Sign Message i Legitimeringstjänster

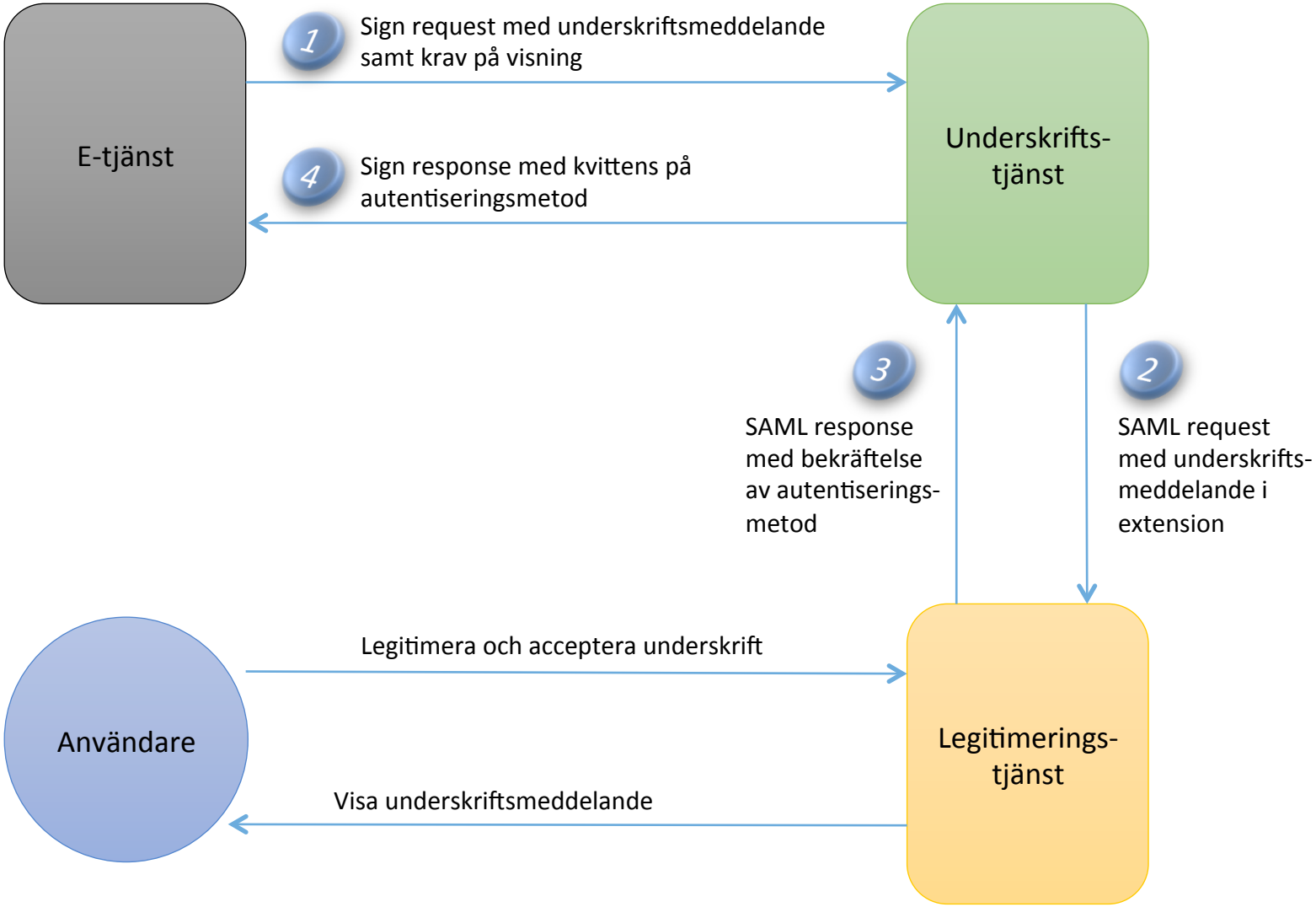
Designförslag

Stefan Santesson (Stefan@aaa-sec.com)

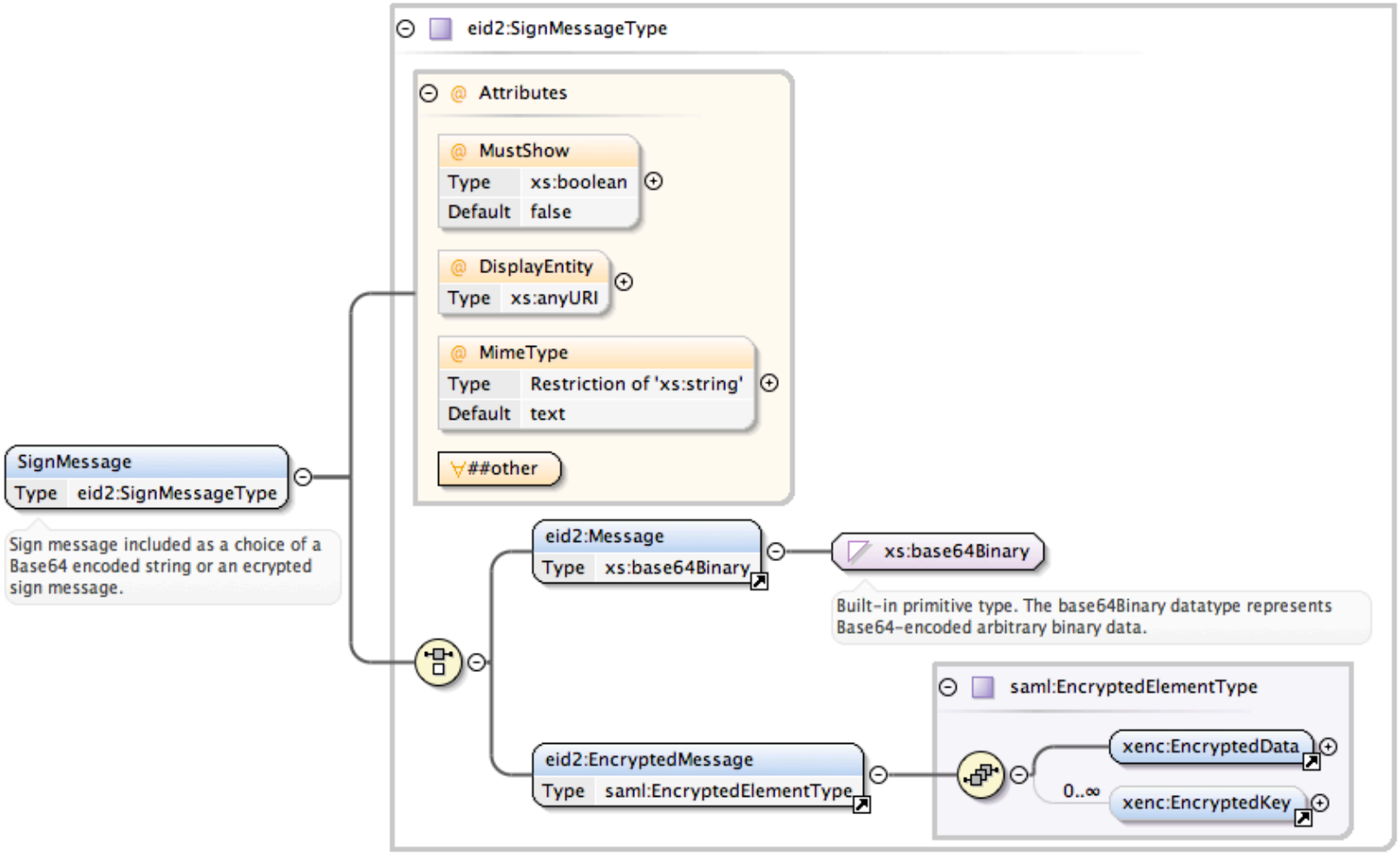
Kompatibilitetskrav

	SAML	Open SAML	Shibboleth	Övriga std prod
SP	Krav	Krav	Krav	Krav
Sig tjänst (SP)	Krav	Krav		
IdP	Krav	Krav	Krav	

Lösningsförslag - överblick



Protokollelement – Uppdatering av DSS extension (version 1.1)



XML Schema

```
<xs:element name="SignMessage" type="eid2:SignMessageType">
<xs:complexType name="SignMessageType">
    <xs:choice>
      <xs:element ref="eid2:Message"/>
      <xs:element ref="eid2:EncryptedMessage"/>
    </xs:choice>
    <xs:attribute name="MustShow" type="xs:boolean" default="false"/>
    <xs:attribute name="DisplayEntity" type="xs:anyURI"/>
    <xs:attribute name="MimeType" default="text">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="text/html"/>
          <xs:enumeration value="text"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
<xs:element name="Message" type="xs:base64Binary"/>
<xs:element name="EncryptedMessage" type="saml:EncryptedElementType"/>
```

Protokollelement

Element	Förklaring
MustShow (Attribut)	True = Underskriftsmeddelandet måste visas för att underskrift skall skapas
DisplayEntity (Attribut)	Mottagare av krypterat meddelande. Om detta attribut är närvarande så skall det inom ramen för gällande implementationsprofil innehålla legitimeringstjänstens EntityID.
MimeType (Attribut)	Identifierar MimeType för meddelandeformat. Kan innehålla ett av värdena ”text” eller ”text/html”
Message	Base64Binary innehållande UTF-8 kodat sign message enligt definierat format
EncryptedMessage	Krypterat Message element

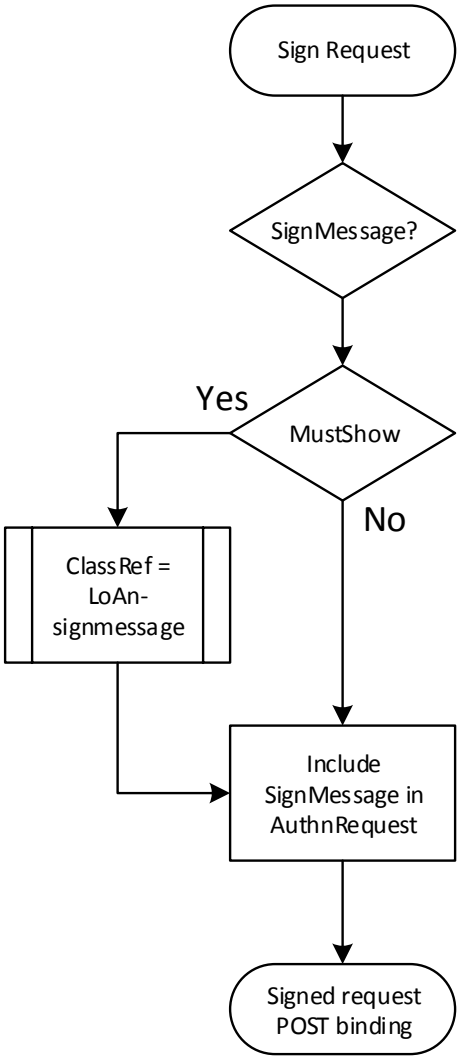
AuthnContextClassRef

- URI som definierar identifierar en specifik autentiseringsmetod.
- Hittills har vi haft en ClassRef URI per LoA
- Enligt detta förslag definieras en extra URI per LoA som ställer krav på att legitimeringstjänsten implementerar ett flöde som innefattar visning av signeringsmeddelande. Ex:
  - http://id.elegnamnden.se/loa/1.0/loa3
  - http://id.elegnamnden.se/loa/1.0/loa3-sigmessage

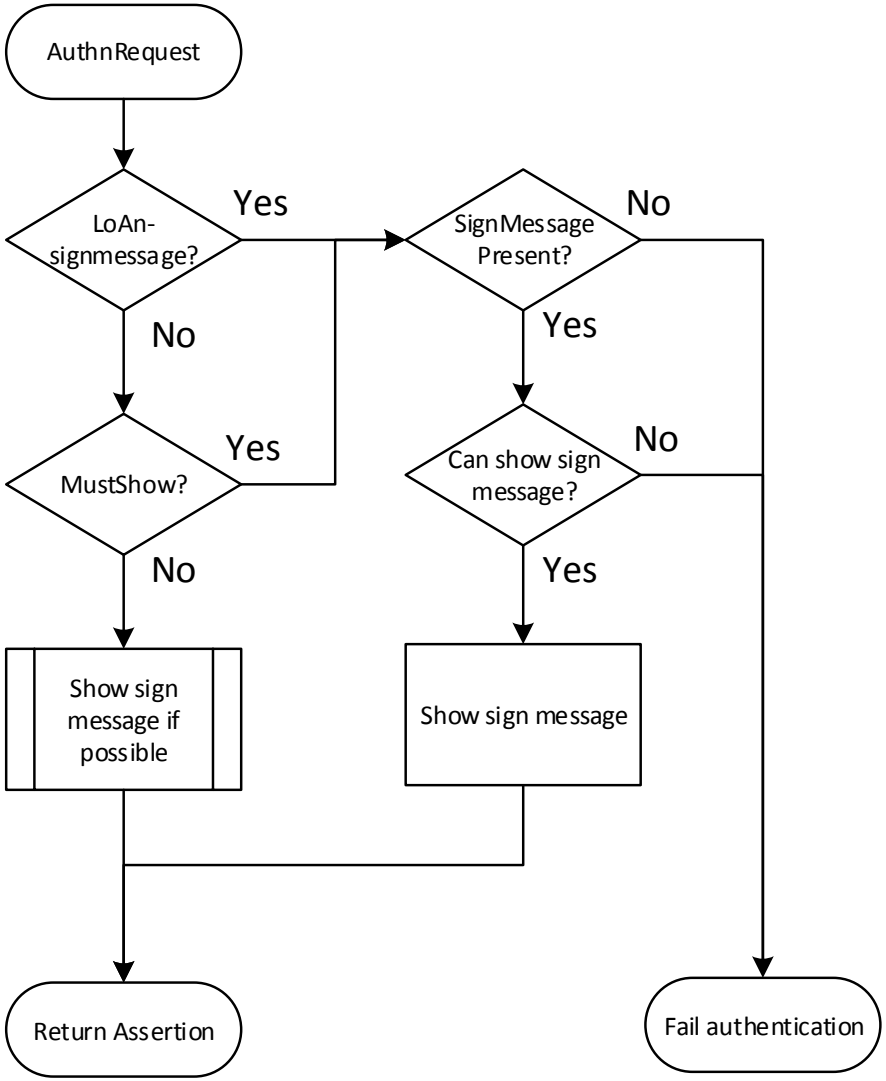
Implementering i protokoll

- Eid2 DSS Extension
  - Elementet SignMessage ingår i SignRequestExtension
  - Elementet uppdaterat i version 1.1
  - Krav på AuthnContextClassRef infogas i elementet CertRequestProperties.
- SAML AuthnRequest
  - Elementet SignMessage infogas som extension i Extensions elementet oförändrat så som det mottogs i sign request.
  - Krav på AuthnContextClassRef infogas som attribut i request
- SAML Assertion
  - AuthnContextClassRef i Assertion bekräftar att Legitimeringstjänsten har stöd för autentisering med visning av sign message och att sådan visning skett.

Beslutsflöde - Signeringstjänst



Beslutsflöde - Legitimeringstjänst



Kompatibilitetsanalys

- E-tjänst
  - möjlighet att använda standard produkter påverkas inte (implementeras i stödtjänst för underskrift).
- Underskriftstjänst
  - AuthnRequest kan inte skapas av Shibboleth SP (ej support för extensions)
  - OpenSAML kan användas för att skapa request samt för att validera response från legitimeringstjänst.
  - Ingen implementering av anvisning eller SSO gör detta relativt enkelt.
- Legitimeringstjänst
  - Shibboleth IdP version 2 kan inte användas (ingen access till request extension)
  - Shibboleth IdP version 3 kan användas. Testimplementerat.

Meddelandeformat

- Restricted HTML
  - Tillåtna HTML taggar (TAG[attr,…])
    - div[style], span[style], p[style], b[style], u[], i[], br[], strong[style], table[style], tr[style], td[style]
  - Tillåtna entities (ex   < etc)
    - amp (&), gt (>), lt (<), quot ("), nbsp ( )
  - Övrigt:
    - Character encoding = UTF-8;
    - Syntax kontroll/rättelse (Inga öppna taggar)
- Allt annat är förbjudet
  - Ex: script, länkar (<a>) och bilder (<img>) som hämtar data från extern källa. Inga kommentarfält (<!-- -->)
- Ger e-tjänsten begränsad möjlighet att ”styla” meddelandet
- Reglerna förutsätter att display sker mot vit bakgrund.

Övriga krav – Underskriftstjänster

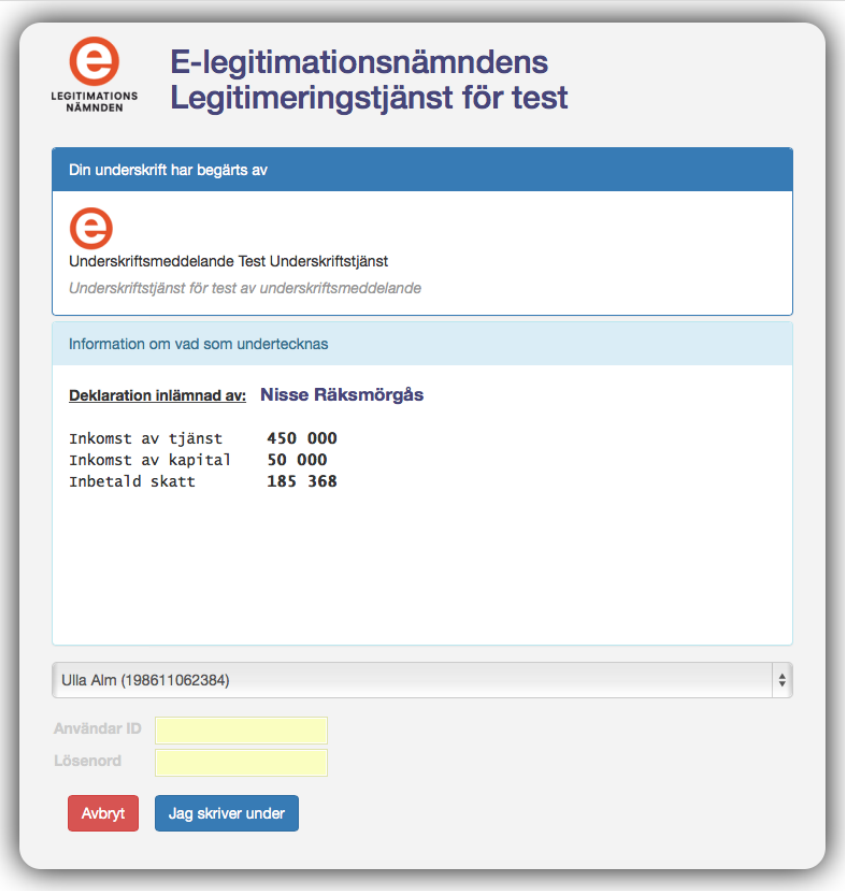
- Underskriftstjänster SKALL begära legitimering med signerade AuthnRequest som SKALL skickas enligt HTTP POST binding.
- Underskriftstjänster skall i sin metadata inkludera följande attribut i sin SPSSODescriptor:
  - AuthnRequestsSigned="true“
  - WantAssertionsSigned="true“
    - Sign Response skall innehålla en signed Assertion. Detta för att förhindra att bara response är signerat.
- AuthnRequest för “http://id.elegnamnden.se/loa/1.0/loa3-sigmessage” SKALL alltid inkludera attributet ForceAuthn=“true“

Övriga krav - Legitimeringstjänster

- Skall i metadata stödja SingleSignOnService med Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST”
- Skall i metadata inkludera EntityAttribute urn:oasis:names:tc:SAML:attribute:assurance-certification med stöd för nya AuthnContextClassRef för autentiseringsflöde med visning av sign message.
- Skall inte acceptera SSO inloggning för request som anger ClassRef för autentisering med visning av signmessage (även om ForceAuthn ej är satt till ”true”).

Testimplementering

- https://eid.svelegtest.se/shibv3testsp/start



Öppna frågor

- Krav på meddelandeformat – Balans mellan funktion och enkelhet (HTML vs Text)
- Synpunkter på protokollelement och extensibilitet
- Behov av explicit kvittens i identitetsintyg av visat meddelande? Alternativ:
  - Föreslagen lösning = Endast deklarera AuthnContextClassRef för sign message visning
    - Enklast och tämligen komplett då sign message finns med i slutlig sign response (i sign request elementet).
  - Hash av Message data från e-tjänst
    - Tillför inte mycket och fortfarande inte entydigt vad som visades efter filtrering/rättning.
  - Hela faktiskt visade meddelande efter filtrering/rättning.
    - Mest vattentätt men mer data och komplexitet.
- Övrigt?