sup> It is here assumed that the attributes contained in the assertion are sufficient for the Service Provider to establish a session. This will often be the case if the assertion contains for example CPR- or OCES PID numbers. Later in this section more advanced scenarios will show the interaction when this assumption cannot be made.

The steps are:

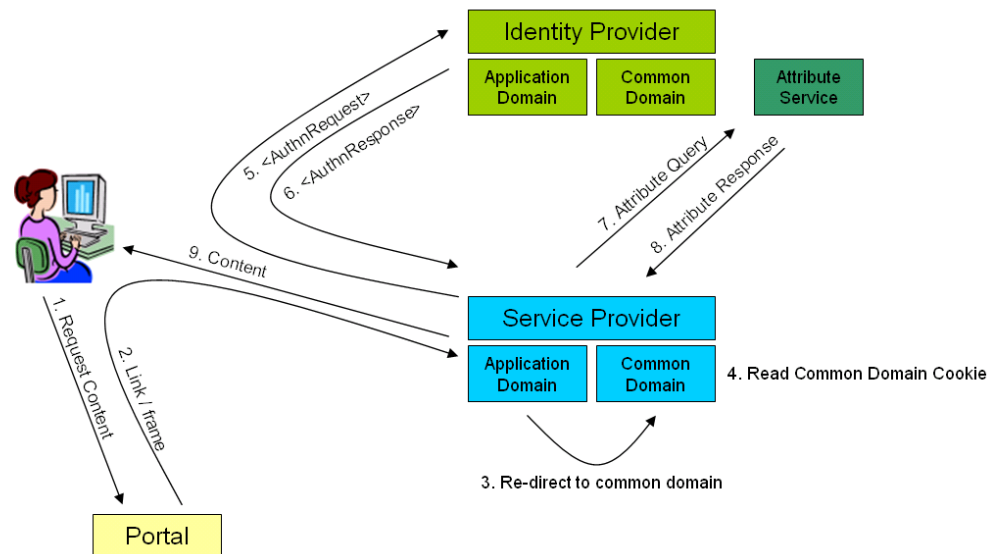
1. The user requests (via her browser) a web application resource from the Service Provider.
2. The Service Provider determines that the resource is protected and that the user has no current session. The Service Provider therefore re-directs the user to his common domain web server in order to discover the user's Identity Provider(s).
3. The Service Provider reads the common domain cookie to discover the user's Identity Provider(s) (via the SAML Identity Discovery Profile). The cookie contains a reference to the user's current Identity Provider with whom she has a session.
4. The Service Provider creates and signs an authentication request and re-directs the user to the discovered Identity Provider with the request as a parameter.
5. The Identity Provider receives the authentication request, learns that the user has an active session, and therefore initiates single-sign on. The Identity Provider re-directs the user back to the Service Provider with a response containing a SAML assertion.
6. The Service Provider validates the assertion, creates a user session, and performs an authorization check on the resource originally requested by the user. If the authorization check succeeds the requested application resource is returned to the user.

### **2.3 Access via a Portal and Attribute Retrieval**

The third scenario shows the interaction where a user accesses a Service Provider via a portal and an Identity Provider session has previously been established. The Service Provider still redirects the user to the Identity Provider but here the previous session is re-used and no user authentication takes place. Furthermore, the Service Provider requires additional identity attributes about the user in order to e.g. make an access decision or perform its service. It therefore sends an attribute query to an Attribute Service co-located with the Identity Provider.

The scenario shows the following profiles:

- Web Browser SSO Profile described in chapter 4
- Identity Provider Discovery Profile described in chapter 5
- Authentication Assertion Profile described in chapter 7
- Attribute Service Profile described in chapter 10



**Figure 3: Service Access via Portal with Attribute Query**

The steps are:

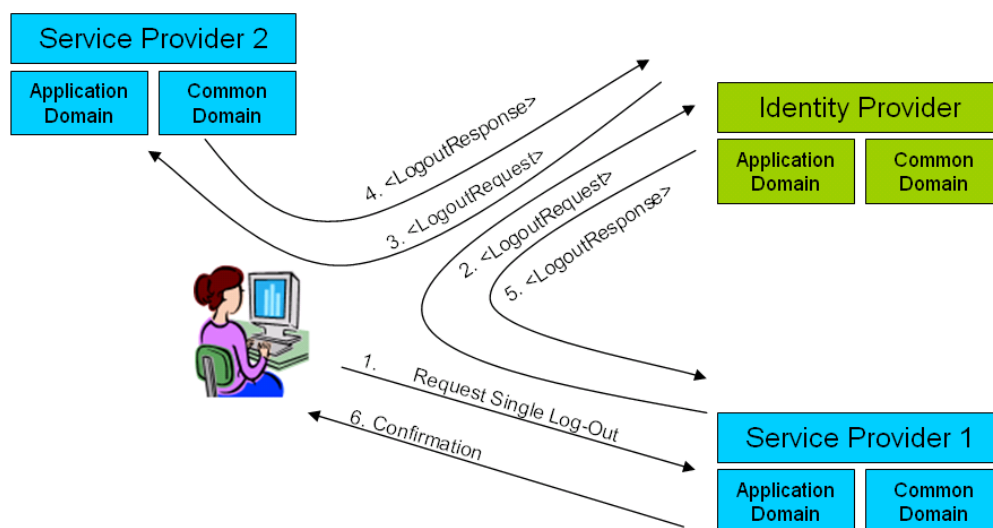
1. The user accesses the Portal which aggregates content and services from different Service Providers.
2. Via the portal, the user requests an application resource from a Service Provider. In browser-based integration scenarios, the portal will either link to the Service Provider or frame its content (e.g. using an iFrame). Web service integration is thus not considered.
3. The Service Provider determines that the resource is protected and that the user has no current session. The Service Provider therefore re-directs the user to his common domain web server in order to discover the user's Identity Provider(s).
4. The Service Provider reads the common domain cookie to discover the user's Identity Provider(s) (via the SAML Identity Discovery Profile). The cookie contains a reference to the user's current Identity Provider with whom she has a session.
5. The Service Provider creates and signs an authentication request and re-directs the user to the discovered Identity Provider by posting this request.
6. The Identity Provider receives the authentication request, learns that the user has an active session, and therefore initiates single sign-on. The Identity Provider re-directs the user back to the Service Provider with a response containing a SAML assertion. The Service Provider validates the assertion,

creates a user session, and performs an authorization check on the resource originally requested by the user.

7. The Service Provider determines that it needs additional attributes about the user in order to either make an authorization decision or deliver its service, so it sends an attribute query to an Attribute Service co-located with the Identity Provider<sup>3</sup>.
8. The Attribute Service authenticates and authorizes the query and returns an attribute assertion. The assertion is validated by the Service Provider and the attributes are extracted for use e.g. in an access decision.
9. The application resource originally requested by the user is returned (if access decision allows it).

## 2.4 Single Logout

A natural supplement to Single Sign-On is Single Logout whereby a user can terminate her current sessions with all Service Providers and Identity Providers. The illustration below shows a scenario where the user requests logout at a Service Provider – alternatively the user can request logout directly at the Identity Provider.



**Figure 4: Single Logout**

The scenario shows the following profiles:

<sup>3</sup> Implicit in the sequence is that the Service Provider may be required to collect the user's consent to retrieve the attributes.

- Single Logout Profile described in chapter 6.

The steps are:

1. The user contacts Service Provider 1 (e.g. via an application) to request single
2. Service Provider 1 contacts the user's Identity Provider to request Single Log out.
3. The Identity Provider determines which additional Service Providers the user has active sessions with (Service Provider 2) and sends them a request for logout.
4. Service Provider 2 terminates his user session and responds to the Identity Provider.
5. The Identity Provider terminates his user session and responds to the Service Provider.
6. The Service Provider responds with a confirmation to the user that all current sessions have been terminated.

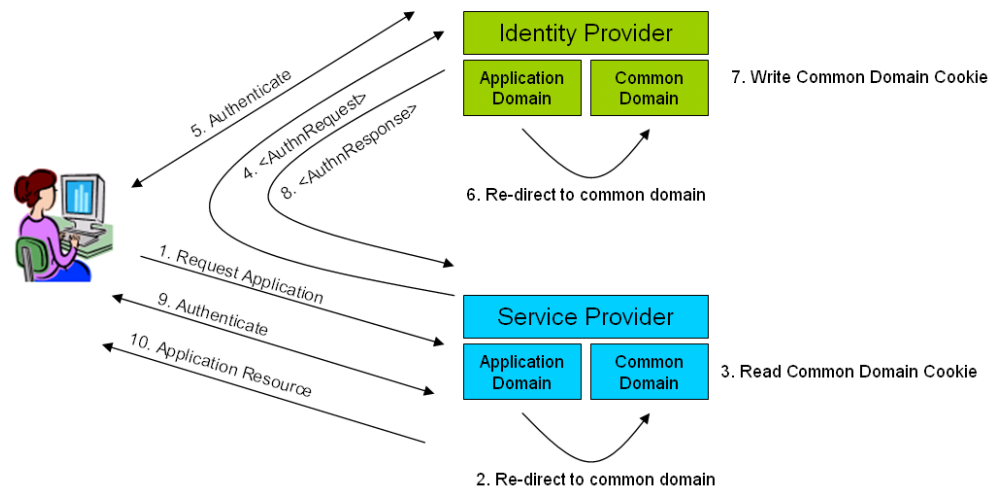
## 2.5 Federation using Persistent Pseudonyms

In the previous scenarios it has been assumed that assertions issued by the Identity Provider contain information that allows the Service Provider to uniquely identify the user and establish a session. This will often be the case if the assertion contains CPR or OCES PID numbers and the Service Provider has organized his internal user registry with these data as keys. Hence, no explicit linking of user accounts between Service Provider and Identity Provider needs to take place. This mode of operation is commonly known as “federation using identity attributes” or simply “account mapping”.

In order to support enhanced privacy requirements, it must be possible for Service Providers to avoid using CPR or PID numbers in their internal user registries. This will make it more difficult to correlate user information across different Service Provider organizations. Therefore, this profile mandates support of federation using persistent pseudonym identifiers as described below. This will facilitate dynamic (on-the-fly) creation of federated identities as part of the normal SSO message exchange.

Further, it is desirable to support individual migration of locally registered users into the federation.

Note that the strongest disadvantage of this scheme is that the user needs to (initially) authenticate twice in order to establish a federation of identities between Identity Provider and Service Provider. This process establishes “account linking”.



**Figure 5: Federation Using Persistent Pseudonyms**

The scenario shows the following profiles:

- Web Browser SSO Profile described in chapter 4
- Identity Provider Discovery Profile described in chapter 5
- Authentication Assertion Profile described in chapter 7
- Persistent Pseudonym Attribute Profile described in chapter 9

The steps are:

1. The user requests (via her browser) a web application resource from the Service Provider.
2. The Service Provider determines that the resource is protected and that the user has no current session. The Service Provider therefore re-directs the user to his common domain web server in order to discover the user's Identity Provider(s).
3. The Service Provider reads the common domain cookie to discover the user's Identity Provider(s) (via the SAML Identity Discovery Profile). The cookie will be empty in this scenario since the user has no current SSO session with an Identity Provider.
4. The Service Provider creates and signs an authentication request and re-directs the user to his default Identity Provider with the request as a parameter. The request instructs the Identity Provider (via a NameIDPolicy element) to provide an assertion containing a persistent name identifier for the user.
5. The Identity Provider receives the authentication request, learns that the user has no current session, and therefore initiates authentication of the user. The user authenticates with valid credentials (e.g. his OCES digital signature).

---

>

---

6. After successful authentication the Identity Provider establishes a session with the user and re-directs the browser to his common domain server.
7. The Identity Provider stores his identifier in the common domain cookie. This will facilitate later discovery of the Identity Provider and re-use of the session (hence Single Sign-On).
8. The Identity Provider generates and stores (or retrieves should one already exist) a persistent pseudonym identifier, includes it in a SAML assertion, and re-directs the user back to the Service Provider.
9. The Service Provider validates the assertion. In order to establish a mapping from the received pseudonym identifier to the internal user account, the Service Provider initiates authentication of the user.
10. Upon successful authentication of the user, the mapping between the pseudonym identifier and internal account is stored for later re-use. Subsequently a user session is established, and an authorization check on the resource originally requested by the user is performed. If the authorization check succeeds the requested application resource is returned to the user.

Note: It is only during the first interaction between a Service Provider and Identity Provider that the user has to authenticate twice. This is performed in order to establish the link between accounts; in subsequent SSO flows the persistent identifier is re-used and the user only has to authenticate to the Identity Provider.

## 2.6 Profiles supporting the scenarios

The scenarios in this chapter illustrate parts of the requirements that have gone into the Danish SAML 2.0 federation profile. The following chapters detail the restrictions and additions that has been added to the OASIS SAML 2.0 profiles in the adaption into the DK-SAML 2.0 profile.

To sum up, and for reference when reading on, the following table lists which of the profiles that apply to the different scenarios earlier in this chapter.

Scenario -> / Profile:	Basic Service Access with Authentication	Service Access with Single Sign- On	Access via a Portal and Attribute Retrieval	Single Logout	Federation using Persistent Pseudonyms
Web Browser SSO Profile	X	X	X		X
Identity Provider Discovery Profile	X	X	X		X
Single Logout Profile				X	
Authentication Assertion Profile	X	X	X		X
OCES Attribute Profile	X (implicit)	X (implicit)	X (implicit)		
Persistent Pseudonym					X
Attribute Service Profile			X		

---

## 3 DK-SAML Profile Content

>

---

This chapter begins the normative part of the Danish SAML profile, DK-SAML.

DK-SAML consists of a set of sub-profiles of the SAML 2.0 profiles [SAMLProf]. These are described in subsequent chapters:

- Web Browser SSO Profile in chapter 4,
- Identity Provider Discovery Profile in chapter 5,
- Single Logout Profile in chapter 6,
- Attribute Service Profile described in chapter 10.

The goal of DK-SAML is to provide further specialization of the SAML profiles, impose restrictions and limit options left open by SAML in order to ensure a high level of interoperability. This further specialization is described in the following chapters, and structured into the following profiles:

- Authentication Assertion Profile in chapter 7.
- OCES Attribute Profile in chapter 8.
- Persistent Pseudonym Attribute Profile in chapter 9.

Where DK-SAML does not explicitly provide SAML guidance, one must implement in accordance with applicable OASIS SAML 2.0 requirements.

### 3.1 Profile Information

**Identification:** dk:gov:saml-profile:2.0

**Contact Information:** itst@itst.dk

**SAML Confirmation Method Identifiers:** The SAML V2.0 "bearer" confirmation method identifier, urn:oasis:names:tc:SAML:2.0:cm:bearer, is used by this profile.

**Description:** Given below.

**Updates:** SAML2.0 profile for SSO in Danish Public Sector V1.1

### 3.2 Governance and Management of Profile

The profile is intended to require a minimal amount of central management and governance by IT- og Telestyrelsen / Ministry of Science, Technology, and Innovation.

The table below describes a few management / governance areas and how they are to be handled:

Area	Comment
Profile Versioning	The versioning and content of the base DK-SAML profile is maintained solely by the IT and Telecom Agency in Denmark.

---

>

---

	The version of the profile is included explicit in assertions.
Identifiers and certificates	Participants must choose unique identifiers according to the syntax and rules defined in this profile (must be an URL reference within their domain). Per construction there will be no need to centrally manage these identifiers to ensure uniqueness.
New attributes and sub-profiles	Identity Providers are allowed to add identity attributes to the profile and even establish sub-profiles containing specific sets of attributes (e.g. for the healthcare sector). However, it must be done according to the rules describes in this document to avoid confusion with the “standard” attributes. Special attention must be paid to Danish and International legislation (e.g. “Persondataloven”).
Compliance to profile	There will (so far) be no central authority to evaluate whether a given implementation is compliant with this profile. Note however that the Liberty Alliance Conformance testing procedures [LibInterop] will cover large parts of the profile. More guidance regarding compliance is found in Chapter 12.
Trust	Trust will be handled via business agreements between the participants and the trust organization of the federation. It is established technically by defining which certificates to trust.
Meta Data Repository	The IT and Telecom Agency will <i>not</i> maintain a central repository with meta data (e.g. service end points) and will not specify mechanisms for automated meta data exchange. It must be handled via agreements between the involved parties.

### 3.3 Errata

Errata and updates to this profile will be published at the following URL:  
<http://tobedecided.dk>.

Comments to the profile should be sent to: [itst@itst.dk](mailto:itst@itst.dk)

---

## 4 Web Browser SSO Profile

---

>

This chapter contains a profile which is a further specialization of the Web Browser SSO Profile from [SAMLProf]. Unless stated explicitly, all messages, policies, processing rules etc. of the original profile are inherited.

The steps in the basic scenario covered by the profile are illustrated in the figure below (figure from [SAMLProf]):

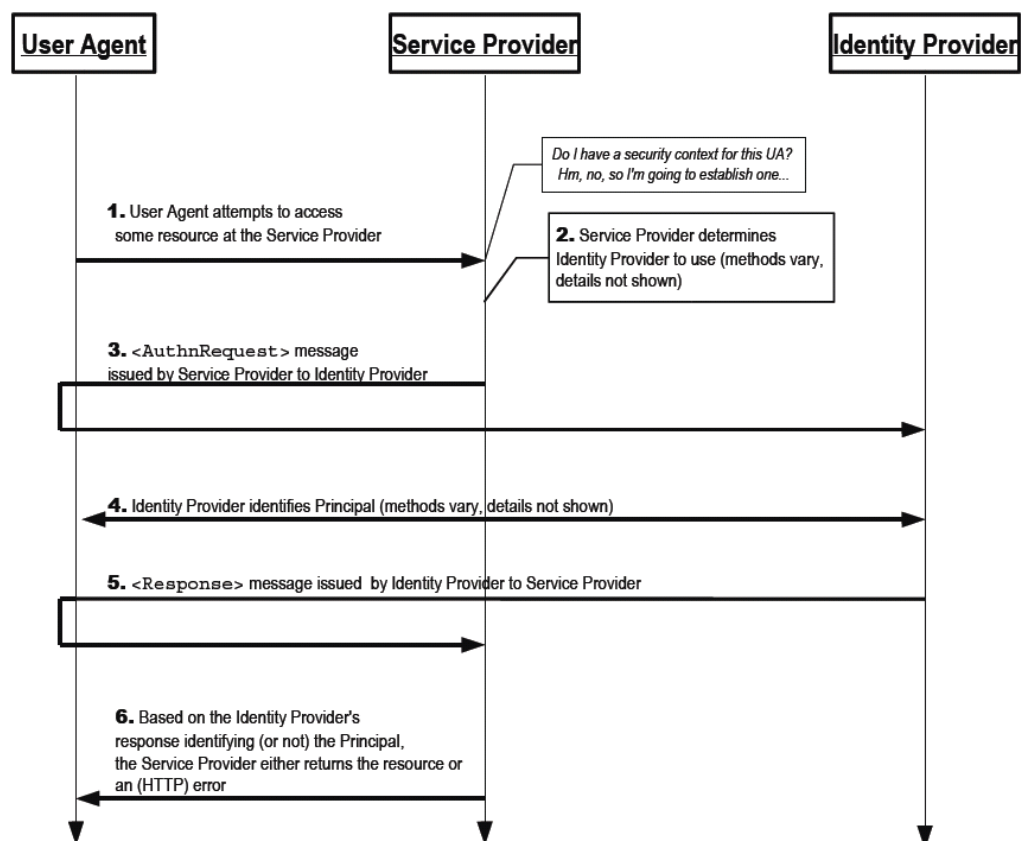


Figure 6: Steps in basic SSO

In the following, each step will be described in detail including specifics of bindings and processing rules.

### 4.1 User Agent accesses Resource

This profile contains no restrictions on this step as it is governed by the HTTP protocol. Note that a resource may be requested via a link or frame from the portal, but it will still result in plain HTTP(s) request from the user agent to the Service Provider.

As in the SAML profile, the RelayState mechanism MAY be used by the Service Provider to associate subsequent profile exchanges with the original request. However,

---

>

---

for privacy reasons this parameter must not reveal any details of the request (e.g. it must be opaque).

## **4.2 Service Provider Determines Identity Provider**

In the original OASIS SAML profile, this step is implementation dependent and a number of different options exist. In this profile, the step **MUST** follow the Identity Provider Discovery Profile described in chapter 5. This will help to ensure that the architecture is open towards multiple Identity Providers.

## **4.3 Service Provider sends <AuthnRequest>**

### **4.3.1 Location of Identity Provider**

In order to send the request, the Identity Provider's single sign-on service must first be located. The SAML profile states that meta data **MAY** be used for this purpose but in the Danish profile this is a **MUST**. No prior exchanges between Service and Identity Providers should take place without prior establishment of legal- and business agreements and exchange of meta data.

### **4.3.2 Binding Selection**

The SAML profile allows a selection of different bindings; this profile mandates use of HTTP Redirect binding based on the deployment experiences from the American e-Authentication initiative. The HTTP exchange **MUST** take place over (one-way) SSL / TLS to ensure confidentiality of the request (integrity and authenticity is provided by digitally signing the request as described in the next subsection).

### **4.3.3 Signing the Request**

In the original OASIS SAML profile, signing of the request is optional. In this profile, digital signing of the request is mandatory and should be performed using the Service Provider's OCES Company<sup>4</sup> signature whose certificate is exchanged as part of the meta data.

---

<sup>4</sup> When new OCES Certificate Policies (e.g. device certificates) are published, these will be analyzed to determine whether they can be used for this purpose.

## 4.4 Identity Provider Authenticates Principal

This step is governed by the requirements to the individual Identity Provider. For example, the Identity Provider must support authentication using OCES digital signatures.

### 4.4.1 Single Sign-On

If the Identity Provider already has a valid session with the user, authentication of the user should not be performed and instead single sign-on be used. Two exceptions to this are:

- The user may have chosen to opt-out of single sign-on via his preferences with the Identity Provider.
- The Service Provider may have included the ForceAuthn attribute in the request with a value of “true”. This instructs the Identity Provider to re-authenticate the user even if he already has a session.

### 4.4.2 Selecting Authentication Mechanism

An Identity Provider may support several authentication mechanisms each providing a different assurance level for the user’s identity. Examples are username/password login, authentication via digital signatures bound to OCES certificates, PIN code login etc.

It is not allowed for Service Providers to specify the requested level of authentication via extensions to the <AuthnRequest> message as this may be problematic for many software products to handle.

Instead an Identity Provider MAY deploy different logical IdP services with different end-points and let the Service Provider choose between these. Alternatively, an Identity Provider MAY let the user select among different mechanisms interactively or let the choice be a part of the user’s preferences.

## 4.5 Identity Provider sends <Response>

When an Identity Provider processes a request and produces a response, it must follow the rules defined in this section.

### 4.5.1 Processing Rules

Only Service Providers with prior agreements may be served by the Identity Provider.

If the Identity Provider wishes to return an error, it MUST NOT include any assertions in the <Response> message. Otherwise, if the request is successful, the <Response> element MUST conform to the following:

- The <Issuer> element MAY be omitted, but if present it MUST contain the unique identifier of the issuing Identity Provider; the Format attribute MUST be omitted or have a value of `urn:oasis:names:tc:SAML:2.0:nameid-format:entity`.

---

>

---

- A successful response **MUST** contain exactly one <Assertion> with exactly one <AuthnStatement> element. Each assertion's <Issuer> element **MUST** contain the unique identifier of the issuing Identity Provider; the Format attribute **MUST** be omitted or have a value of `urn:oasis:names:tc:SAML:2.0:nameid-format:entity`.

The background for the above restrictions is limitations in COTS products and a desire to make the profile easy to deploy.

#### **4.5.2 Assertion Contents**

The assertion included in a response must follow one of the two attribute profiles described later in this profile. Specifically, the assertion **MUST** state the level of authentication achieved.

#### **4.5.3 Location of Service Provider**

In order to send the response, the Service Provider's assertion consumer service must first be located. The SAML profile states that meta data **MAY** be used for this purpose but in the Danish profile this is a **MUST**.

#### **4.5.4 Bindings**

The OASIS SAML profile allows several different bindings; this profile mandates use of the HTTP POST binding based on the deployment experiences from the American e-Authentication initiative. The HTTP exchange **MUST** take place over (one-way) SSL / TLS to provide for confidentiality of the request (integrity and authenticity is provided by digitally signing the request).

#### **4.5.5 Signing**

The response message **MUST** be signed using the Identity Provider's OCES Company signing key<sup>5</sup>.

### **4.6 Service Provider grants or denies access**

The Service provider receives and processes the response message with the enclosed assertion. In addition to the processing mandated by the SAML profiles, the Service Provider must check that the level of authentication in the received assertion is equal to or higher than the level required by the resource requested by the user.

---

<sup>5</sup> When new OCES Certificate Policies (e.g. device certificates) are published, these will be analyzed to determine whether they can be used for this purpose.

---

>

---

Based on this information from the assertion it creates a session with the user and performs an authorization decision for the resource originally requested by the user. If the access check is successful the requested (web) resource is returned to the user.

---

## 5 Identity Provider Discovery Profile

>

---

The Identity Provider Discovery Profile described in [SAMLProf] enables a Service Provider to discover which Identity Providers a principal is using with the web browser SSO profile.

The profile relies on a cookie that is written in a domain common between Identity Providers and Service Providers in a deployment. The cookie contains a list of Identity Provider identifiers and the most recently used IdP should be at the end of the list.

*DK-SAML directly adopts the profile and requires conforming Service and Identity Providers to support it. This will facilitate an open architecture where multiple Identity Providers can be leveraged.*

*The cookie must be transient such that it is not stored between browser sessions.*

Note however that the identifier for the Identity Provider must follow the requirements specified in this profile in section 11.1 (i.e. be an URL reference within their domain).

The name of the common domain is to be determined by the federation organization that the entity is part of.

### 5.1 If Discovery Fails

There may be situations where a Service Provider cannot discover an Identity Provider via the above mechanism. For example, the user may not yet have a session with an Identity Provider or may have deleted the cookies in his browser.

In such a situation, the Service Provider can select its default Identity Provider. If the Service Provider supports multiple Identity Providers, he may prompt the user to select Identity Provider.

---

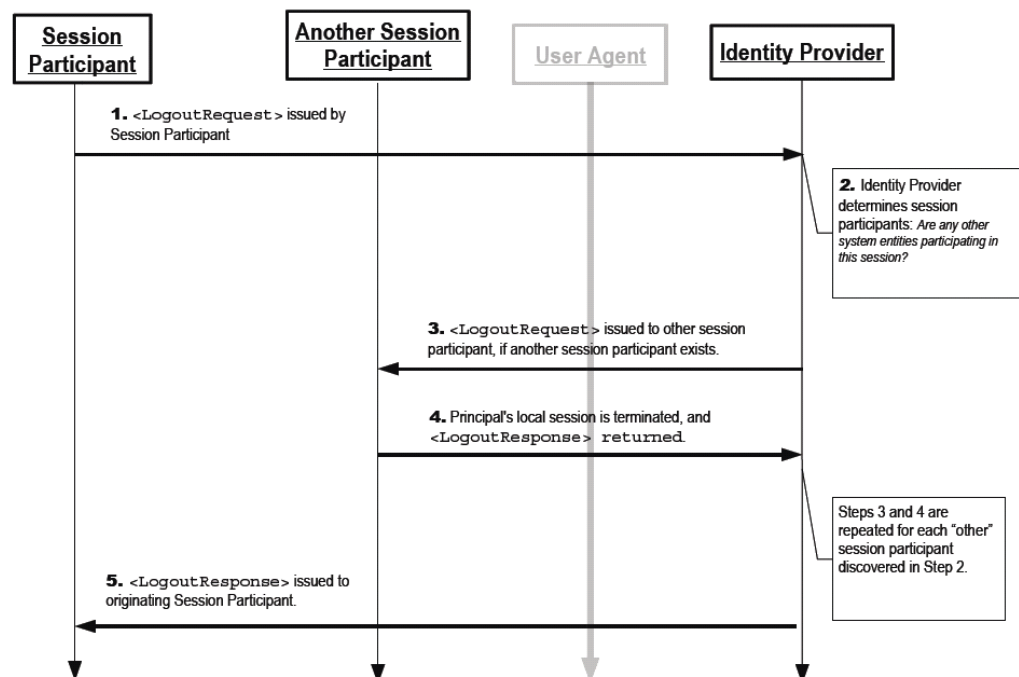
## 6 Single Logout Profile

---

>

SAML 2.0 supports the concept of single logout and describes both a Single Logout Protocol in [SAMLCore] and a Single Logout Profile in [SAMLProf]. These allow Identity- and Service Providers to terminate multiple user sessions by exchanging <LogoutRequest> and <LogoutResponse> messages. In this way, a user can perform near-simultaneous logout to all Service Providers whose session originate from a particular Identity Provider (i.e. "single logout"). The user may either contact a Service Provider or an Identity Provider to initiate the logout.

The figure below from [SAMLProf] shows an example message flow:



**Figure 7: Message flow during Single Logout**

Note: The grayed-out user agent illustrates that the message exchange may pass through the user agent or may be a direct exchange between system entities, depending on the SAML binding used to implement

The possible variations in the OASIS Single Logout Profile pertain to which binding that is used. The choices are SOAP binding, HTTP Redirect, HTTP POST, and Artifact binding. Note that the OASIS profile clearly distinguishes between the first request from Service Provider to Identity Provider (which is strongly recommended to use a front-channel binding) and subsequent message exchanges.

In DK-SAML, the following restrictions must be followed:

- HTTP Redirect binding **MUST** be used for the first request going from a Service Provider to an Identity Provider. This will allow the Identity Provider to determine the user session by e.g. reading browser cookies.
- Either HTTP Redirect or SOAP Binding **MUST** be used for subsequent request/response messages from the Identity Provider to a Service Provider.

---

>

---

- All Service Providers and Identity Provider **MUST** support the HTTP Redirect binding.
- Support for SOAP Binding is *optional* for Service Providers.
- Support for SOAP Binding is *mandatory* for Identity Providers.
- When a Service Provider supports SOAP, SOAP is preferred as it offers numerous advantages including reliability.
- All request and response messages **MUST** be signed.

See the architectural decision in section 13.5 for the detailed background behind these choices.

*DK-SAML adopts the Single Logout Profile with the binding restrictions and signing requirements described above. Conforming Service and Identity Providers are required to support it.*

## **6.1 Local Logout Requirements**

In addition to the Single Logout profile described above, each Service Provider should also offer local logout for stand-alone applications to the user. A local logout means that the user will be logged out of the local Service Provider application only, but will keep any active session with the Identity Provider and other Service Providers.

Note that for Service Providers who are part of a portal, a local logout may not make sense and may be handled as part of the portal framework instead.

---

## 7 Authentication Assertion Profile

---

>

This chapter describes overall requirements for the content of SAML assertions exchanged via the Web SSO profiles. These include rules for encoding attributes and define core attributes that must always be present in an authentication assertion.

Subsequent chapters contain attribute profiles which define additional attributes for specific scenarios including:

- OCES Attribute Profile
- Persistent Pseudonym Attribute Profile

### 7.1 Generic Assertion Requirements

The following section describes generic requirements for assertions which must be followed by all attribute profiles in order to achieve consistency and interoperability. The structure of a generic SAML assertion is illustrated in the figure below:

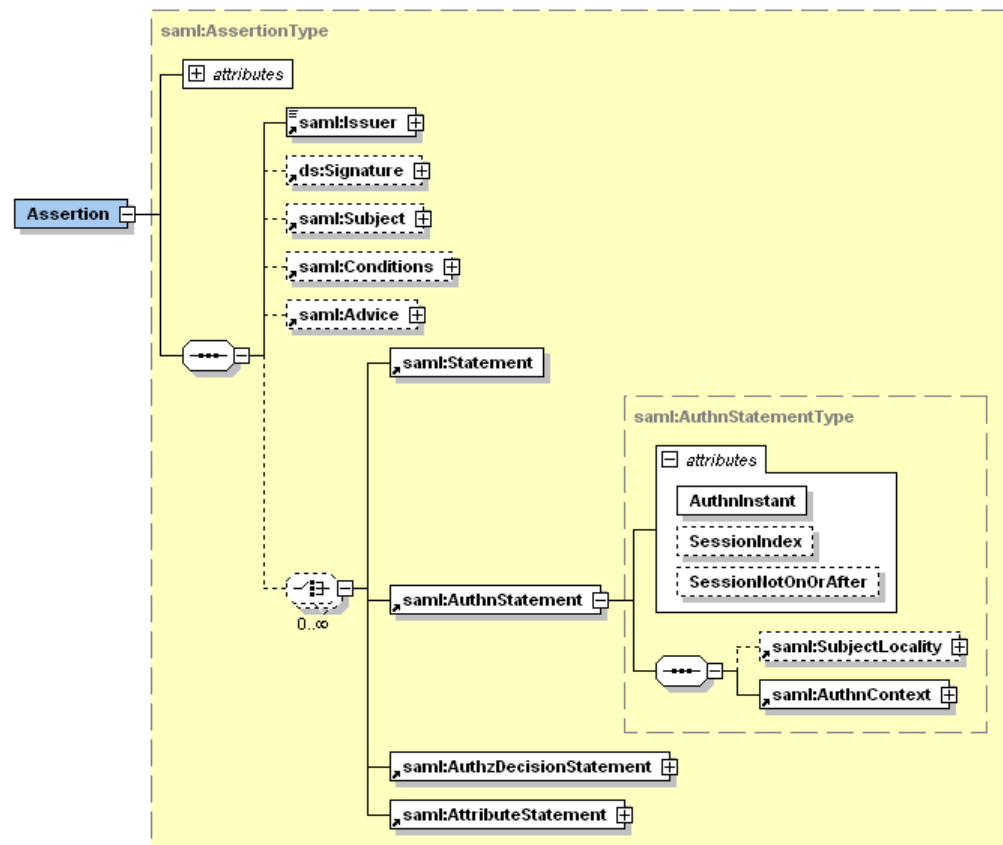


Figure 8: Structure of a SAML Assertion

The following sub-sections describe each of the main elements of the assertion. Since SAML 2.0 provides a great degree of flexibility an important goal of DK-SAML will be to tailor the format to local Danish requirements. This will facilitate consistency and interoperability - and assure that identity attributes needed in the Danish public sector are properly specified.

Note that the <AuthzDecisionStatement> in the above figure is not allowed in the DK-SAML profile. This element is deprecated in SAML 2.0 and is addressed in the

XACML standard instead. Specifically, an <XAMLAuthnDecisionStatement> element is defined as a SAML extension which replaces the current <AuthzDecisionStatement> element.

#### 7.1.1 Main Assertion Element

The assertion must contain exactly one <AuthnStatement> and exactly one <AttributeStatement> element. All other statements are disallowed since they are outside the scope of the profile.

Encryption of assertions is required by this profile via the <EncryptedAssertion> element. Encryption will ensure end-to-end confidentiality when sensitive information is transferred. Encryption must be performed with the recipient's public key bound to an OCES company certificate<sup>6</sup>.

Note the use of encryption requires that a Service Provider has included his OCES company certificate as part of the meta data exchanged with the Identity Provider.

#### 7.1.2 The Issuer Element

The Issuer element is mandatory and MUST contain a string with the (unique) issuer id. In this profile, the issuer id will be a Uniform Resource Locator containing the issuer's domain. See section 11.1 for a further discussion of identifiers in the profile.

The element is of type NameIDType which defines four other attributes (NameQualifier, SPNameQualifier, Format and SPProvidedID). The qualifiers are not needed since the identifiers in this profile are unique per construction. Further, there is no need to indicate special processing rules via a format attribute and affiliation of issuers are not needed here.

Therefore, none of these four attributes are allowed in DK-SAML.

#### 7.1.3 The Signature Element

This element can be used to hold a digital signature over the assertion which provides integrity protection and message authentication.

Normally however, an assertion is contained in a <Response> message which itself is signed – see e.g. section 4.5. In this case, the assertion will inherit the integrity protection and authentication from the outer signature.

The signing rules in DK-SAML are therefore:

- Signing of an Assertion is *optional* if it is embedded in a signed <Response> message.

---

<sup>6</sup> When new OCES Certificate Policies (e.g. device certificates) are published, these will be analyzed to determine whether they can be used for this purpose.

- Signing is *mandatory* if the assertion is not embedded in a signed message.

Furthermore, the private key used for signing must be bound to the Identity Provider's OCES Company certificate or equivalent.

#### 7.1.4 Subject Element

An assertion MUST contain one <Subject> element holding the subject id. Specific attribute profiles define requirements for the subject ID (e.g. for OCES profile it must contain certain fields from the OCES certificate).

Encrypted identifiers are generally disallowed (see section 11.5 on security considerations for a discussion) in order to avoid processing overhead for individual elements.

The subject element must contain at least one <SubjectConfirmation> element containing a Method of `urn:oasis:names:tc:SAML:2.0:cm:bearer`.

The bearer <SubjectConfirmation> element described above MUST contain a <SubjectConfirmationData> element that has a Recipient attribute containing the Service Provider's assertion consumer service URL and a NotOnOrAfter attribute that limits the window during which the assertion can be delivered. It MUST NOT contain a NotBefore attribute.

#### 7.1.5 Conditions Element

The assertion MUST contain an <AudienceRestriction> including the Service Provider's unique identifier as an <Audience>.

#### 7.1.6 Advice Element

There are no profile-specific requirements for this element; it can safely be ignored by Service Providers.

#### 7.1.7 AuthnStatement Element

An assertion MUST contain exactly one element describing authentication of the subject to the Identity Provider.

To support the Single Logout profile, any such authentication statements MUST further include a SessionIndex attribute to enable per-session logout requests by the Service Provider.

When authenticating subjects using an OCES certificate, the <AuthnContext> element SHOULD refer to the following authentication context class in an <AuthContextClassRef> element: `urn:oasis:names:tc:SAML:2.0:ac:classes:X509`.

Note that the <AssuranceLevel> attribute defined in DK-SAML and used in <AttributeStatements> will also provide information about the authentication context. Specifically, it will contain a classification of the authentication strength according to the scheme defined in [ITTAAuthLevel].

### 7.1.8 AttributeStatement Element

This element is a mandatory part of the assertion and will mainly be specified by the attribute profiles contained in subsequent chapters.

The purpose of these attribute profiles is to ensure that different organizations use a common set of attributes to match different accounts for the same user and to provide a consistent naming of attributes. This will simplify integration and exchange of user attributes across organizational boundaries.

*It is allowed to further profile the attribute profiles in this specification in a local context e.g. by adding new attributes in a separate name space.*

For example, it is anticipated that different sectors will need additional attributes which can thus be added provided that the requirements to the “ancestor” profile are still followed.

## 7.2 Attribute Encoding Rules

DK-SAML defines the following rules for attribute encoding:

- The <NameFormat> XML attribute on the <Attribute> element must be: `urn:oasis:names:tc:SAML:2.0:attrname-format:uri`
- Attribute names must be a URI (as indicated by the name format above).
- The <FriendlyName> XML attribute is optional.
- Attributes with an Object Identifier should use this identifier as their name (e.g. “urn:oid:2.3.4.5”).
- Attributes without an Object Identifier which are defined by the National IT and Telecom Agency have the following name prefix: “dk:gov:saml:attribute”.
- All attribute values must be simple text strings with type “xs:string”.
- Optional attributes MAY be set with blank values.

For a detailed rationale behind these choices, see architectural decision 13.8. Most of the restrictions are defined to ensure support in COTS products. Examples can be found in the next section on core attributes.

Implementations SHOULD NOT rely on the FriendlyName XML attribute but instead on the Name attribute.

Encrypted attributes are not permitted (see section 11.5 on security considerations). Instead the entire assertion is encrypted.

## 7.3 Core Attributes

In [ITTAtrib] a set of core attributes are identified which must always be part of a SAML authentication assertion. Thus, attribute profiles defined in subsequent chapters or elsewhere must include this core set. However, if an attribute profile is a pseudonym profile targeted for privacy, the core attributes may of course be excluded (see e.g. the persistent pseudonym profile in chapter 9).

The defined set of (mandatory) core attributes in [ITTAtrib] are:

- sn - Surname
- cn - Common name.
- uid - User id
- mail - email address

In addition, the following attributes are mandatory in DK-SAML:

- AssuranceLevel – States how strongly the user was authenticated (see below).
- SpecVer – States the applied version of the DK-SAML profile (see below).

The following attributes are optional in [ITTAtrib]:

- uniqueAccountKey - Unique key to match and synchronize user information across systems and organisations
- cvrNumberIdentifier - An employee's organization identifier

In the following subsections it will be shown how to encode these attributes according to the rules defined in section 7.2.

See the architectural decision in section 13.9 for the rational behind these choices.

Note: if the value of a mandatory attribute is unknown to the Identity Provider, it MUST be filled with an empty value. Note further that an attribute profile may interpret the value of an attribute in a specific context (e.g. uid) or declare that an optional core attribute is mandatory (e.g. cvrNumberIdentifier).

### 7.3.1 Sur Name Attribute

The Sur Name attribute is encoded via its OID:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.4"
  FriendlyName="surName">
  <saml:AttributeValue xsi:type="xs:string">
    Jensen
  </saml:AttributeValue>
</saml:Attribute>
```

### 7.3.5 Common name Attribute

The Common Name attribute is encoded via its OID:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.3"
  FriendlyName="CommonName">
  <saml:AttributeValue xsi:type="xs:string">
    Hans Jensen
  </saml:AttributeValue>
</saml:Attribute>
```

### 7.3.6 Uid Attribute

The uid attribute specifies the user id in the user's (principal's) home organization (or credential issuing organization where home organization is unknown or doesn't exist – which is the case for citizens).

The actual content of the uid attribute is left to the discretion of the IdP, and should be documented by the IdP.

Note that attribute profiles may specify how this attribute is used in a specific context (e.g. OCES).

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:0.9.2342.19200300.100.1.1">
  <saml:AttributeValue xsi:type="xs:string">
    JMogensen
  </saml:AttributeValue>
</saml:Attribute>
```

### 7.3.7 Email Attribute

The Email attribute is encoded via its OID:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:0.9.2342.19200300.100.1.3"
  FriendlyName="email">
  <saml:AttributeValue xsi:type="xs:string">
    jens@email.dk
  </saml:AttributeValue>
</saml:Attribute>
```

### 7.3.8 Assurance Level Attribute

The AssuranceLevel attribute which provides the Service Provider an indication of how strongly the user was authenticated. The attribute can have the values “1”, “2”, “3”, “4” and “test” and the semantics of the levels is defined in [ITTAAuthLevel].

Below is given an example representation of the assurance level attribute:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="dk:gov:saml:attribute:AssuranceLevel">
  <saml:AttributeValue xsi:type="xs:string">2</saml:AttributeValue>
</saml:Attribute>
```

### 7.3.9 SpecVer Attribute

The SpecVer attribute tells the Service Provider which version of the DK-SAML profile the assertion was issued under. The current value is “DK-SAML-2.0”. This

---

>

---

makes it easier to change the profile in the future without hurting backwards compatibility.

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="dk:gov:saml:attribute:SpecVer">
  <saml:AttributeValue xsi:type="xs:string">
    DK-SAML-2.0
  </saml:AttributeValue>
</saml:Attribute>
```

Note that the version number is not in any way connected to the OASIS SAML version number.

#### 7.3.10 cvrNumberIdentifier Attribute (Optional)

The cvrNumberIdentifier Attribute is used to represent the organization where the subject is employed:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="dk:gov:saml:attribute:CvrNumberIdentifier">
  <saml:AttributeValue xsi:type="xs:string">
    20688092
  </saml:AttributeValue>
</saml:Attribute>
```

#### 7.3.11 uniqueAccountKey Attribute (Optional)

The uniqueAccountKey Attribute contains an account ID that is unique across organizations:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="dk:gov:saml:attribute:UniqueAccountKey">
  <saml:AttributeValue xsi:type="xs:string">
    xri://@DK-XRI*19-43-70-19/Borger*($d/2005-08-02T16:16:42+01:00Z)/OJEN
  </saml:AttributeValue>
</saml:Attribute>
```

The attribute value should follow the recommendations in [ITTUID].

### 7.4 Sector-specific attributes

It is anticipated that different sectors and perhaps even individual Identity Providers may need to specify their own attributes.

In order to avoid conflicts with attributes in other sectors (and this specification) the following rules must be followed:

- Sector or IdP-specific attributes must be placed in responses to attribute queries and not in authentication assertions. This ensures that a user can be logged in from any IdP and still access all Service Providers in the federation.

---

>

---

If an IdP introduced sector-specific attributes in the authentication assertions and a SP (within the same sector) relied on these for logon this would not work seamlessly.

- Attributes specific to a sector (e.g. the health care sector) or an Identity Provider must use a name URI containing their DNS domain.
- Sector-specific attributes must follow the encoding rules described in section 7.2.

---

## 8 OCES Attribute Profile

>

---

This chapter describes an attribute profile which transfers identity attributes available after OCES digital signature authentication. This includes fields from the OCES certificate such as distinguished name, PID, CVR and RID numbers plus (optionally) a CPR number which can be resolved from OCES citizen certificates (and some employee certificates) by Government authorities.

The profile facilitates easy identification of the user by a Service Provider who internally use OCES attributes in their existing registries and applications (the CPR number most likely). In other words, federation occurs dynamically via identity attributes and not by an explicit account linking process.

While this scheme provides simple and efficient integration in practice, it is also important to consider the following:

- Since the account linking process is not explicit, the user may not be able to control it.
- If all Service Providers organize user data using the same key attributes (e.g. CPR numbers) it may in theory be easier to (illegally) correlate information across organizational boundaries with loss of privacy as a consequence.

If these concerns are paramount, the persistent pseudonym attribute profile described in chapter 9 should be used instead.

In the following, a set of attributes and their associated representations are described which is either a mandatory or optional part of the profile.

### 8.1.1 Requirements for the Subject Element

In the OCES Attribute Profile the user is identified primarily via attributes (e.g. CPR, CVR and PID numbers) and less via the subject element in the assertion. Some SAML products may however require a valid subject element.

The SAML Deployment Profile Draft for X.509 Subjects [SAMLDepl] recommends using the Distinguished Name (DN) from the certificate in the Subject. This convention is followed in the OCES profile as shown below:

```
<saml:Subject>
  <saml:NameID
    Format="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName">
    C=DK,O=Pølsevognen,CN=Hans Jensen
  </saml:NameID>
  <saml:SubjectConfirmation
    Method="urn:oasis:names:tc:SAML:2.0:cm:bearer">
    <saml:SubjectConfirmationData
      Recipient="http://SomeServiceProvider.dk"
      NotOnOrAfter="2001-12-31T12:00:00"
      InResponseTo="Authn_request_identifier_1234567">
    </saml:SubjectConfirmationData>
    </saml:SubjectConfirmation>
  </saml:Subject>
```

### 8.1.2 Certificate Serial Number (Mandatory)

This attribute holds the certificate serial number which is not to be confused with the subject serial number (holding PID, RID and CVR numbers). The certificate serial

---

>

---

number identifies a certificate uniquely within a given CA and is encoded as shown below:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.5" FriendlyName="serialNumber">
  <saml:AttributeValue xsi:type="xs:string">
    234-2345-76745-23
  </saml:AttributeValue>
</saml:Attribute>
```

The certificate serial number can be used by a Service Provider to:

- perform revocation checks with the CA
- check whether a certificate used for signing was the same certificate used for login

### 8.1.3 Organization Name (Mandatory for Employees / Companies)

This attribute is mandatory for companies and employees and contains the name of the organization:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.10" FriendlyName="organizationName">
  <saml:AttributeValue xsi:type="xs:string">
    Pelles Pølsefabrik
  </saml:AttributeValue>
</saml:Attribute>
```

### 8.1.5 Organization Unit (Optional)

This optional attribute contains the name of the department within an organization:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.11" FriendlyName="organizationUnit">
  <saml:AttributeValue xsi:type="xs:string">
    Kvalitetsafdelingen
  </saml:AttributeValue>
</saml:Attribute>
```

### 8.1.6 Title (Optional)

As the name indicates, this attribute holds the title of an employee:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.12" FriendlyName="title">
  <saml:AttributeValue xsi:type="xs:string">
    Chefkontrollant
  </saml:AttributeValue>
</saml:Attribute>
```

### 8.1.7 Postal Address (Optional)

The optional postal address contains the address where a company or person is registered:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.16" FriendlyName="postalAddress">
  <saml:AttributeValue xsi:type="xs:string">
    Kvægtorvet 5, 2150 Kødbyen
  </saml:AttributeValue>
</saml:Attribute>
```

### 8.1.8 OCES Pseudonym (Optional)

A person or employee may have a pseudonym associated with their certificate:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:2.5.4.65" FriendlyName="pseudonym">
  <saml:AttributeValue xsi:type="xs:string">
    mister x
  </saml:AttributeValue>
</saml:Attribute>
```

Note: this pseudonym refers to a field in the OCES certificate and is not to be confused with pseudonyms used in the SAML protocols to establish federation of user identities.

### 8.1.9 User Certificate (Optional)

In some cases an Identity Provider may want to deliver the user's entire OCES certificate to the Service Provider. Here the below SAML attributes can be used. The attribute value must be a base64 encoded string representing the DER encoded X.509 certificate:

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="urn:oid:1.3.6.1.4.1.1466.115.121.1.8"
  FriendlyName="userCertificate">
  <saml:AttributeValue xsi:type="xs:string">
    MIIB5DCCAUI0CBAJQodoZIhvcNAQ...
  </saml:AttributeValue>
</saml:Attribute>
```

Delivering entire certificates to a Service Provider in an assertion will however result in additional processing overhead and assertion / message footprint.

#### 8.1.10 PID Number Attribute (Mandatory for Persons)

For OCES person certificates, the most interesting attribute is the PID number which contains a unique identifier for the person<sup>7</sup>. The advantage of PID numbers over CPR numbers is that they can be freely exchanged without risk of violating personal data protection acts.

A Service Provider receiving a PID number can subsequently ask the user for his CPR number and validate the PID-CPR correspondence by contacting the Certificate Authority. Alternatively, if the Service Provider is a Government institution with authority to look up CPR numbers it can be done directly without user interaction. With this scheme, the Identity Provider is thus able to transfer the CPR number indirectly. The CPR number is generally a very useful attribute since many systems use it as identifier or primary key.

The PID number is mandatory if the user has authenticated using a person certificate and should be encoded according to the following example (syntax and semantics of the number itself is defined in [OCESPers] and DS843-1):

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="dk:gov:saml:attribute:PidNumberIdentifier">
  <saml:AttributeValue xsi:type="xs:string">
    9802-2002-2-9142544
  </saml:AttributeValue>
</saml:Attribute>
```

#### 8.1.11 CPR Number Attribute (Optional)

In some scenarios it may be easier to transfer the CPR number directly in the assertion. The CPR number attribute is optional and must only be included when:

- A formal agreement has been made to exchange it
- The Service Provider is authorized to receive it (e.g. is a Government entity)
- The surrounding assertion is encrypted (which is mandatory in this profile)

An Identity Provider must have the technical capability to resolve and insert the CPR number both for citizens and employees who have one<sup>8</sup>. The CPR number attribute is however optional such that it can be omitted from assertions for Service Providers who do not need it / are not allowed receiving it.

When used, the CPR number should be represented according to the following example:

<sup>7</sup> The Subject Serialnumber in OCES person certificates can be constructed by prefixing the number with "PID:"

<sup>8</sup> Some employee certificates are associated with a CPR number; this is e.g. used in the health care sector where there is often a need to know the CPR number of a health care professional.

>

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="dk:gov:saml:attribute:CprNumberIdentifier">
  <saml:AttributeValue xsi:type="xs:string">
    2702681273
  </saml:AttributeValue>
</saml:Attribute>
```

#### 8.1.12 CVR Number (Mandatory for Employees and Companies)

This attribute is mandatory when the user has authenticated with company or employee certificates.

Note that the attribute is part of the core set of attributes defined in section 7.3.

#### 8.1.13 Employee Number / RID (Mandatory for Employees)

This attribute is mandatory when the user has authenticated with an employee certificate and should be encoded according to the following example (syntax and semantics of the number is defined in DS844):

```
<saml:Attribute
  NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
  Name="dk:gov:saml:attribute:RidNumberIdentifier">
  <saml:AttributeValue xsi:type="xs:string">
    2342-345623423
  </saml:AttributeValue>
</saml:Attribute>
```

The Subject Serial Number for OCES Employee certificates can be constructed from the CVR and RID numbers by a simple string concatenation (e.g. CVR:20688092-RID:1180636224562).

#### 8.1.14 Uid Core Attribute

Section 7.3 defines a set of core attribute that must always be included in an authentication assertion.

In the OCES attribute profile, the following conventions apply for the uid attribute:

- The uid attribute must contain the Subject Serial number from the OCES certificate. The field from the certificate is included literally.

This means that the PID and RID numbers will be present twice in the assertion, but this may be convenient:

- If the Service Provider needs a unique ID within the credential issuing organization or he needs the Subject Serial Number he may simply pick the uid attribute.
- If the Service Provider wants to know whether the Subject is a person or employee or needs the RID/PID/CPR/CVR numbers, he can pick the corresponding (atomic) attributes without having to parse the serial number string.

---

## 9 Persistent Pseudonym Attribute Profile

---

>

While the OCES attribute profile facilitates smooth integration between Identity Providers and Service Providers without explicit account linking, it implies that Service Providers organize their internal user registries to use the OCES attributes (e.g. CPR numbers). While most government organizations probably do this today, the architecture should not mandate this.

In order to support enhanced privacy requirements, it must be possible for Service Providers to avoid using CPR or PID numbers in their user registries. This will make it more difficult to correlate user identities across different Service Provider organizations.

Therefore, this attribute profile has been defined to support of federation using persistent pseudonym identifiers. A pseudonym identifier is in effect a random value that a IdP-SP pair establish and use to refer to the same user; each maintain a mapping from the shared identifier to their internal representation. The goal of this attribute profile is to define the content of assertions and attributes supporting this scenario.

### 9.1 Rolling Migration

In addition to privacy goals, the profile also allows rolling migration from scenarios where a Service Provider has established a local user id which cannot be inferred from the SAML assertion sent by an Identity Provider. Here, the pseudonym can be used as a link from the federated identity to the local identity.

This will often be the case when a Service Provider is replacing an existing local logon system with a federated solution using an external Identity Provider.

### 9.2 Profile Requirements

The requirements for this attribute profile are simply:

- The only kernel attributes to be included in the assertion are:
  - AssuranceLevel attribute
  - SpecVer attribute
- No other attributes are included which reveals the user's (external) identity.
- The assertion Subject element contains a persistent pseudonym identifier. The identifier must be truly opaque such that the user identity cannot be inferred from it. The pseudonym identifier is shared between Identity Provider and Service Provider and is established during the very first interaction between these. On subsequent interactions, the pseudonym is re-used.

Below an example of a subject element containing an opaque name identifier is given:

```
<saml:Subject>
  <NameID Format="urn:oasis:names:tc:SAML:2.0:nameid-format:persistent">
    005a06e0-ad82-110d-a556-004005b13a2b
  </NameID>
</saml:Subject>
```

Both Identity Provider and Service Provider need to store the pseudonym and the mapping to the corresponding internal user identity for future references.

---

## 10 Attribute Service Profile

>

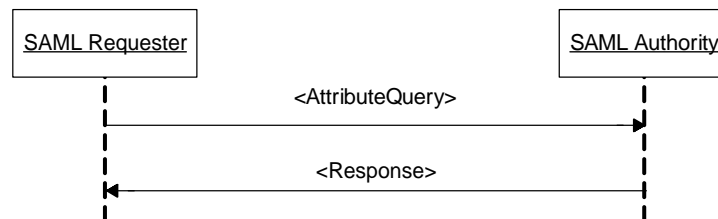
---

This chapter specifies an attribute service profile for querying and returning identity attributes from an Attribute Service. It is used in scenarios where a Service Provider after the initial authentication of the user needs further information to e.g. grant access to a resource or personalize an application front-end.

### 10.1 Profile Overview

This profile is a specialization of the "Assertion Query/Request Profile" described in [SAMLProf] which again is based on the "Assertion Query and Request Protocol" defined in [SAMLCore]. Where nothing else is specified, this profile inherits messages, processing rules and other properties of the "Assertion Query/Request Profile".

The messages exchanged in the profile are illustrated below:



**Figure 9: Basic Message Exchange**

The steps are:

1. The SAML Requester (e.g. a Service Provider) sends an `<AttributeQuery>` message as defined in [SAMLCore]. None of the other types of request elements defined in the SAML Assertion Query and Request Protocol are allowed in this profile.
2. The SAML Authority (an Attribute Service) returns a `<Response>` message containing an `<Assertions>` with an `<AttributeStatement>` element.

### 10.2 Requirements for Request/Response Messages

#### 10.2.1 The `<AttributeQuery>` Message

This attribute profile has the following requirements for the request message:

- The `Consent` attribute is mandatory.
- The `<Issuer>` element is mandatory.
- The `<ds:Signature>` element is mandatory and the query **MUST** be signed with a key bound to the requester's OCES company certificate or equivalent.
- It is recommended that the Service Provider further identifies the Subject by including the `uid` core attribute (with attribute value) in the request (see section 7.3.2 for details on this attribute).

#### 10.2.2 The `<Response>` Message

The attribute profile has the following requirements for the response message:

---

>

---

- The <Issuer> element is mandatory.
- The <ds:Signature> element is mandatory and the response MUST be signed with a key bound to the responder's OCES company certificate (or equivalent).
- If the <Subject> element is present in the query, the <Subject> element MUST also be present in the response and match the request.

Any assertion(s) in the response MUST comply with the requirements for authentication assertions stated in chapter 7 with the following exceptions:

- The Assertion MUST not carry an <AuthnStatement> element.
- The <SubjectConfirmation> element in the assertion is optional.
- The assertion does not have to include the kernel attributes; instead the attribute requested in the query are returned.

### 10.3 Processing Rules

Some error situations do not seem to be covered by the SAML specifications. Differences in error handling may lead to non-interoperable implementations and the recommended behavior is therefore detailed below.

The error situations which appear to be unspecified by SAML are:

- a) The subject specified in the request is not recognized by the Attribute Service.
- b) Attributes are requested which the Attribute Service does not recognize.
- c) Attributes are requested which the Attribute Service does not want to disclose to the requestor according to its attribute release policy<sup>9</sup>.
- d) A known attribute is requested, but the Attribute Service does not know the attribute value for this particular subject.

In case a) it is recommended to return a second-level status code with the following URI reference:

- `urn:oasis:names:tc:SAML:2.0:status:UnknownPrincipal`

In case b) it is recommended to use the following approach:

- The top-level error code is set to "Success" if any of the requested attributes can be returned; otherwise it is set to `urn:oasis:names:tc:SAML:2.0:status:Requester`.
- An assertion is returned with all known attributes (provided it is allowed by the attribute release policy).

---

<sup>9</sup> SAML lacks the concept of "Attribute Release Policy". Such a concept is part of the Identity Governance Framework which currently is being standardized by Liberty Alliance, and it will be considered once standardized.

- A nested status code element is included specifying a status code being `urn:oasis:names:tc:SAML:2.0:status:InvalidAttrNameOrValue`
- A sequence of `<StatusDetail>` elements are included, one per unknown attribute, specifying the name of the unknown attribute to the requester.

In case c), return a second-level status code being:

`urn:oasis:names:tc:SAML:2.0:status:RequestDenied` followed by a sequence `<StatusDetail>` elements describing the reason for not disclosing the attribute.

In case d) there is no meaningful SAML second-level error code and one can further discuss whether this situation is an error at all. To achieve consistency across implementations, it is recommended to return an `<Attribute>` element in the response with the corresponding `<AttributeValue>` element empty and with the reserved attribute `xsi:nil` with a value of “true” or “1” (see [SAMLCore] p. 31).

Since status codes are generally URI references it is easy for Attribute Services to invent their own and thereby create interoperability issues. Therefore, it is recommended to only use status code URIs defined in [SAMLCore] or optionally (if the need appears) specify additional status codes through the OIO standardization initiative.

### 10.3.1 Identifying the Subject

The Attribute Service must identify the Subject based on the information in the request. For this purpose the SAML Subject is included. In some situations however, this information is not enough. For the OCES attribute profile for example, the subject contains the Distinguished Name (DN) of the Subject which is not sufficient for unique identification.

In these cases the profile recommends that the requester also includes the uid core attribute in the request (including the attribute value) such that the Attribute Service can identify the user.

## 10.4 Attribute Naming and Encoding

Generally, attribute names and encoding should follow the rules stated in section 7.2. No other attributes are specified in this profile.

## 10.5 Meta Data

An Attribute Service should declare as part of its meta data which attributes it understands (as specified in [SAMLMeta]). Note that this is not the same as an attribute release policy which cannot be defined within the context of the SAML framework alone. Attribute release policies are therefore not in scope for this profile.

## 10.6 Discovery

This profile does not specify any mechanisms for discovery of Attribute Services. Generally, the discovery mechanism present in SAML 2.0 does not cover Attribute

Services but is instead targeted SSO. Service Providers must know the location at relevant attribute services through out-of-band discovery<sup>10</sup>.

## 10.7 Binding

Use the SOAP Binding.

## 10.8 Privacy

Before requesting private or personal data<sup>11</sup> from an Attribute Service, the Service Provider MUST prompt the user for her consent or in other ways be able to prove having consent to request the data. The `Consent` attribute MUST be included in the query to accurately reflect the collected consent and the request MUST be digitally signed by the Service Provider. For publicly available data consent is however not required.

Danish legislation (including `persondataloven`, `registerloven`, `forvaltningsloven`) must be followed when dealing with personal data. An Attribute Service MUST thoroughly investigate legal obligations before attributes are released.

Furthermore, an Attribute Service MUST audit log all situations where private data is released so it is capable of accurately stating which data has transferred to whom and when it has happened, and what type of consent was given by the user. It MAY choose to notify the user when attributes are released but this is not required.

All communication containing sensitive data MUST be strongly encrypted (according to the rules specified by Datatilsynet) to avoid disclosure of sensitive data in transit (see security section below).

## 10.9 Security

The SAML `<AttributeQuery>` and `<Response>` messages MUST be digitally signed by signature keys bound to the sender's OCES Company Certificates (or equivalent).

Any returned assertions MUST be encrypted and signed according to DK-SAML. As with authentication assertions, since the `<Response>` message is signed, the contained assertion does not have to be signed also.

---

<sup>10</sup> Future profiling of id-based web services may include a discovery service that holds information about the individual users attribute stores, but that is beyond the scope of the current profile.

<sup>11</sup> For definitions of these terms, please consult the Danish law "Lov om behandling af personoplysninger", chapter 2.

The communication between requester and responder **MUST** be strongly encrypted and integrity protected using at least one of the following mechanisms:

- SOAP security using the WS-\* protocols (signing and encryption).
- SSL / TLS transport security with mandatory client authentication.

Where both options are available, the recommended choice is to use SOAP security.

If SOAP security is used, the underlying X.509 certificates **MUST** be the OCES company certificates of the Service Provider and Attribute Service. Furthermore, security headers should comply with the WS-Security specification.

If SSL / TLS security is used, the client certificate **MUST** be the OCES company certificate of the Service Provider (the server certificate **MUST** be an SSL certificate and therefore cannot be an OCES certificate). Use of SSL / TLS should be compliant with OWSA Model T defined by the IT and Telecom Agency.

SOAP stacks and SSL Cipher Suites **MUST** be configured to avoid weak encryption.

---

## 11 Profile Considerations

>

---

This chapter describes a number of common considerations for the different profiles described in this document.

### 11.1 Naming and Identifiers

In various SAML elements there is a need for expressing unique identifiers representing Service and Identity Providers. In order to ensure uniqueness without central management it has been decided to use URL references containing (unique) domain names as identifiers:

- <http://loginservice.dk>
- <http://serviceprovider.dk/x/y/z>
- <http://someportal.dk/samlsp>

### 11.2 Assertion ID as Transaction Identifier

A SAML assertion is always required to contain an ID attribute which is unique (to an extremely high probability), see [SAMLCore]. This identifier is thus suitable as a transaction identifier that allows correlation of events across Service Providers and Identity Providers.

Service and Identity Providers are therefore required to use this ID in their internal log files such that all logged events relevant to a given SSO session can be tracked.

### 11.3 Meta Data

All entities supporting the DK-SAML profiles must support the SAML Meta Data specification [SAMLMeta].

Additional requirements to meta data in this profile are:

- All entities must be able to export and import meta data files.
- All entities must include their OCES company certificates literally (i.e. not just references) in order to allow others to verify signatures from *them* and encrypt messages *to* them.
- All relevant services required by this profile must be described in meta data, including SingleLogonService, SingleLogoutService, AttributeService, AssertionConsumerService, ManageNameIDService, NameIDMappingService. Requirements for the Attribute Service are defined in [IdpReq].
- All attributes supported by the Attribute Service should be described in the <AttributeAuthorityDescriptor>. Please note that if an attribute is mentioned, this does not imply that a Service Provider can or will receive it.
- All entity identifiers must conform to the requirements of section 11.1.
- No proprietary information may be included in the SAML meta data (e.g. in <Extensions> elements) including required / supported levels of authentication. This is to ensure that meta data can be exchanged without interoperability issues.

- The root of every metadata file must be <EntityDescriptor>.

Signing and verification of meta data is *not* required by this profile – see architectural decision 13.3. However, Service and Identity Providers must ensure that meta data is authentic and has not been modified before using it.

#### 11.3.1 Exchanging meta data

This profile does not mandate any particular mechanism for exchanging meta data (out of band).

Publication of meta data locations in DNS records is left optional.

### 11.4 Privacy in a Danish Context

The privacy issues take their starting point in any user's right to be ensured that private or personal data is treated in accordance with Danish and International Privacy Legislation.

Private or personal data is e.g. data about racial or ethnical background, political, religious or philosophical beliefs, union membership, or data about health or sexual affairs. A CPR-number is as such considered private information and must accordingly be treated with special care.

The following will not be a thorough discussion of all the necessary precautions but the most important will be described. They fall in 3 sections:

- Registration of private or personal data
- Information and choices that must be given to the user
- Transfer of private or personal data such as attributes about the user between Identity Provider and Service Provider.

The Service Provider should always follow the procedures necessary due to the nature of the service provided.

#### 11.4.1 Registration and use of private or personal data

Generally it is not allowed to register any private or personal data about a physical person unless there is a law that says it is allowed or the party making the registration has a genuine need to register the data in order to carry out otherwise legal business.

However, if the person whose data is to be registered gives his or hers explicit consent to the registration and use of the data, the data can be registered, used and transferred to third parties only for the consented purpose and with sufficient security precautions. Security precautions are stated in [Sikkerhedsbekendtgørelsen].

The Identity Provider therefore has to get the users consent to the registration and use of any private or personal. If the Service Providers register and use private or personal data about the user, each Service Provider has to explicitly get the user's consent. If the Identity Provider or Service Provider is a Public Authority special restrictions may apply.

Registration and use of private or personal data must be reported to Datatilsynet prior to the first release of the service.

Note: Publicly available data such as common name and PID are not considered private or personal.

#### **11.4.2 Information and choices that must be given to the user**

The user will by default opt-in to Single Sign-on, but the Identity Provider must provide a mechanism by which the user can opt-out. If the user has opted-out the Identity Provider must remember this and give the user the possibility to opt-in at a later point in time.

The Identity Provider must inform the user which data is collected about the user and the purpose of collecting the data.

If private or personal data about the user is about to be transferred to any third party, the Identity Provider must inform the user hereof and give the user the possibility to abort the action.

The Identity Provider can choose to inform the user of which data regarding the user is transferred to which Service Providers in a general consent-form. If no user-consent is given the data must not be transferred to a third party.

The user is to be informed of the consequences of the choices made regarding transfer of private and personal data, e.g. that the user will not be able to be verified at a sufficient security level to access the service chosen.

#### **11.4.3 Transfer of private or personal data between Identity Provider and Service Provider.**

The Identity provider has the responsibility to ensure that attributes regarding a user can lawfully be transferred to the Service Providers.

If the data is publicly available the attributes can be transferred, when the user is informed of the transferral either by the initial agreement with the Identity Provider or in a general text at the Identity Provider's site.

If however the data is personal or private - like which occupation the user has or the user's CPR-number - consent to the registration and transfer of the attribute is required from the user.

Furthermore, special care must be taken regarding security when these types of attributes are exchanged. Not only must the user give his consent for collection and exchange of his private attributes to each individual service provider, strong encryption must also be used to secure the data in transit.

[Sikkerhedsbekendtgørelsen] gives instructions for other security measures to be applied.

### **11.5 Security Considerations and Requirements**

This section contains a number of security considerations and requirements for the SAML profiles.

The security of the entire solution will generally not be better than the security of the authentication mechanism used by the Identity Provider to authenticate the end-user (which is outside the scope of SAML). However, use of the AssuranceLevel attribute means that compromise of weak authentication methods or credentials (e.g. a user loses a static password) will only have limited effect.

### 11.5.1 Transport Level Security

DK-SAML leverages security mechanisms from the HTTPs transport bindings in order to ensure authentication, confidentiality and integrity of in-transit protocol messages and assertions. This is in conformance with the requirements in OWSA Model T (see [http://www.oio.dk/files/Model\\_T\\_Godkendt.pdf](http://www.oio.dk/files/Model_T_Godkendt.pdf)).

More specifically, the following requirements exist for transport level security:

- The HTTP connection used for the POST and Redirect bindings must be secured with SSL 3.0 / TLS 1.0. The connection is not required to use client authentication since that would mean that the end-user would have to authenticate server traffic. Instead, messages transported via this channel will be digitally signed.
- Only SSL / TLS cipher suites providing strong encryption are allowed.
- The SSL certificates must be trusted by commercially available browsers including Internet Explorer, Mozilla, Firefox, Safari and Opera.

The use of SSL / TLS requires that trust mechanisms are established between the communicating entities. Typically, this is done by requiring each entity to maintain a store of trusted peer certificates and/or trusted CA certificates. Secure connections MUST only be allowed from parties who own a private key whose public key can be validated with this store; i.e. a certificate path to a trusted certificate can be established.

It is outside the scope of this profile to specify how these trust mechanisms are set up.

### 11.5.2 Signing and Encryption of SAML elements

Security mechanisms are built into SAML elements themselves and they can thus be independent of transport / binding security mechanisms. The main security mechanisms applicable to SAML elements are XML encryption and XML digital signing.

Digital signing of an entire assertion and request / response protocol messages is possible via the <ds:Signature> element. The advantage over transport-based mechanisms is that the message will be integrity-protected end-to-end (beyond the point where the SSL session is terminated) and that the protection will out-live any SSL sessions. Signing assertions and messages will also allow the recipient to store them as evidence – e.g. should an Identity Provider later repudiate having issued an assertion.

Since the front channel bindings are used, it is generally mandatory to sign assertions and protocol messages with a key bound to an OCES company certificate<sup>12</sup>. Note that an assertion does not have to be signed if it is embedded in a signed <Response> message.

---

<sup>12</sup> When new OCES Certificate Policies (e.g. device certificates) are published, these will be analyzed to determine whether they can be used for this purpose.

SAML 2.0 leverages XML encryption both whole assertions (<saml:EncryptedAssertion>), attributes (<saml:EncryptedAttribute>), and identifiers (e.g. <saml:EncryptedID>).

Encryption of entire assertions is mandatory in this profile. Regarding encryption of individual attributes or identifiers, these more advanced security mechanisms are really not needed in this profile, and they are therefore not recommended for the sake of simplicity.

### 11.5.3 Verification of Signatures

A recipient must verify signed messages including performing a revocation check on the certificate via one of the following methods:

- CDP Extensions – can be used when the certificate includes a Certificate Revocation List Distribution Point extension.
- OCSP – can be used to perform an on-line certificate status check.
- CRL – a certificate revocation list can be downloaded from the CA periodically.

All three mechanisms are available for OCES certificates; OCSP provides the best security characteristics since it always provides an up-to-date answer on the revocation status.

Furthermore, the certificate must be trust-validated to ensure that it has been issued by a trusted CA and that the certificate path is well-formed.

### 11.5.4 Minimum Required Algorithms

The following are the minimum required algorithms which must be supported by all Identity and Service Providers:

- Encryption algorithm must be AES with at least 128 bit keys.
- Signature algorithm must be SHA1withRSA or SHA256withRSA with minimum 1024 bit modulus.

Thus, it is allowed to use AES or RSA with longer keys than specified above. All DES-variants and MD5 hashing are forbidden.

When using 1024 bit RSA modulus, federation participants should prepare to upgrade a longer modulus within 6-24 months.

### 11.5.5 Other Security Mechanisms

There exist a number of additional security mechanisms besides encryption and signing which are to be used by the profile. These are intended to ensure that assertions are not misused (e.g. towards a wrong Service Provider):

- The <SubjectConfirmationData> element of the assertion contains a Recipient attribute referring the Service Provider. This ensures that an assertion can only be used at the Service Provider for which it was intended.
- It further contains a NotOnOrAfter attribute (which is mandatory) that limits the window during which the assertion can be delivered. Thus a stolen assertion could only be used within a small time window (less than 15 minutes).

- The <AuthnStatement> element MAY include a <SubjectLocality> element to specify the DNS domain and IP address for the system from which the subject was apparently authenticated. This will prevent stolen session cookies to be used by an attacker.
- The <Conditions> element MUST contain an <AudienceRestriction> referring to the Service Provider's id. Again this prevents use of the assertion at a wrong Service Provider.

Note that a Service Provider must enforce a one-time semantics for assertions to ensure that an assertion cannot be re-played (e.g. by saving the assertion's identifier).

#### 11.5.6 Analysis of Risks Associated with POST Binding

When the HTTP POST binding is used, the assertion from the Identity Provider to the Service Provider is sent in two steps via the user's browser:

1. The Identity Provider sends an HTML page to the user's browser which contains embedded Java Script, an URL to the Service Provider and the assertion embedded in the page (typically as a hidden form variable).
2. When the page is processed by the browser, the Java Script will launch and submit (via HTTP POST) the SAML assertion to the Service Provider.

The advantage of the POST binding is that there is no direct communication between the Identity Provider and Service Provider. This means that the technical configuration (SSL, firewalls etc) is simple and performance potentially better (depending on the user's Internet connection).

The immediate disadvantage of this binding is that the end user's computer may be easily be compromised by virus, trojan horses etc. and this is further not in control of the federation. This implies a risk of hostile code eavesdropping, modifying or fabricating data transported via this channel.

These risks can be effectively countered by well-known mechanisms including

- Digitally signing SAML assertions and protocol messages
- Encrypting assertions with the Service Provider's public key

These mechanisms are built into SAML 2.0 and achieve confidentiality, integrity, authenticity and non-repudiation of the communication. A fundamental assumption is of course that strong encryption, signing and hashing algorithms with proper key lengths are used.

It is further important to note that XML encryption of assertions from Identity Providers to Service Providers will result in true end-to-end confidentiality, such that data never appears in clear text during transport (e.g. when SSL is terminated). The only threats to SAML assertions during transport are therefore:

- Encryption is broken which is highly unlikely when strong encryption is used.
- Encryption or signing keys are compromised so they can be used by an attacker. This can (and should) be countered by exercising strict access control and procedures etc. for these keys in the IdP and SP organizations. Strong protection of keys can be achieved by generating and storing the keys in tamper-proof hardware - although this is not required.

In general, if encryption keys can be compromised, all types of communication channels will be insecure (including SOAP / WS-Security, SSL / TLS) so this is not a problem specific to the HTTP Post binding.

It must therefore be concluded that usage of digital signing and strong encryption can ensure that the user's browser does not pose any risk for compromise of data in SAML assertions.

This leaves the request / response protocol messages between Service Provider and Identity Provider to be considered:

- Request messages (<AuthnRequest>) are required to be signed by the Service Provider. They will further be encrypted during transport via SSL / TLS but appear in clear form on the user's computer because of the front-channel binding. This means that there is a risk of eavesdropping on the content of the request message at this point (but no modification is possible due to the signature). This is however not an important issue because the request message does not carry any sensitive data. Furthermore, the fact that the user is accessing a given application at a given Service Provider would be evident anyway if the user's computer is compromised.
- Response messages are also required to be signed by this profile and confidentiality is realized at the transport level via SSL / TLS. Again, they contain no sensitive data except the assertion payload which is heavily secured.

#### 11.5.7 Securing Session Cookies

With the security mechanisms described above, the most vulnerable point in the SSO architecture is probably the session cookie established by the Identity Provider. Should an attacker be able to steal this cookie he may attempt to sign on to services at or below the given assurance level until the session times out.

All session cookies must be transient to avoid persistent storage by the browser. The architecture therefore relies on the browser to protect the session cookie established by the Identity Provider.

There are however additional steps which can be taken to greatly mitigate such attacks:

- An Identity Provider should check that all SSO requests bound to a particular session cookie originate from the same client IP address. This will (in most cases) prevent an attacker from using a stolen cookie at another system. In fact, the attacker would have to fake the IP address as well.
- Use of the <SubjectLocality> attribute has a similar effect but the check occurs at the Service Provider side. It MUST be checked by the Service Provider if present.
- A Service Provider can force a fresh re-authentication before access is granted to critical applications. This is done by setting a parameter in the <AuthnRequest> message to the Identity Provider.

## 11.6 Error Handling

In the previous version of DK-SAML errors were handled by the Authentication Portal which provided an abstraction layer on top of the federation technology (SAML). Since the portal component has been removed from the new architecture errors must instead be handled via the mechanisms specified in SAML 2.0 and in some cases by transport level mechanisms (e.g. HTTP error codes, SOAP faults).

The primary way of communicating errors in SAML 2.0 is the <Status> element present in response message. A considerable number of status codes have been defined in [SAMLCore], and additional status messages and details can be included to inform the requester of the problem.

It is recommended that rich error information is returned (when products can be configured to provide it) to facilitate debugging of problems.

---

## 12 Guidance on determining product compliance

---

>

This non-normative chapter discusses how to determine whether a given product is compliant to the Danish SAML 2.0 federation profile.

Solutions that are compliant with the OASIS SAML 2.0 standard complies with different parts of the standard according to their role. For example a solution that acts as Identity Provider will have to implement more of the standard than a solution that acts as a Service Provider. To assist in determining which parts of the SAML 2.0 standard a solution must comply with OASIS has defined a set of operational modes that describe different roles for solutions, like

- Identity Provider,
- Service Provider,
- Attribute Service,
- etc.

For each operational mode it is described which parts of the SAML 2.0 standard that must be implemented and which that are optional.

An example of this is shown in the table below. The tables lists SAML 2.0 features required by the OASIS defined operational modes: Identity Provider (IdP), Identity Provider Lite (IdP Lite), Service Provider (SP), Service Provider Lite (SP Lite), and Enhanced Proxy Client (ECP)

Feature	IdP	IdP Lite	SP	SP Lite	ECP
Web SSO, <AuthnRequest>, HTTP redirect	MUST	MUST	MUST	MUST	N/A
Web SSO, <Response>, HTTP POST	MUST	MUST	MUST	MUST	N/A
Web SSO, <Response>, HTTP artifact	MUST	MUST	MUST	MUST	N/A
Artifact Resolution, SOAP	MUST	MUST	MUST	MUST	N/A
Enhanced Client/Proxy SSO, PACS	MUST	MUST	MUST	MUST	MUST
Name Identifier Management, HTTP redirect (IdP initiated)	MUST	MUST NOT	MUST	MUST NOT	N/A
Name Identifier Management, SOAP (IdP-initiated)	MUST	MUST NOT	OPTIONAL	MUST NOT	N/A
Name Identifier Management, HTTP redirect	MUST	MUST NOT	MUST	MUST NOT	N/A
Name Identifier Management, SOAP (SP initiated)	MUST	MUST NOT	OPTIONAL	MUST NOT	N/A
Single Logout (IdP-initiated) – HTTP redirect	MUST	MUST	MUST	MUST	N/A
Single Logout (IdP-initiated) – SOAP	MUST	OPTIONAL	MUST	OPTIONAL	N/A
Single Logout (SP initiated) HTTP redirect	MUST	MUST	MUST	MUST	N/A
Single Logout (SP-initiated) – SOAP	MUST	OPTIONAL	MUST	OPTIONAL	N/A
Identity Provider Discovery (cookie)	MUST	MUST	OPTIONAL	OPTIONAL	N/A

The table above that describes a subset of the OASIS SAML 2.0 operational modes has been taken [SamlConf]

Vendors normally state SAML 2.0 compliance by describing the operational modes their product support. To be able to prove actual compliance with the operational modes Liberty Alliance has included SAML 2.0 in its Liberty Interoperable testing program<sup>13</sup>.

When considering operational modes for the Danish SAML 2.0 federation profile, the following are relevant:

- Identity Provider (DK-IdP)
- Service Provider (DK-SP)
- Attribute Service (DK-Attr-Svc)




The majority of the requirements towards Identity Provider and Service Provider are covered by the operational modes shown in the table below.

---

<sup>13</sup> More information about the Liberty Interoperable™ program can be found at [http://projectliberty.org/index.php/liberty/liberty\\_interoperable](http://projectliberty.org/index.php/liberty/liberty_interoperable)

>

Feature	IdP	IdP Lite	SP	SP Lite
Web SSO, <AuthnRequest>, HTTP redirect	MUST	MUST	MUST	MUST
Web SSO, <Response>, HTTP POST	MUST	MUST	MUST	MUST
Web SSO, <Response>, HTTP artifact	MUST	MUST	MUST	MUST
Artifact Resolution, SOAP	MUST	MUST	MUST	MUST
Enhanced Client/Proxy SSO, PAOS	MUST	MUST	MUST	MUST
Name Identifier Management, HTTP redirect (IdP-initiated)	MUST	MUST NOT	MUST	MUST NOT
Name Identifier Management, SOAP (IdP-initiated)	MUST	MUST NOT	OPTIONAL	MUST NOT
Name Identifier Management, HTTP redirect	MUST	MUST NOT	MUST	MUST NOT
Name Identifier Management, SOAP (SP-initiated)	MUST	MUST NOT	OPTIONAL	MUST NOT
Single Logout (IdP-initiated) – HTTP redirect	MUST	MUST	MUST	MUST
Single Logout (IdP-initiated) – SOAP	MUST	OPTIONAL	MUST	OPTIONAL
Single Logout (SP-initiated) – HTTP redirect	MUST	MUST	MUST	MUST
Single Logout (SP-initiated) – SOAP	MUST	OPTIONAL	MUST	OPTIONAL
Identity Provider Discovery (cookie)	MUST	MUST	OPTIONAL	OPTIONAL

 Part of DK SAML 
  Not part of DK SAML 
  More strict requirements in DK SAML 2.0 \*)

\*) IdP MUST support SOAP binding for Single Logout. SP and SP Lite MUST support IdP Discovery

In addition, an Attribute Service (DK-Attr-Svc) is covered by the SAML Attribute Authority operational mode also described in "Conformance Requirements for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, 15 March 2005".

Further, DK-SAML contains specific Danish SAML 2.0 profiles in addition to the adopted subsets of OASIS profiles. These are:

- Authentication Assertion Profile
- OCES Attribute Profile
- Persistent Pseudonym Attribute Profile

It is expected that the requirements in these profiles can be fulfilled in COTS<sup>14</sup> products simply through configuration of product functionality.

Note that the OASIS operational modes do not explicit mention the following items relevant to DK-SAML 2.0:

- Ability to exchange metadata. Solutions conforming to the Danish profile are required to support generation and import of metadata.
- Support for persistent pseudonyms, which DK-IdP MUST support and DK-SP MAY support

Further beyond the scope of a SAML 2.0 profile the Danish federation architecture requires that an IdP must give the user an option to opt out of SSO (and thus be challenged for each authentication request). In addition Service Provider products being used for SSO with persistent pseudonyms must support or be modified to support dynamic account-linking where some form of authentication of the user on the SP side is performed when the link is created.

Thus when considering functional support for the DK-SAML 2.0 operational modes we can see in the above table that they are pretty well covered by the OASIS operational modes (besides the above mentioned items where the Danish profile goes further).[TG1]

The “IdP” mode supports the DK-IdP mode

The “IdP Lite” mode supports the DK-IdP mode with the one exception that the Danish SAML 2.0 federation profile also requires support for SOAP binding for Single Logout, where this is left as optional in for the “IdP Lite” mode.

The “SP” as well as the “SP Lite” mode supports the DK-SP mode with the exception that the Danish SAML 2.0 federation profile also requires support for IdP Discovery.

Thus when looking for COTS products adhering to the Danish SAML 2.0 profiles a quick way to find relevant products can be to restrict the search to products supporting the relevant operational modes listed above.[TG2]

Further, when acquiring a SAML 2.0 COTS-product it is recommended to ask for products where interoperability has been verified through participation in the Liberty Interoperable program. The test result from the interoperability testing is documented in a manner which makes it easy to determine for which operational modes a given product successfully has proved interoperability.

---

<sup>14</sup> COTS = Commercial Off The Shelf

---

## 13 Architectural Decisions

>

---

This chapter contains a number of architectural decisions which provide the rationale behind important choices made in the SAML profiles.

### 13.1 Attribute Profile in Requests

<b>Problem</b>	Should a Service Provider be able to specify which attribute profile he wishes a SAML assertion issued under?!
<b>Assumptions</b>	<p>An Identity Provider may support more than one attribute profile – in fact two different profiles are specified in this document.</p> <p>Some Service Providers may have different applications which require different profiles and it may therefore be an advantage to be able to state this in the authentication request going to the Identity Provider.</p>
<b>Alternatives</b>	<ol style="list-style-type: none"><li>1. Specify the desired attribute profile in the request.</li><li>2. Leave it to some out-of-band mechanism to determine this (e.g. the agreements between Identity and Service Provider).</li></ol>
<b>Analysis</b>	<p>There is no built-in mechanism in SAML 2.0 for specifying a desired profile, but the information could be passed as extensions (the &lt;AuthnRequest&gt; element is extensible). This profile could therefore define a new element for this under a common namespace.</p> <p>While this would allow for dynamic selection of attribute profiles, a local extension may be difficult to support for standard SAML products (needs to be tested in practice). The requirement could therefore lead to costly customization.</p>
<b>Decision</b>	Avoid extending the <AuthnRequest> message since it will require difficult and expensive customization by Service Providers.

### 13.2 Authentication Level in Requests

<b>Problem</b>	Should a Service Provider be able to specify the desired level of authentication in authentication requests to an Identity Provider?!
<b>Assumptions</b>	<p>An Identity Provider may support more than one authentication mechanism classified to different levels of authentication, see [ITTAAuthLevel].</p> <p>A Service Provider may have applications with different requirements for authentication level – based on the sensitivity of the applications. In this situation, it can be desirable that the Service Provider can tell which authentication level that is required for the resource the user is currently trying to access. This will ensure that the Identity Provider does not allow the user to authenticate by a mechanism that does not live up to the Service Provider's requirements and therefore will not</p>

	grant him access to the desired resource.
<b>Alternatives</b>	<ol style="list-style-type: none"> <li>1. Specify the desired authentication level in the request.</li> <li>2. Treat each authentication mechanism as a separate Identity Provider.</li> </ol>
<b>Analysis</b>	<p>Extending authentication requests with elements stating the desired level of authentication will allow dynamic selection of authentication mechanism and ensure that a user is not allowed to select or use mechanism that is not applicable.</p> <p>However, a local extension may be difficult to support for standard SAML products. The requirement could therefore lead to costly customization.</p> <p>SAML specifies an element called &lt;RequestedAuthnContext&gt; which can be used in authentication requests to specify requirements for the authentication context. Additionally, a large number of identifiers for different mechanisms are specified in [SAMLAuthnCxt]. The problem with this approach is that it specifies concrete mechanisms (e.g. MobileOneFactorUnregistered) and not a more abstract level of authentication.</p> <p>In the Virk portal it was chosen to include the authentication level in the request. This may suggest that their software suite (CA) may be configured to use it. The extensions element is:</p> <pre>&lt;samlp:Extensions&gt;   &lt;saml:Attribute     NameFormat="http://itst.dk/federated/attribute"     Name="AssuranceLevel"&gt;     &lt;saml:AttributeValue&gt;3&lt;/saml:AttributeValue&gt;   &lt;/saml:Attribute&gt; &lt;/samlp:Extensions&gt;</pre> <p>Here, the local assurance level attribute originally defined for assertions is reused.</p>
<b>Decision</b>	Avoid extending the <AuthnRequest> message since it will require difficult and expensive customization by Service Providers.

### 13.3 Signing of Meta Data

<b>Problem</b>	Should we require meta data to be signed and verified before use?!
<b>Assumptions</b>	The SAML specification optionally allows meta data to be signed.
<b>Alternatives</b>	<ol style="list-style-type: none"> <li>1. Require signing and verification of meta data.</li> <li>2. Rely on other mechanisms (e.g. signed emails) to secure meta data.</li> </ol>
<b>Analysis</b>	It is important that Service and Identity Providers never use meta data

	<p>which is not authentic or has been modified. Meta data contains data such as certificates and end-point which play a crucial role in the overall security.</p> <p>Signing of meta data (and verification before use) is a means to guarantee the authenticity and integrity of the data which is independent of how meta data was transferred. It is mandatory in the E-Authentication initiative from USA.</p> <p>However, some standard software products may not be able to support signed meta data; this has been indicated by some of the presentations from the American egov initiative.</p> <p>Another problem with signing meta data is that it will impossible to add, remove or change elements by hand. The POC for borgerportalen has for example shown that it was necessary to modify the XML file exported from one product in order to be able to import it in another product.</p>
<b>Decision</b>	Signing of meta data is left optional by this profile and it is left open to decide what will constitute an adequate protection of meta data in transit.

### 13.4 OCES Subject as Attribute

<b>Problem</b>	Should the OCES subject be included as a compound attribute in the OCES attribute profile (see section 8)?!
<b>Assumptions</b>	<p>The most interesting user attributes in OCES certificates are located in the subject field. For matters of simplicity and completeness, this field could be included in all assertions under the OCES attribute profile.</p> <p>The subject field may include the following information about the user / company (see [OCES-Pers] and [OCES-Medarb]):</p> <ol style="list-style-type: none"> <li>1. Country (M)</li> <li>2. Organization (O)</li> <li>3. Organizational Unit (O)</li> <li>4. Common Name (M)</li> <li>5. email address (O)</li> <li>6. Serial number (M) which holds: <ul style="list-style-type: none"> <li>o PID numbers for persons</li> <li>o CVR-RID numbers for employees</li> <li>o CVR numbers for companies</li> </ul> </li> </ol>
<b>Alternatives</b>	<ol style="list-style-type: none"> <li>1. Include OCES subject as compound attribute.</li> <li>2. Split OCES subject in atomic attributes.</li> </ol>

<b>Analysis</b>	<p>The advantage of including the entire subject in the assertion is</p> <ul style="list-style-type: none"> <li>• Completeness - all user attributes are included</li> <li>• Extensibility - if new subject attributes are specified in future editions of the OCES certificate policies, these will automatically be included in assertions as well.</li> </ul> <p>The disadvantage of this approach is that the attribute is not atomic (as is normally expected from an attribute) and therefore requires parsing by the Service Provider.</p> <p>It may be difficult for COTS products to extract relevant information from the assertion (e.g. using XPath expressions) when mapping the assertion to a local account. Furthermore, the attribute is a lot less "typesafe" compared to other attributes whose values can be defined by XML schemas.</p>
<b>Decision</b>	Split OCES subject in atomic attributes.

### 13.5 Binding for Single Logout Profile

<b>Problem</b>	Which binding should be chosen for the Single Logout Profile?!
<b>Assumptions</b>	
<b>Alternatives</b>	<ul style="list-style-type: none"> <li>a) SOAP binding</li> <li>b) HTTP Redirect binding</li> <li>c) HTTP POST binding</li> <li>d) Artifact binding</li> <li>e) A combination of different bindings – for example a front channel binding for the first message request and back channel / SOAP bindings for subsequent messages.</li> </ul> <p>Option a) is required by SAML conformance requirements for the IdP and SP operational modes – but optional for IdP Lite and SP Lite modes.</p> <p>Option b) is required by SAML conformance in all operational modes.</p> <p>Options c) and d) are not mentioned by SAML conformance requirements.</p> <p>Option e) is not mentioned directly by the SAML conformance requirements but is found in many descriptions and white papers. For example, it seems common to use HTTP Redirect for the first message and then use SOAP for subsequent message exchanges.</p>
<b>Analysis</b>	<p><b>SOAP Binding</b></p> <p><b>Pros:</b></p> <ul style="list-style-type: none"> <li>• The user's browser is not relied upon to transfer messages; this may be an advantage if the user has a slow or unreliable</li> </ul>

	<p>Internet connection, or if the user closes his browser before all logout requests have been sent. Thus, SOAP is more reliable than front-channel bindings, especially if there are many Service Providers with an active session.</p> <ul style="list-style-type: none"> <li>• The single logout process may “flicker” less.</li> </ul> <p><b>Cons:</b></p> <ul style="list-style-type: none"> <li>• Back-channel bindings are not recommended by the SAML profile [SAMLProf] for the first request message.</li> <li>• The SP or IdP receiving a single logout request via this binding does not get a “handle” to the user’s browser which may store important session information (e.g. in cookies). This may create problems in identifying which session(s) to terminate.</li> <li>• The SOAP Binding is not used by the other DK-SAML profiles. This may lead to a configuration overhead for Service Providers – e.g. for configuring SOAP security.</li> <li>• SOAP support is not mandatory according to SAML conformance requirements for the IdP Lite and SP Lite operational modes. Therefore, some lightweight SAML products may not support it.</li> </ul> <p><b>Synchronous Bindings</b></p> <p>All synchronous front-channel bindings have the following advantages:</p> <ul style="list-style-type: none"> <li>• They are recommended by the profile [SAMLProf] for the first message exchange. This is because the browser will propagate cookies which may contain important session information for the Identity Provider to identify the session.</li> </ul> <p>They all share the following disadvantages:</p> <ul style="list-style-type: none"> <li>• If the user closes his browser quickly after requesting single logout, the logout requests may not reach all Service Providers.</li> <li>• If one of the Service Provider fails to respond the logout “chain” will be broken and the user will not be logged out.</li> <li>• The logout process may cause the browser to flicker.</li> </ul> <p>The differences between the front-channel bindings are:</p> <ul style="list-style-type: none"> <li>• HTTP Redirect Binding uses URL parameters to transfer SAML protocol messages. Even though URL lengths theoretically can be infinite, they are unpredictably limited in practice. <ul style="list-style-type: none"> <li>○ The URL length limitation may create problems in</li> </ul> </li> </ul>
--	---

	<p>cases of long messages.</p> <ul style="list-style-type: none"> <li>○ HTTP Redirect is clearly favored by the SAML conformance requirements as support is required for all operational modes.</li> <li>• HTTP Artifact binding transfers SAML protocol messages by a small reference (an artifact). The real message is resolved via a second step using a synchronous back-channel (like SOAP). <ul style="list-style-type: none"> <li>○ The disadvantage of this binding would be the extra step required plus the requirement for SOAP for the back channel.</li> <li>○ The binding is not mentioned by SAML conformance requirements.</li> </ul> </li> <li>• HTTP POST binding transfers SAML protocol messages via a HTML form being submitted using the HTTP POST protocol. <ul style="list-style-type: none"> <li>○ The binding does not cause browser problems by many re-directs as described below (pro).</li> <li>○ The binding is supported by the Ping Federate and Oracle Identity Federation products (pro).</li> <li>○ The binding is not mentioned by the SAML conformance requirements. This may mean that fewer COTS products will support it. The Liberty Interoperability tests however define optional features for the POST binding that can be tested (con).</li> </ul> </li> </ul> <p>The Ping Federate release notes state the following problem with HTTP Redirect Binding:</p> <p><i>“Issuing an SLO request over the Redirect binding causes the user’s browser to be redirected between the IdP and each SP in turn resulting in a potentially large number of HTTP 302 Redirects. The number of redirects may exceed these browsers’ allowable redirect limit. When this limit is reached, the browser believes that a web site is mistakenly generating these redirects and displays the error.</i></p> <p><i>We recommend that for federation hubs that support users with multiple simultaneous open sessions, a binding other than Redirect be used for SLO.”</i></p>
<b>Decision</b>	<p>HTTP Redirect binding must be used for the first request going from a SP to the IdP. Subsequent request/response message exchanges must either use HTTP Redirect or SOAP.</p> <p>Support for HTTP Redirect is mandatory via the SAML conformance requirements.</p> <p>Support for SOAP is optional for SPs and mandatory for IdPs.</p> <p>SOAP is preferred when supported because it is more reliable than</p>

>

	HTTP Redirect.
--	----------------

### 13.6 Requirements for Identity Provider Discovery Profile

<b>Problem</b>	Should the DK-SAML profile require that Service Providers support the Identity Provider Discovery Profile from SAML?!
<b>Assumptions</b>	
<b>Alternatives</b>	<ul style="list-style-type: none"><li>a) Require discovery support from Service Providers.</li><li>b) Allow Service Providers to skip discovery and hard-code the Identity Provider.</li></ul>
<b>Analysis</b>	<p>In the SAML conformance requirements documents, the IdP Discovery Profile is mandatory to implement for the IdP and IdP Lite operational modes, but optional to implement for SP and SP Lite modes.</p> <p>Therefore, some SAML products on the market may not support it and the requirement could therefore create problems for Service Providers.</p> <p>On the other hand, support of discovery is an important element in the architecture in order to ensure that multiple Identity Providers can later co-exist. This is important in the future where multiple Identity Providers can easily emerge.</p> <p>To get an indication of actual product support, four representative products have been investigated for compliance:</p> <ul style="list-style-type: none"><li>• Computer Associates Site Minder Federation Services</li><li>• Ping Federate</li><li>• Oracle Identity Federation</li></ul> <p>The first two of these products are claimed as IdP Lite and SP Lite conformant in the Liberty interoperability test matrixes.</p> <p>Study of products documentation shows that all three products support the Identity Provider Discovery Profile.</p> <p>See <a href="http://www.projectliberty.org/liberty_interoperable/interoperable_products/saml_2_0_test_procedure_v2_0_interoperable_product_table">http://www.projectliberty.org/liberty_interoperable/interoperable_products/saml_2_0_test_procedure_v2_0_interoperable_product_table</a></p>
<b>Decision</b>	The Identity Provider Discovery Profile is required. It seems to be well supported by commercial products even though it is not formally required by SP and SP Lite operational modes as defined by SAML conformance.

### 13.7 Name Identifier Management Profile

<b>Problem</b>	Should the “Name Identifier Management Profile” be required and what binding should be selected?!
<b>Assumptions</b>	
<b>Alternatives</b>	<p>a) Require support – use HTTP Post Binding  b) Require support – use SOAP Binding  c) Require support – use HTTP Redirect Binding  d) Require support – use HTTP Artifact Binding  e) Don’t require support of the profile.</p> <p>Note that according to the SAML conformance feature matrix, the IdP Lite and SP Lite operational modes must not support the profile.</p> <p>Option a) and d) above are not mentioned by the SAML conformance.</p> <p>Option b) is required for the IdP mode but optional for the SP mode.</p> <p>Option c) is required for both the IdP and SP mode.</p>
<b>Analysis</b>	<p>The Web SSO profile allows a Service and Identity Provider to establish a shared persistent pseudonym during their first SSO interaction by requiring the user to initially login at both locations. Furthermore, an attribute profile is created to govern the content of assertions in this scenario (see section 9).</p> <p>After a persistent pseudonym identifier has been established, it may require management in the future. For example, if either Service- or Identity Provider wishes to terminate the identifier or change it to a different value or format. This management is handled by the Name Identifier Management Profile.</p> <p>As mentioned above, the profile is optional to implement for the IdP Lite and SP Lite operational modes. Therefore, some lightweight SAML products on the market may not support it. This may create problems for small Service Providers.</p> <p>Note further that the vast majority of Service Providers are expected to use the OCES attribute profile and not establish persistent pseudonyms (account linking versus account mapping). For these, the profile is of no benefit and requirement of mandatory support will only be a burden.</p> <p>Regarding binding selection, HTTP Redirect is clearly the most favored binding in the SAML conformance requirements. It is therefore expected that it will be widely supported in product implementations since vendors generally seek compliance.</p>
<b>Decision</b>	<p>Avoid requirements of the profile because:</p> <ol style="list-style-type: none"> <li>1. Real-life requirements and needs are very unclear at this point</li> <li>2. COTS support is very limited</li> </ol>

## 13.8 Attribute Encoding

<b>Problem</b>	How should identity attributes be encoded in SAML?!
<b>Assumptions</b>	<p>In order to enable a powerful federation and simplify life for Service Providers, there is a need to exchange a rich set of identity attributes between an Identity Provider and Service Providers.</p> <p>The required set of attributes includes X.509 attributes (common name, e-mail...), OCES-specific attributes (PID, RID, CVR, CPR) and sector specific attributes.</p> <p>Attributes are exchanged either via an assertion or in response to an attribute query.</p> <p>In addition to generic (federation-wide) attributes, some sectors, communities or portals may need to define attributes with local semantics.</p>
<b>Alternatives</b>	<ol style="list-style-type: none"> <li>1. Use the “Basic Attribute Profile” defined in [SAMLProf].</li> <li>2. Use the “X.500/LDAP Attribute Profile” defined in [SAMLProf].</li> <li>3. Define an attribute encoding based on URIs.</li> <li>4. Define an attribute encoding based on OIOXML schemas.</li> </ol>
<b>Analysis</b>	<p>The basic attribute profile defined in [SAMLProf] basically allows attributes of simple types to be encoded and referenced with a simple string name.</p> <p>The allowed set of attribute values are thus simple XML Schema types (for example xs:string). The names are simple strings and the profile therefore does not guarantee unique attribute naming.</p> <p>The advantage of this profile is simplicity and the avoiding extensions schemas to validate syntax. Furthermore, since the profile is covered by SAML conformance requirements and Liberty Interoperability testing procedures, COTS support can be expected to be quite good. Investigations of representative implementations further indicate that this is indeed the case.</p> <p>Many of the OCES attributes are defined with an OID and usage of the X.500/LDAP Attribute Profile would therefore be natural (the previous version of the DK-SAML profile used it extensively). However, it has since become evident that support for this profile is very limited in COTS products. Furthermore, the attribute profile specification in [SAMLProf] is broken and produces XML that does not conform to the schemas.</p> <p>Using an encoding with URIs instead has several advantages:</p> <ul style="list-style-type: none"> <li>• SAML Conformance requires support of the URI name format identifier “urn:oasis:names:tc:SAML:2.0:attrname-format:uri”. COTS support can therefore be expected to be quite good.</li> <li>• Attribute names will be unique.</li> </ul>

	<ul style="list-style-type: none"> <li>• The scheme can take advantage of the many OCES attributes with an Object Identifier (OID) [which can be represented by an URI].</li> <li>• The scheme is used by the E-Authentication initiative in USA and is therefore expected to have strong attention by COTS vendors.</li> </ul> <p>As a last option, the use of OIO XML has been considered. For example, the attribute name could be the unique path to the attribute's schema in the ISB. However, a number of disadvantages of this approach exist:</p> <ul style="list-style-type: none"> <li>• Few of the required attributes currently exist in OIO XML.</li> <li>• OIOXML typically use complex XML types which does not fit well with SAML; few COTS products are expected to support it.</li> </ul> <p>After discussion with OIO XML experts, it was agreed that OIO XML is not a good fit for this purpose.</p>
<b>Decision</b>	Use an attribute encoding where attribute names are URIs.

### 13.9 Core User Attributes to include in Authentication Assertion

<b>Problem</b>	<p>[ITTA<sub>Attrib</sub>] recommends that the following attributes always are included when exchanging user information:</p> <ul style="list-style-type: none"> <li>• sn - Surname</li> <li>• cn - Common name.</li> <li>• uid - User id</li> <li>• mail - email address</li> </ul> <p>and optionally:</p> <ul style="list-style-type: none"> <li>• uniqueAccountKey - Unique key to match and synchronize user information across systems and organization</li> <li>• cvrNumberIdentifier - An employee's organization identifier</li> </ul>
<b>Assumptions</b>	<p>The sn and cn attributes are prerequisites to create a user in an LDAP directory based on the inetOrgPerson (and person) schema.</p> <p>The uid attribute specifies the user id in the user's (principals) home organization (or credential issuing organization where home organization is unknown or doesn't exist – which is the case for citizens).</p> <p>The e-mail attribute is considered of general utility.</p> <p>The original goals with the “core attributes recommendation” was to supply at set of attributes that could be used to</p>

	<ul style="list-style-type: none"> <li>• Search/locate a user when direct account linking isn't possible</li> <li>• Supply basic "start" information for a Service Provider that wants to create an account for the user</li> </ul> <p>The ability to locate a user without having an exact identifier mapping to the user record does not seem to be a strong requirement currently. However, the ability to get basic user information in order to create a local user record still seems to be a valid requirement.</p> <p>Usage of the uniqueAccountKey hasn't really taken hold yet. However, as federated provisioning takes hold utilization may begin.</p> <p>CvrNumberIdentifier is widely used as an attribute for employees today.</p>
<b>Alternatives</b>	<ol style="list-style-type: none"> <li>1. Drop those "core" attributes that does not seem relevant in the current situation from authentication assertions.</li> <li>2. Include all "core" attributes in the authentication assertions.</li> </ol>
<b>Analysis</b>	<p>The biggest issue is whether it is relevant to include the uid attribute.</p> <p>Some potential credentials for usage in the Danish public sector in the near term are:</p> <p>OCES Digital Signature, Pin codes (from Tax Agency, KMD, local govt), NetID, Local Net login (Miljøportalen phase 2 federation), DK-AAI credentials.</p> <p>For some of these credentials situations may appear where the SAML subject is different from the user id at the credential supplier. For example, the subject may be amended to assure uniqueness. However, it may still be of value for the service provider to receive the users correct local user id.</p>
<b>Decision</b>	<p>The attributes from <b>[ITTAtrib]</b> must be included in all Danish attribute profiles – except pseudonym profiles targeted at privacy – with the same provisions for which attributes are mandatory and which are optional.</p> <p>The contents of the uid attribute should be the user id in his home organization. The actual content of the uid attribute is left to the discretion of the IdP, and should be documented by the IdP.</p> <p>Examples:</p> <ul style="list-style-type: none"> <li>• For a POCES certificate the uid can hold the PID number</li> <li>• For a MOCES certificate the uid can hold the RID number</li> <li>• For a locally authenticated user the uid can hold the local user id (while the SAML subject may be an amended user id to assure uniqueness outside the local organization)</li> </ul>

### 13.10 Include Certificate Issuer in OCES Attribute Profile

<b>Problem</b>	Should the OCES attribute profile include an attribute which identifies
----------------	---

---

>

---

	the issuer of the certificate?!
<b>Assumptions</b>	<p>A Service Provider may need to contact the issuer of a certificate in order to perform a revocation check or query attributes about the subject (e.g. the current OCES PID2CPR and isLRA services).</p> <p>In the future, there may be several different OCES CAs.</p>
<b>Alternatives</b>	<ol style="list-style-type: none"><li>1. Include Issuer identification in the attribute profile (i.e. the Issuer DN from the certificate).</li><li>2. Don't include Issuer identification.</li></ol>
<b>Analysis</b>	<p>Generally, a certificate serial number is only unique within a CA. However, within the OCES PKI it has been ensured that these are unique across CAs<sup>15</sup>.</p> <p>Furthermore, in the coming version of the OCES PKI, the services offered by each CA such as revocation check and the PID2CPR and isLRA services will be offered by a common front-end, such that relying parties do not have to contact each CA individually depending on the current certificate being validated.</p> <p>Hence, a Service Provider only has to know the certificate serial number to perform lookup of revocation status or PID2CPR and isLRA services.</p> <p>Normally, a Service Provider will trust the Identity Provider to have performed a revocation check on the user certificate before issuing an authentication assertion.</p>
<b>Decision</b>	Do not include issuer identification in the OCES attribute profile as it is not necessary.

---

<sup>15</sup> This has been stated by Sikkerhedskontoret at the IT and Telecom Agency.

---

## Appendix A: Overview of Profile Changes

>

---

This appendix provides an overview of the major changes in the new edition of the DK-SAML profile. First, however, a number of problems and issues with the old architecture will be highlighted to provide the rationale behind the changes.

### 13.11 Experience from the e-Authentication initiative

The e-Authentication initiative from USA has deployed a similar federation architecture for American eGovernment. Experience from this project should be leveraged in the Danish federation and includes the following findings [EAuth-V2]:

- The old architecture is expensive and time consuming for federation members:
  - a. Mutually authenticated TLS is difficult to configure due to lack of product GUI and poor documentation.
  - b. Mutually authenticated TLS requires non-standard ports for web services leading to firewall issues
  - c. Federation members must develop custom code for integrating with the authentication portal.
- There are technical issues with signing and encryption of assertions.
- The old architecture did not scale well (the authentication portal is a bottleneck).
- There are operational issues with error handling.
- Some SAML bindings are better than others in practice; HTTP Post is for example simpler to implement, faster to deploy and scales better than artifact binding.
- There are usability issues by having an additional party (the authentication portal) interacting with the user and performing many re-directs (confusing).

### 13.12 Profile changes

The following lists the most important changes in the new version of the profile:

- The Authentication Portal component (and all interaction with it) is removed. Users will instead approach a Service Provider application directly (via their browser) or perhaps navigate via a portal (such as borger.dk) which links to or frames application content.
- The Attribute Service Profile [AttrProf] has been incorporated into this profile and revised to be consistent with the new profile (e.g. regarding choice of bindings).
- Proprietary HTTP variables for communicating selected application and login service are removed.
- The SAML 2.0 <AuthnRequest> message is used for integration from Service Providers to Identity Providers.
- HTTP Post Binding replaces HTTP Artifact Binding.
- Request and response messages must be signed.

---

>

---

- All assertions are required to be signed and (XML) encrypted to evolve from transport based security to message based security. Note that an assertion is considered signed if it is embedded in a signed <Response> message.
- The serial number attribute now holds the certificate serial number (and not the subject serial numbers).
- The subject serial number attribute from the certificate (which contains combined PID-CVR, CVR, or CPR number) is now split in “atomic” attributes and encoded differently to avoid confusion with the certificate serial number.
- More fields from the OCES certificates have been added to the OCES attribute profile.
- Attributes are no longer encoded via the X.500/LDAP attribute profile. Instead attribute names are URIs.
- An additional profile supporting enhanced user privacy via persistent pseudonyms is introduced.
- The SAML 2.0 Identity Provider Discovery Profile is used instead of implementing discovery via the authentication portal.

These changes lead to a simpler, more standards-based architecture.

---

## Appendix B: References

---

>

[SAMLCore]	“Assertion and Protocols for the OASIS Security Assertion Markup Language 2.0”, OASIS Standard. <a href="http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf">http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf</a>
[SAMLProf]	“Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0”, OASIS Standard. <a href="http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf">http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf</a>
[SAMLBind]	“Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0”, OASIS Standard. <a href="http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf">http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf</a>
[SAMLTechOver]	”Security Assertion Markup Language (SAML) V2.0 Technical Overview”, OASIS, Working Draft 21 February 2007”.
[SAMLMeta]	“Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0”, OASIS Standard 15 March 2005.
[SAMLConf]	“Conformance Requirements, OASIS Security Assertion Language (SAML) V2.0”, OASIS Standard 15. March 2005. <a href="http://docs.oasis-open.org/security/saml/v2.0/saml-conformance-2.0-os.pdf">http://docs.oasis-open.org/security/saml/v2.0/saml-conformance-2.0-os.pdf</a>
[SAMLDepl]	”SAML V2.0 Deployment Profiles for X.509 Subjects”, OASIS Draft, 26 March 2007.
[ITTArch]	“Anbefaling om fælles arkitektur for tværgående autenticitetssikring”. <a href="http://www.oio.dk/files/Horing.B.st.tvergaendeautenticitetssikring.v3.pdf">http://www.oio.dk/files/Horing.B.st.tvergaendeautenticitetssikring.v3.pdf</a>
[ITTAuthLevel]	”Vejledning vedrørende niveauer af autenticitetssikring”. <a href="http://www.oio.dk/files/Horing.B.st.niv.autenticitetssikring.v3.pdf">http://www.oio.dk/files/Horing.B.st.niv.autenticitetssikring.v3.pdf</a>
[ITTAttrib]	”Anbefaling til kerneattributter for bruger”. <a href="http://www.oio.dk/files/Horing.B.st.kerneattributter.v3.pdf">http://www.oio.dk/files/Horing.B.st.kerneattributter.v3.pdf</a>
[ITTUID]	”Anbefaling til unik id-nøgle”. <a href="http://www.oio.dk/files/Horing.B.st.id-nogle.v3.pdf">http://www.oio.dk/files/Horing.B.st.id-nogle.v3.pdf</a>
[EgovTechApp]	“Technical Approach for the Authentication Service Component Version 1.0.0 June 28, 2004”. <a href="http://www.cio.gov/eaauthentication/documents/TechApproach.pdf">http://www.cio.gov/eaauthentication/documents/TechApproach.pdf</a>
[EgovSAMLProf]	“SAML Artifact Profile as an Adopted Scheme for E-Authentication”. <a href="http://www.cio.gov/eaauthentication/documents/SAMLprofile.pdf">http://www.cio.gov/eaauthentication/documents/SAMLprofile.pdf</a>
[EgovIntf]	“E-Authentication Interface Specifications for the SAML Artifact Profile” <a href="http://www.cio.gov/eaauthentication/documents/SAMLspec.pdf">http://www.cio.gov/eaauthentication/documents/SAMLspec.pdf</a>
[NistElAuth]	“Electronic Authentication Guideline,NIST Special Publication 800-63 Version 1.0.1 <a href="http://www.csrc.nist.gov/publications/nistpubs/800-63/SP800-63v6_3_3.pdf">http://www.csrc.nist.gov/publications/nistpubs/800-63/SP800-63v6_3_3.pdf</a>
[OCESPers]	“Certifikatpolitik for OCES-personcertifikater”, Version 3.0, IT- og Telestyrelsen
[OCESMedarb]	“Certifikatpolitik for OCES-medarbejdercertifikater”, Version 4.0, IT- og Telestyrelsen
[GartBusCas]	”Business Case” Fælles login service og rettighedsstyring. 24. april 2006, Gartner Consulting.

- 
- [EAuth-V2]** E-Authentication SAML 2.0 Working Architecture Document, February 2007, Version 1-0-0
- [EAuthIntf]** “E-Authentication Federation Architecture 2.0 Interface Specifications”, Version 1.0.0, May 4, 2007.  
<http://www.cio.gov/eauthentication/TechSuite.htm>
- [EAuthTechApp]** “Technical Approach for the Authentication Service Component”, Version 2.0.0, May 4, 2007.  
<http://www.cio.gov/eauthentication/TechSuite.htm>
- [Sikkerheds-bekendtgørelsen]** ”Sikkerhedsbekendtgørelsen”, Datatilsynet, 2001.  
[http://www.datatilsynet.dk/include/show.article.asp?art\\_id=495](http://www.datatilsynet.dk/include/show.article.asp?art_id=495)
- [Terms]** ”Fælles Brugerstyringsløsning, Koncepter og Definitioner”, The IT and Telecom Agency, Søren Peter Nielsen, Februar 2007.
- [AttrProf]** “SAML Attribute Service Profile for eGovernment”, The IT and Telecom Agency, December 2006.
- [LibInterop]** “SAML 2.0 Interoperability Testing Procedures – Version 2.0”, Liberty Alliance Project.
-

