APPENDIX 2 – Technical Requirements

Version 1.52

Table of Contents

Technical requirements for membership in Sambi

Sambi has, like many other federation initiatives, the goal to use the following SAML[1]-profiles:

- The implementation profile eGov[2] 2.0 (describes which parts of SAML that must be implemented)
- The deployment profile saml2int[3] (describes which parts of SAML that must be in use and how these shall be used)

These SAML capabilities, where most of them are part of the implementation profile eGov2, and how they are affected of the deployment profile saml2int, are shown comprehensively in the following description of the technical requirements.

Requirements on Members

Service Provider, SP

The Service Provider often states requirements on Identification concepts, required Attributes and Level of Assurance, LoA.

Identity Provider, IdP

The Identity Provider is presupposed to have access to the registers needed to supply the Attributes that are requested by the Service Provider. Attributes can contain personal characteristics or other information that are the basis for a decision of system authorization and access control in the Services that rely on the IdP.

User

Every User in Sambi is a representative for, or acts on behalf of, a corporate body.

Federation Operator

One of the most important responsibilities of the Federation Operator is to supply an aggregation of digitally signed SAML metadata. This can be regarded as the technical core of the Federation, which ties the parties in the federation together.

---

1 Organization for the Advancement of Structured Information Standards (OASIS) Security Assertion Markup Language 2.0 (SAML)
2 Kantara Initiative eGov 2.0 profile
3 Interoperable SAML 2.0 Web SSO deployment profile

General Technical Requirements

Key Management

Security requirements for keys for signing and encryption

All Members in the Federation **must** in a secure way create, manage and store signing and encryption keys for at least:

a) metadata
b) authentication responses
c) communication

If nothing else is stated, algorithms and key lengths for authentication, encryption and signing **must** follow NIST SP 800-131[4] or ETSI TS 102 176-1[5]. Regarding the choice of algorithm, the requirements can be fulfilled by using SHA-256 and RSA with a key length (modulus) of at least 2048 bits.

Please note that the requirements on key lengths and algorithms are subject to constant evaluation and that the requirements can be changed over time.

The publishing of the Federation Operator´s public key

The Federation Operator´s public key is used for verifying signatures of published metadata. The current key is published as a text file on the web site of Sambi, www.sambi.se.

Verification of the Federation Operator´s public key

When updating the Federation Operator´s public key in a Member´s local configuration, the Member **must** verify its authenticity against at least two different sources. The following are acceptable verification sources:

- fetch the certificate including the public key directly from where it is published (www.sambi.se), including a positive verification of the certificate that identifies the site of publication
- contact with Sambi´s support service, where the certificates digital fingerprint is verified over telephone (SHA-1 hex).

4 http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf
5 http://www.etsi.org/deliver/etsi_ts/102100_102199/10217601/02.01.01_60/ts_10217601v020101p.pdf

Routines for changing a Member´s encryption keys

When changing a Member´s encryption keys, the following steps **shall** be performed:

1. The Member conveys metadata including a new certificate, with the new public encryption key, to the Federation Operator for publishing.
2. Until the new encryption key has reached all other parties within the Federation, the Member **shall** use double private keys for decryption.

Routines for changing a Member´s signing keys

When changing a Member´s signing keys, the following steps **shall** be performed:

1. The Member conveys metadata including both the new and the old (public) signing key to the Federation Operator for publishing.
2. During a transition period, all other parties **must** use double keys for verifying the authenticity of the signature.
3. When the new key has reached all other parties in the Federation, the Member **shall** convey updated metadata including only the new (public) signing key to the Federation Operator.

SAML metadata (MD)

In order to enable trust for assertions from other parties between Members in the Federation, exchange of public keys between the parties is needed, for verifying e.g. the authenticity of signatures.

This exchange is performed by aggregating local SAML metadata (MD) by the Federation Operator. This metadata describes the Member´s characteristics, capacities and public keys. The Federation Operator performs an analysis of the metadata, prior to signing and publishing the aggregated SAML metadata. The aggregated and signed SAML metadata published by the Federation Operator is hence the collective description of all the Federation actors' characteristics, capacities and public keys.

Publishing of SAML metadata

The aggregated and signed SAML metadata for the Federation is published on Sambi´s web site, www.sambi.se.

Verification of the signed SAML metadata

Every Member **must** verify the digital signature that encloses the SAML metadata at every update of the local copy, using the public key published by the Federation Operator.

SAML metadata format

Sambi uses saml2int as *deployment profile,* which describes how SAML metadata shall be presented. The format of SAML metadata is regulated in *SAML V2.0 metadata specification* [SAML2Meta[6]] and the handling of SAML metadata is regulated in OASIS *Metadata Interoperability Profile* [MetaIOP[7]]. All Members in the Federation **must** support these profiles.

For metadata for Service Providers and Identity Providers, the following requirements apply. For all <EntityDescriptor>, the following **must** be included:

- <Organization>, with <OrganizationName>, <OrganizationDisplayName> and <OrganizationURL> and with xml:lang="sv". <OrganizationDisplayName> **must** confirm to a separate name standard.
- <ContactPerson> with contactType="technical" and with contactType="support". Every <ContactPerson> **must** at least include one <EmailAddress>. The support contact refers to support for other Federation Members, and as an escalation path from first line support at other Federation Members.

The Federation Operator´s aggregated metadata **must** follow:

- Metadata **shall** consist of one <EntitiesDescriptor> root node, which contains <EntityDescriptor> nodes. Nested <EntitiesDescriptor> nodes **must not** be present.
- cacheDuration and valid Until **must** be present.

Updating local metadata

Local copies of the aggregated Federation Metadata **should** be updated at least with the periodicity that is stated in cacheDuration and **must not** be considered valid after validUntil.

Metadata **should** be updated automatically according to cacheDuration. If fetching metadata fails, old metadata may be used until the time stated in validUntil. After that time communication with the SP and IdP listed in the old metadata **must not** occur.

Directory Service (DS)

In the basic scenario, where a User wishes to use a Service, for which the User is not yet identified, he is asked to identify himself. In a two party relation it is unambiguous which Identity Provider to use for this. In a Federation, such as Sambi, with the possibility for a

6 http://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf
7 http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-iop.pdf

large number of Identity Providers, a generic function is needed to assign the User to “his” Identity Provider.

A Directory Service uses SAML metadata to show the User the Identity Providers in the Federation.

The address of the central Directory Service is published on Sambi´s web site, www.sambi.se.

A central Directory Service is not needed for cooperation within a Federation. The Service Provider can choose to implement his own function for local assigning, based on SAML metadata.

A Directory Service **must** show IdPer in order to mitigate the risk of confusion between these. If there is a risk of confusion the OrganizationDisplayName **must** be shown.

In addition to the basic scenario, it is possible to use an *unsolicited response,* which means that the User first connects to his Identity Provider with a parameter in the call, which then is used to assign the User to the correct Service.

The assigning of Identity Providers is regulated in the OASIS *Identity Provider Discovery Service Protocol Profile* [IdPDisco[8]]. All Members of the Federation **should** support this profile.

Pseudonymised Identities (NameID)

A cornerstone for Sambi is to continuously protect personal integrity. Hence pseudonyms **should** be used as far as possible as the identification concept (NameID).

There are two kinds of pseudonyms. *Persistent pseudonyms,* (permanent), have the characteristic that they always represent the same User in the Service in question. *Transient pseudonyms* (non-permanent) are temporary and are never reused.

When using persistent pseudonyms different pseudonyms are presented for every Service. When using transient pseudonyms a new pseudonym is presented for every occasion and every Service.

Pseudonyms are part of the standard specification for SAML 2.0 [SAML2Core[9]] and the following **shall** be supported:

8 http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-idp-discovery.pdf
9 http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf

- urn:oasis:names:tc:SAML:2.0:nameid-format:persistent
- urn:oasis:names:tc:SAML:2.0:nameid-format:transient

Authentication Request

In the basic scenario when a User wants to access a Service Provider, but has not been identified earlier, he will be asked to identify himself. The Service will create a request called AuthenticationRequest. The User will send this to his Identity Provider through an http-redirect.

Sambi uses saml2int as deployment profile. It describes how SAML *V2.0 Web Browser SSO Profile* [SAML2Prof[10]] shall be used, including Authentication Requests. The profile states among other things that:

- An Identity provide **may** omit to verify signed Authentication Requests if it can be suspected that they might be used of Denial of Service (DoS) attacks.

  Specific requirements:
- Communication **shall** be protected by TLS in the transport layer in accordance with RFC 7525[11]
- <RequestedAuthnContext> **may** be given, and if so, it **shall** be set according to [IAP] and [IANA LoA] <AuthnContextClassRef> http://id.sambi.se/loa/loa3. This will be the case as long as Sambi only supports LoA3.
- An Identity Provider **shall** validate that the AssertionConsumerServiceURL is in compliance with the Service Provider's metadata.

Authentication Response

An Authentication Response can be the result of an Authentication Request, but it can also be a response without a prior request. The latter response is called an *unsolicited response*.

Sambi uses saml2int as deployment profile. It describes how *SAML V2.0 Web Browser SSO Profile* [SAML2Prof] shall be used, including Authentication Responses. The profile states among other things that:

- The Authentication Response **shall** be signed with a key that is associated with the Identity Provider´s SAML metadata.
- Service Providers **must** accept *unsolicited responses.*

10 http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf
11 https://datatracker.ietf.org/doc/rfc7525/

- Service Providers **must** verify signatures and decrypt responses with one of the valid keys that are published in the SAML metadata. This mechanism **must** be possible to use at key exchange.
- Service Providers **must not** use any validity of the certificates used as bearers of SAML metadata as an indication of the validity of the key. All keys available in the SAML metadata **shall** be considered valid.
- Keys that are not in use **must** be removed from the SAML metadata.

Specific requirements:

- Communication **shall** be protected by TLS in the transport layer in accordance with RFC 7525[11]
- If TLS/SSL cannot be used, the Authentication Response, *AthenticationResponse,* **must** be encrypted in its entirety with the Service Provider´s public key that is published in the SAML metadata.
- <AuthnStatement> **shall** be given in accordance with [IAP] and [IANA LoA] with <AuthnContextClassRef> set to http://id.sambi.se/loa/loa3. This will be the case as long as Sambi only supports LoA3.

Managing different Levels of Assurance (LoA)

Sambi is a Federation that handles different Levels of Assurance, in accordance with the Trust Framework. It is the Service Provider that chooses the Level of Assurance needed based on how sensitive its information is. Members must be able to exchange information regarding which Levels of Assurance Identity Providers can offer and which levels Service Providers request. The information regarding Levels of Assurance can be added to the SAML metadata, as well as within an Authentication Request and an Authentication Response.

Information regarding the Level of Assurance in the SAML metadata has the advantage that a Directory Service can reduce the selection of Identity Providers for a User. Only those that fulfil the requested Level of Assurance need to be shown.

In the SAML metadata, the Level of Assurance is represented by one or more Attribute. All Members **should** be able to manage extended SAML metadata that allows the presentation of Attributes according to SAML V2.0 Metadata Extension for Entity Attributes Version 1.0[12]. The Attributes for Level of Assurance are shown according to SAML V2.0 Identity Assurance Profiles Version 1.0[13] and have the following names:

- http://id.sambi.se/loa/loa2

[12] http://docs.oasis-open.org/security/saml/Post2.0/sstc-metadata-attr.pdf

[13] http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-assurance-profile.pdf

- http://id.sambi.se/loa/loa3
- http://id.sambi.se/loa/loa4

Exchange of information regarding Level of Assurance concerning an Authentication Request and an Authentication Response give the possibility to manage the fact that an Identity Provider can represent different categories of Users, where it is not evident that the Users´ identities have the same Level of Assurance. Furthermore, a User can have access to different methods for authentication, which lead to different Levels of Assurance. Exchange of such information is performed according to Authentication Context for the OASIS Security Assertion Markup Language (SAML) V2.0 [14]. All Levels of Assurance are presented as an Authentication Context. This presentation of the Levels of Assurance is performed according to SAML V2.0 Identity Assurance Profiles.

The Levels of Assurance for Sambi are referenced by a specific URI for every level. They define the authentication classes:

- http://id.sambi.se/loa/loa2
- http://id.sambi.se/loa/loa3
- http://id.sambi.se/loa/loa4

This URI is found in the schema as targetNamespace.

The Attribute governingAgreementRef in the element GoverningAgreement in the schema contains a URL that refers to the external documentation that defines the level.

The signaling of Levels of Assurance in Authentication Requests and Responses **shall** be handled within AuthnContextClassRef.

An Authentication Request **should** signal the required Level of Assurance. The Attribute [Comparison] **must** be set to “exact” or be left out. Certain SAML software products support only exact matching of <saml:AuthnContextClassRef>. If [Comparison] is left out, it **must** be interpreted as “exact”, according to SAML-Core-2.0.

To eliminate problems for a User that already is authenticated with a higher Level of Assurance, an Authentication Request **should** contain all Levels of Assurance that fulfil the Service´s requirements. A set of Levels of Assurance **shall** be interpreted as an ordered list, where the first element represents the preferred level.

[14] http://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf

An Authentication Response **shall** signal the Level of Assurance that the User has been authenticated with. If the Identity Provider is not able to match the requested level, `<StatusCode> [urn: oasis:names:tc:SAML:2.0:status:NoAuthnContext]` shall be given in the response. Unsolicited responses **shall** also signal the Level of Assurance that the User has been authenticated with.

It is always the consumer of the Authentication Response that has the responsibility to make a correct judgement of the Level of Assurance in the response. If the response is incorrect and lacks the signaling of Level of Assurance, no level can be presupposed.

A schema for the context classes is found in http://www.iana.org/assignments/loa-profiles/loa-profiles.xhtml and also later in this document.

sambi.se-loa2

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
   targetNamespace="http://id.sambi.se/loa/loa2"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   xmlns="http://id.sambi.se/loa/loa2"
   finalDefault="extension"
   blockDefault="substitution"
   version="2.0">
  <xs:redefine schemaLocation="http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-authn-context-types-2.0.xsd">
  <xs:annotation>
    <xs:documentation>
    Class identifier: http://id.sambi.se/loa/loa2 Defines Level 2 of the
Sambi.se Assurance Framework
    </xs:documentation>
  </xs:annotation>
  <xs:complexType name="AuthnContextDeclarationBaseType">
    <xs:complexContent>
    <xs:restriction base="AuthnContextDeclarationBaseType">
      <xs:sequence>
      <xs:element ref="GoverningAgreements"/>
      </xs:sequence>
      <xs:attribute name="ID" type="xs:ID" use="optional"/>
    </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="GoverningAgreementRefType">
    <xs:complexContent>
    <xs:restriction base="GoverningAgreementRefType">
      <xs:attribute name="governingAgreementRef"
      type="xs:anyURI"
      fixed="https://www.sambi.se/sambi-loa"
      use="required"/>
    </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
  </xs:redefine>
</xs:schema>
```

sambi.se-loa3

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
   targetNamespace="http://id.sambi.se/loa/loa3"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   xmlns="http://id.sambi.se/loa/loa3"
   finalDefault="extension"
   blockDefault="substitution"
   version="2.0">
  <xs:redefine schemaLocation="http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-authn-context-types-2.0.xsd">
  <xs:annotation>
    <xs:documentation>
    Class identifier: http://id.sambi.se/loa/loa3 Defines Level 3 of the
Sambi.se Assurance Framework
    </xs:documentation>
  </xs:annotation>
  <xs:complexType name="AuthnContextDeclarationBaseType">
    <xs:complexContent>
    <xs:restriction base="AuthnContextDeclarationBaseType">
      <xs:sequence>
      <xs:element ref="GoverningAgreements"/>
      </xs:sequence>
      <xs:attribute name="ID" type="xs:ID" use="optional"/>
    </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="GoverningAgreementRefType">
    <xs:complexContent>
    <xs:restriction base="GoverningAgreementRefType">
      <xs:attribute name="governingAgreementRef"
      type="xs:anyURI"
      fixed="https://www.sambi.se/sambi-loa"
      use="required"/>
    </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
  </xs:redefine>
</xs:schema>
```

sambi.se-loa4

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
   targetNamespace="http://id.sambi.se/loa/loa4"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   xmlns="http://id.sambi.se/loa/loa4"
   finalDefault="extension"
   blockDefault="substitution"
   version="2.0">
  <xs:redefine schemaLocation="http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-authn-context-types-2.0.xsd">
  <xs:annotation>
    <xs:documentation>
    Class identifier: http://id.sambi.se/loa/loa4 Defines Level 4 of the
Sambi.se Assurance Framework
    </xs:documentation>
  </xs:annotation>
  <xs:complexType name="AuthnContextDeclarationBaseType">
    <xs:complexContent>
    <xs:restriction base="AuthnContextDeclarationBaseType">
      <xs:sequence>
      <xs:element ref="GoverningAgreements"/>
      </xs:sequence>
      <xs:attribute name="ID" type="xs:ID" use="optional"/>
    </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
  <xs:complexType name="GoverningAgreementRefType">
    <xs:complexContent>
    <xs:restriction base="GoverningAgreementRefType">
      <xs:attribute name="governingAgreementRef"
      type="xs:anyURI"
      fixed="https://www.sambi.se/sambi-loa"
      use="required"/>
    </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
  </xs:redefine>
</xs:schema>
```

Single Logout (SLO)

Sambi **does not**, at the moment, require that Members shall support *single-logout*. The Federation does not however restrict Members from implementing *single-logout.*

The technical specification for managing *single-logout* is found in *Single-logout Profile*[15]. It should be noted that session handling is not part of the SAML framework, which makes the matter larger than only a part of a technical specification.

If a Service Provider implements *single-logout* it is important that it is clear from the user interface that a single-logout is carried through and that the User is logged out from all Services that he is logged in to.

Attribute Authority (AA)

Sambi does not offer any Attribute provisioning service common to all Members in the Federation.

Time

It is crucial for Sambi that all Members use a reliable time source. The time source **must** be traceable to the Swedish national UTC(SP)[16]. This should be implemented using the standardized Network Time Protocol (NTP). The accuracy should never be less than one second.

15 http://docs.oasis-open.org/security/saml/v2.0/saml-profiles-2.0-os.pdf
16 http://www.sp.se/sv/index/services/time_sync/ntp/Sidor/default.aspx