MedCom

Den Gode Webservice
Bilag

Version 1.0 - 13-07-2006

Bilag 1: Digital signering med XML

Den Gode Webservice anvender XMLSignature-standarden http://www.w3.org/TR/xmldsig-core/ til at indlejre digitale signaturer, der er lavet med private nøgler, der hører til OCES-certifikater. Dette bilag gennemgår, hvordan en digital signatur af XML-elementer skabes og indlejres i Den Gode Webservice, og hvordan signaturen efterfølgende valideres igen. Eksemplet tager udgangspunkt i et id-kort på niveau 4.

En webserviceklient ønsker at kalde en webserviceudbyder, og udbyderen kræver sikkerhedsniveau 4, dvs. at der skal vedlægges et id-kort med en MOCES digital signatur. Klienten starter derfor med at danne SOAP DGWS-kuverten og indlejre id-kortet i ikke-underskrevet form. Det illustreres i nedenstående eksempel (<soap:body> er tom af hensyn til læsbarheden):

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope …>
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2006-06-01T08:01:00Z</wsu:Created>
   </wsu:Timestamp>
   <saml:Assertion IssueInstant="2006-06-01T07:53:00Z" Version="2.0" id="IDCard">
    <saml:Issuer>LægeSystemA</saml:Issuer>
    <saml:Subject>
     <saml:NameID Format="medcom:cprnumber">2606444917</saml:NameID>
     <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-
key</saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
       <ds:KeyInfo>
        <ds:KeyName>OCESSignature</ds:KeyName>
       </ds:KeyInfo>
      </saml:SubjectConfirmationData>
     </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Conditions NotBefore="2006-06-01T08:00:00Z" NotOnOrAfter="2006-07-01T07:53:00Z"/>
    <saml:AttributeStatement id="IDCardData">
     <saml:Attribute Name="sosi:IDCardID">
      <saml:AttributeValue>AAATX</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardVersion">
      <saml:AttributeValue>1.0</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardType">
      <saml:AttributeValue>user</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:AuthenticationLevel">
      <saml:AttributeValue>4</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:OCESCertHash">
      <saml:AttributeValue>ALiLaerBquie1/t6ykRKqLZe13Y=</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <saml:AttributeStatement id="UserLog">
     <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
      <saml:AttributeValue>2606444917</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserGivenName">
      <saml:AttributeValue>Ole H.</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserSurName">
      <saml:AttributeValue>Berggren</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserEmailAddress">
      <saml:AttributeValue>ohb@nomail.dk</saml:AttributeValue>
```

```
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserRole">
       <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserOccupation">
       <saml:AttributeValue>Maskinarbejder</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserAuthorizationCode">
       <saml:AttributeValue>24778</saml:AttributeValue>
      </saml:Attribute>
     </saml:AttributeStatement>
     <saml:AttributeStatement id="SystemLog">
      <saml:Attribute Name="medcom:ITSystemName">
       <saml:AttributeValue>LægeSystemA</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:ynumber">
       <saml:AttributeValue>079741</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:CareProviderName">
       <saml:AttributeValue>Lægehuset, Vandværksvej</saml:AttributeValue>
      </saml:Attribute>
     </saml:AttributeStatement>
    </saml:Assertion>
   </wsse:Security>
   <medcom:Header>
    <medcom:SecurityLevel>4</medcom:SecurityLevel>
    <medcom:Linking>
     <medcom:FlowID>AMRRMD</medcom:FlowID>
     <medcom:MessageID>AGQ5ZW</medcom:MessageID>
    </medcom:Linking>
    <medcom:RequestPriority>RUTINE</medcom:RequestPriority>
   </medcom:Header>
  </soap:Header>
  <soap:Body/>
</soap:Envelope>
```

Signeringen foregår nu i følgende faser. Resten af bilaget beskriver processen i detaljer:

1) Opret al XML i <ds:Signature> (dog uden indhold endnu)
2) Indsæt indhold i <ds:Reference>-elementet
    a. Udpeg det XML, der skal underskrives (id-kortet)
    b. Transformér id-kortet ved at ”kanonisere det” med C14N-algoritmen
    c. Beregn et SHA-1-”fingeraftryk” (digest) af det transfomerede id-kort, base64-konvertér fingeraftrykket og gem det i <ds:DigestValue>
3) Opret indhold i <ds:SignedInfo> elementet
4) Signér <ds:SignedInfo>-elementet
    a. Kanonisér <ds:SignedInfo>-elementet med C14N-algoritmen
    b. Beregn et SHA-1-”fingeraftryk” af det kanoniserede <ds:SignedInfo>-element
    c. Krypter fingeraftrykket med den private RS-nøgle, der hører til OCES-certifikatet
    d. Base64-konvertér signaturen og gem den i <ds:SignatureValue>
5) Base64-konvertér OCES-certifikatet og gem det i <ds:KeyInfo>
6) Indsæt <ds:Signature> i den oprindelige XML’s id-kort som sidste element i kuverten.

#1: Opret <ds:Signature> XML

XML-digitale signaturer indlejres i et <ds:Signature>-element, hvor ”ds” angiver namespacet ” http://www.w3.org/2000/09/xmldsig#”. Det første skridt til at oprette en XML-signatur er derfor at oprette de nødvendige XML-elementer.

En XML-digital signatur, som anvendes i Den Gode Webservice, har følgende grundstruktur ("…" angiver de værdier, der efterfølgende skal udfyldes):

```
<ds:Signature>
 <ds:SignedInfo>
  <ds:CanonicalizationMethod Algorithm="…"/>
  <ds:SignatureMethod Algorithm="…"/>
  <ds:Reference URI="…">
   <ds:Transforms>
    <ds:Transform Algorithm="…"/>
    <ds:Transform Algorithm="…"/>
   </ds:Transforms>
   <ds:DigestMethod Algorithm="…"/>
   <ds:DigestValue>…</ds:DigestValue>
  </ds:Reference>
 </ds:SignedInfo>
 <ds:SignatureValue>…/ds:SignatureValue>


 <ds:KeyInfo>
  <ds:X509Data>
   <ds:X509Certificate>…</ds:X509Certificate>
  </ds:X509Data>
 </ds:KeyInfo>


</ds:Signature>
```

SignedInfo udpeger de data, der er digitalt signeret vha. referenceelementer. Hver reference indeholder et beregnet "fingeraftryk" (digest) og angiver evt. transformationer, der skal foretages på kildeelementerne, inden dette digest kan beregnes.

Den digitale signatur laves af SignedInfo-elementet.
Nøgleinformation, der kan anvendes til at validere den digitale signaturs gyldighed. I DGWS indlejres det OCES-certifikat, der indeholder en nøgle, som kan verificere signaturen.

#2: Indsæt indhold i <ds:Reference> XML

XML-signaturespecifikationen tillader et <ds:Signature>-element at underskrive mere end ét XML-element. De elementer i kilden, der skal underskrives, udpeges vha. "id"-attributter, som angives i <ds:Reference>-elementer.

#2.a: Udpeg det XML, der skal underskrives

I dette tilfælde skal der laves en digital signatur af det id-kort, der er indlejret i DGWS-kuverten. Dette id-kort er en <saml:Assertion> med id="IDCard". Elementet udpeges via attributten URI:

```
<ds:Reference URI="#IDCard">
 …
</ds:Reference>
```

#2.b: Angiv transformationer af XML-kilden

XML kan skrives på flere måder og stadig have samme betydning, f.eks. ved at lave mellemrum mellem tags, ved at anvende en forkortet form af tags uden indhold, f.eks.
 i stedet for
</br> o.lign.

Denne fleksibilitet er nyttig, når man vil fastholde betydningen af XML-dokumenter, men skidt, når man vil danne en digital signatur. Signaturen beregnes nemlig ved at fortolke XML som en strøm af bytes, og derfor vil et ekstra mellemrum ændre signaturen, selvom betydningen af indholdet er den samme!

For at sikre, at modtageren af signaturen er i stand til at validere den, er det derfor nødvendigt at transformere det XML, der skal underskrives til en entydig form, som kan genskabes af modtageren. Denne form angives af <ds:Transform>-elementet med

Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315". Det vil altså sige, at det <saml:Assertion>-element, der har id=”IDCard”, skal transformeres til ”kanonisk form” som angivet af Canonical XML 1.0-specifikationen (se http://www.w3.org/TR/xml-c14n):

```
<ds:Reference URI="#IDCard">
 <ds:Transforms>
  <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
  <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
 </ds:Transforms>
 …
</ds:Reference>
```

XML-signaturstandarden tillader tre forskellige måder at tilknytte en signatur til det XML, den underskriver: Det kan indlejres i XML-kilden (Enveloped), det kan indlejre det data, der undskrives som et underelement af signaturelementet (Enveloping), eller det kan ligge ved siden af (Detached). Den Gode Webservice anvender *kun* Enveloped-signaturer, dvs. at <ds:Signature>-elementet skal indlejres i XML-kilden.

I eksemplet ovenfor er der endnu en <ds:Transform> i <ds:Transforms>-elementet, angivet med Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature". Denne transformation fortæller, at <ds:Signature>-elementet skal fjernes fra XML-dokumentet, inden signaturen valideres.

Transformationerne udføres i den rækkefølge, de forekommer i <ds:Transforms>, dvs. at først fjernes en evt. signatur fra kilden, og derpå kanoniseres kilden.

#2.c: Beregn et ”fingeraftryk” af det underskrevne

Næste skridt er at beregne det egentlige fingeraftryk (kryptografisk digest) af den transformerede udgave af kilden, som udpeget af URI-attributten. Den Gode Webservice anvender SHA-1 (se http://www.itl.nist.gov/fipspubs/fip180-1.htm), der laver digests med følgende egenskaber:

1) De har altid en fast længde på 160 bytes uanset kildens størrelse.
2) To forskellige beskeder giver altid forskellige digest-værdier.
3) Den samme besked giver altid den samme digest-værdi.
4) Man kan ikke genskabe kilden fra digest-værdien.

XML Signature tillader flere forskellige message digest algoritmer (MD5, HMAC, …), men Den Gode Webservice anvender kun SHA-1 (som angives i <ds:DigestMethod>-elementet):

```
<ds:Reference URI="#IDCard">
 <ds:Transforms>
  <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
  <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
 </ds:Transforms>
 <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
 <ds:DigestValue>…</ds:DigestValue>
</ds:Reference>
```

Et SHA-1-digest beregnes nu på den transformerede <saml:Assertion id=”IDCard”>, hvilket resulterer i 160 bytes. Et XML-dokument er altid forbundet med et bestemt tegnsæt, og det er et krav, at alle elementer i dokumentet følger dette. Derfor er det ikke muligt at indlejre de 160 bytes direkte i XML-dokumentet, men det er nødvendigt at transformere

dem til en form, der altid er den samme uanset det valgte tegnsæt i XML-dokumentet. Dette gøres med Base64-algoritmen (se http://www.ietf.org/rfc/rfc3548.txt).

Base64 algoritmen tager som input en mængde bytes og konverterer disse til en delmængde af ASCII-tegnsættet, som kun udnytter 65 forskellige tegn, dvs. 6 bit. Disse tegn har alle samme repræsentation, uanset det tegnsæt man har valgt, og kan derfor indlejres i alle XML-dokumenter. Digsten indsættes i <ds:DigestValue>:

```
<ds:Reference URI="#IDCard">
 <ds:Transforms>
  <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
  <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
 </ds:Transforms>
 <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
 <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
</ds:Reference>
```

#3: Opret indhold i <ds:SignedInfo> XML

Den XML, der underskrives, er ikke, som man måske skulle tro, den digest-værdi, der er beregnet af kilden, men hele <ds:SignedInfo>-elementet, som indeholder potentielt flere <ds:Reference>-elementer med digest-værdier.

For at kunne beregne den egentlige signatur skal der laves endnu et digest, denne gang over <ds:SignedInfo>. Som med den oprindelige kilde kan <ds:SignedInfo> repræsenteres på flere måder og skal derfor kanoniseres. Algoritmen til dette angives i <ds:CanonicalizationMethod>:

```
<ds:SignedInfo>
  <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
  <ds:SignatureMethod Algorithm="…"/>
  <ds:Reference URI="#IDCard">
   <ds:Transforms>
    <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
    <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
   <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
  </ds:Reference>
 </ds:SignedInfo>
```

Endelig skal <ds:SignedInfo> angive, hvilken algoritme der skal bruges til at danne den endelige signatur. Den Gode Webservice anvender OCES digitale certifikater, som benytter RSA-nøgler. Samtidig anvendes SHA-1 til at danne Digsten, hvilket giver den kombinerede signatur-algoritme RSA-SHA1:

```
 <ds:SignedInfo>
  <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
  <ds:Reference URI="#IDCard">
   <ds:Transforms>
    <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
    <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
   <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
  </ds:Reference>
 </ds:SignedInfo>
```

#4: Underskriv <ds:SignedInfo>

Den endelige digitale signatur skabes nu ved at:

a) kanonisere <ds:SignedInfo>
b) lave et SHA-1-digest af den kanoniserede XML
c) kryptere de 160 bytes med den private nøgle, der hører til MOCES-certifikatet
d) base64-kode de krypterede bytes.

Resultatet indlejres i <ds:SignatureValue>-elementet:

```
<ds:Signature>
 <ds:SignedInfo>
  …
 </ds:SignedInfo>
 <ds:SignatureValue>
  BaWKC9PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8GqMJh6gh/cA2mgKJ9b
  qiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2dochINlFg=
 </ds:SignatureValue>
 <ds:KeyInfo>
  …
 </ds:KeyInfo>
</ds:Signature>
```

#5: Gem certifikatet i <ds:KeyInfo>

Når signaturen skal valideres, skal modtageren være i besiddelse af den offentlige nøgle, der kan dekryptere signaturen. Nøglen er indlejret i MOCES-certifikatet, der bl.a. også indeholder et unikt OCES-id, information om, hvem der har udstedt det (TDC), hvem det er udstedt til mv.

Man kunne også vælge blot at indlejre RSA-nøglen i signaturen for at spare plads, men så ville det ikke være muligt at fastslå afsenderens identitet via opslag på OCES-id'et mod TDC's webservices.

Certifikatet base64-kodes inden indlejring og gemmes i <ds:X509Certificate>-elementet, der igen indlejres i <ds:X509Data> i <ds:KeyInfo>-elementet. Den fulde signatur bliver dermed:

```
<ds:Signature>
 <ds:SignedInfo>
  <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
  <ds:Reference URI="#IDCard">
   <ds:Transforms>
    <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
    <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
   <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
  </ds:Reference>
 </ds:SignedInfo>
 <ds:SignatureValue>
  BaWKC9PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8GqMJh6gh/cA2mgKJ9b
  qiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2dochINlFg=
 </ds:SignatureValue>
 <ds:KeyInfo>
  <ds:X509Data>
   <ds:X509Certificate>
    MIIFBDCCBG2gAwIBAgIEQDZLNzANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UENFUyBTeXN0ZW10Z
    XN0IENBIElJMB4XDTA1MDYwNjEyMDQwMDYwNjEyMzQwMFowfTELMAkGA1UEBhMCREsxLzAtBgNVBAoUJlREQyBUT1RB
    TEzYU05JMgLy8gQ1ZSOjI1NzY3NTM1MT0wFAYDVQQDEw1UZXN0IEJydWdlciAyMCUGA1UEBRMeZSOjI1NzY3NTM1LVJ
```

```
    JRDoxMTE4MDYxMDQzMzU2MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBuvze+4T1i0inhmvafWB2d81q3AG7ds06eG
    y+eLjQYumaY5EViSv4qyNwmnV6Y1sVi3LpD//wr7+DBanwBUEXnlzRY4No4U3DrDAjvl4NKjdv/Dkg1pMfUwmaIYkQo
    LTWHe8bCfVPXtovQ12CLO7uydoBzTQIDAQABo4ICzTCCAskwDgYDVR0PAQH/BAQDAgP4MCsGA1UdEAQkMCKAMTIwNDA
    wWoEPMjAwNzA2MDYxMjM0MDBaMEYGCCsGAQUFBwEBBDowODA2BggrBgEFcDovL3Rlc3Qub2NzcC5jZXJ0aWZpa2F0Lm
    RrL29jc3Avc3RhdHVzMIIBAwYDVR0gMIH4MIH1BgkpAQEBAQEBAQIwgecwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cuY
    2VydGlmaWthkay9yZXBvc2l0b3J5MIGzBggrBgEFBQcCAjCBpjAKFgNUREMwAwIBARqBl1REQyBUZXN0IENlEgdWRzd
    GVkZXMgdW5kZXIgT0lEIDEuMS4xLjEuMS4xLjEuxLjIuIFREQyBUZXN0IENlcnRpZmljYXRlcyBmcm9tIHRoaXMgQ0E
    gYXJlIGlzc3VlZCB1bmRlxLjEuMS4xLjEuMS4xLjEuMi4wGgYJYIZIAYb4QgENBA0WC2VtcGxveWVlV2ViMCAGUdEQQ
    ZMBeBFXN1cHBvcnRAY2VydGlmaWthdC5kazCBlgYDVR0fBIGOMIGLMFagVKBSpFAwTjELUEBhMCREsxDDAKBgNVBAoT
    A1REQzEiMCAGA1UEAxMZVERDIE9DRVMgU3lzdGVtdGVzdCBDUEAxMEQ1JMMjAxoC+gLYYraHR0cDovL3Rlc3QuY3JsL
    m9jZXMuY2VydGlmaWthkay9vY2VzLmNybDAfBgNVHSMEGDAWgBQcmAlHGkw4uRDFBC1b8fROgGrMfjAdBgNVHQ4EFgQ
    UpQWIRbZKFhWkcHOi1bgdX4YwCQYDVR0TBAIwADAZBgkqhkiG9n0HQQAEDDAKGwRWNy4xAwIDw0BAQUFAAOBgQBp+zm
    RburdSGirxmMWFFcT4NaP3W+XRPqY3iCiZuW2FcBrTtHyuFrjbQHg9RznxAgHIpzu/txQsSqv+m76Ki8zB2+r0fwlYr
    ABvcloPUfRF6pRksYtYNXsnGSRe1147c9K315hXG3QMmuU+rBFyvRGkWx0wIf3lOrLg==
   </ds:X509Certificate>
  </ds:X509Data>
 </ds:KeyInfo>
</ds:Signature>
```

#6: Indsæt <ds:Signature> i den oprindelige XML

I Den Gode Webservice indlejres signaturen altid i den oprindelige XML (enveloped). Signaturen af id-kortet på niveau 3 og 4 indlejres lige efter det sidste <saml:AttributeStatement> i id-kortet. På niveau 5 er signaturen dannet over hele SOAP-kuverten og indlejres i dette tilfælde lige efter id-kortet.

Den oprindelige kuvert, der nu har signaturen indsat i id-kortets <saml:Assertion>-element, ses nedenfor:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope …>
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2006-06-01T08:01:00Z</wsu:Created>
   </wsu:Timestamp>
   <saml:Assertion IssueInstant="2006-06-01T07:53:00Z" Version="2.0" id="IDCard">
    <saml:Issuer>LægeSystemA</saml:Issuer>
    <saml:Subject>
     <saml:NameID Format="medcom:cprnumber">2606444917</saml:NameID>
     <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-
key</saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
       <ds:KeyInfo>
        <ds:KeyName>OCESSignature</ds:KeyName>
       </ds:KeyInfo>
      </saml:SubjectConfirmationData>
     </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Conditions NotBefore="2006-06-01T08:00:00Z" NotOnOrAfter="2006-07-01T07:53:00Z"/>
    <saml:AttributeStatement id="IDCardData">
     <saml:Attribute Name="sosi:IDCardID">
      <saml:AttributeValue>AAATX</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardVersion">
      <saml:AttributeValue>1.0</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardType">
      <saml:AttributeValue>user</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:AuthenticationLevel">
      <saml:AttributeValue>4</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:OCESCertHash">
      <saml:AttributeValue>ALiLaerBquie1/t6ykRKqLZe13Y=</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <saml:AttributeStatement id="UserLog">
```

```
 <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
  <saml:AttributeValue>2606444917</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:UserGivenName">
  <saml:AttributeValue>Ole H.</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:UserSurName">
  <saml:AttributeValue>Berggren</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:UserEmailAddress">
  <saml:AttributeValue>ohb@nomail.dk</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:UserRole">
  <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:UserOccupation">
  <saml:AttributeValue>Maskinarbejder</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:UserAuthorizationCode">
  <saml:AttributeValue>24778</saml:AttributeValue>
 </saml:Attribute>
</saml:AttributeStatement>
<saml:AttributeStatement id="SystemLog">
 <saml:Attribute Name="medcom:ITSystemName">
  <saml:AttributeValue>LægeSystemA</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:ynumber">
  <saml:AttributeValue>079741</saml:AttributeValue>
 </saml:Attribute>
 <saml:Attribute Name="medcom:CareProviderName">
  <saml:AttributeValue>Lægehuset, Vandværksvej</saml:AttributeValue>
 </saml:Attribute>
</saml:AttributeStatement>
<ds:Signature>
 <ds:SignedInfo>
  <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
  <ds:Reference URI="#IDCard">
   <ds:Transforms>
    <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
    <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
   </ds:Transforms>
   <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
   <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
  </ds:Reference>
 </ds:SignedInfo>
 <ds:SignatureValue>
  BaWKC9PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8GqMJh6gh/cA2mgKJ9b
  qiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2dochINlFg=
 </ds:SignatureValue>
 <ds:KeyInfo>
  <ds:X509Data>
   <ds:X509Certificate>
    MIIFBDCCBG2gAwIBAgIEQDZLNzANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UENFUyBTeXN0ZW10Z
    XN0IENBIElJMB4XDTA1MDYwNjEyMDQwMDYwNjEyMzQwMFowfTELMAkGA1UEBhMCREsxLzAtBgNVBAoUJlREQyBUT1RB
    TEzYU05JMgLy8gQ1ZSOjI1NzY3NTM1MT0wFAYDVQQDEw1UZXN0IEJydWdlciAyMCUGA1UEBRMeZSOjI1NzY3NTM1LVJ
    JRDoxMTE4MDYxMDQzMzU2MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBuvze+4T1i0inhmvafWB2d81q3AG7ds06eG
    y+eLjQYumaY5EViSv4qyNwmnV6Y1sVi3LpD//wr7+DBanwBUEXnlzRY4No4U3DrDAjvl4NKjdv/Dkg1pMfUwmaIYkQo
    LTWHe8bCfVPXtovQ12CLO7uydoBzTQIDAQABo4ICzTCCAskwDgYDVR0PAQH/BAQDAgP4MCsGA1UdEAQkMCKAMTIwNDA
    wWoEPMjAwNzA2MDYxMjM0MDBaMEYGCCsGAQUFBwEBBDowODA2BggrBgEFcDovL3Rlc3Qub2NzcC5jZXJ0aWZpa2F0Lm
    RrL29jc3Avc3RhdHVzMIIBAwYDVR0gMIH4MIH1BgkpAQEBAQEBAQIwgecwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cuY
    2VydGlmaWthkay9yZXBvc2l0b3J5MIGzBggrBgEFBQcCAjCBpjAKFgNUREMwAwIBARqBl1REQyBUZXN0IENlEgdWRzd
    GVkZXMgdW5kZXIgT0lEIDEuMS4xLjEuMS4xLjEuxLjIuIFREQyBUZXN0IENlcnRpZmljYXRlcyBmcm9tIHRoaXMgQ0E
    gYXJlIGlzc3VlZCBlbmRlxLjEuMS4xLjEuMS4xLjEuMi4wGgYJYIZIAYb4QgENBA0WC2VtcGxveWVlV2ViMCAGUdEQQ
    ZMBeBFXN1cHBvcnRAY2VydGlmaWthdC5kazCBlgYDVR0fBIGOMIGLMFagVKBSpFAwTjELUEBhMCREsxDDAKBgNVBAoT
    A1REQzEiMCAGA1UEAxMZVERDIE9DRVMgU3lzdGVtdGVzdCBDUEAxMEQ1JMMjAxoC+gLYYraHR0cDovL3Rlc3QuY3JsL
    m9jZXMuY2VydGlmaWthkay9vY2VzLmNybDAfBgNVHSMEGDAWgBQcmAlHGkw4uRDFBClb8fROgGrMfjAdBgNVHQ4EFgQ
    UpQWIRbZKFhWkcHOi1bgdX4YwCQYDVR0TBAIwADAZBgkqhkiG9n0HQQAEDDAKGwRWNy4xAwIDw0BAQUFAAOBgQBp+zm
    RburdSGirxmMWFFcT4NaP3W+XRPqY3iCiZuW2FcBrTtHyuFrjbQHg9RznxAgHIpzu/txQsSqv+m76Ki8zB2+r0fwlYr
    ABvcloPUfRF6pRksYtYNXsnGSRe1147c9K315hXG3QMmuU+rBFyvRGkWx0wIf3lOrLg==
   </ds:X509Certificate>
  </ds:X509Data>
 </ds:KeyInfo>
</ds:Signature>
```

```
    </saml:Assertion>
   </wsse:Security>
   <medcom:Header>
    <medcom:SecurityLevel>4</medcom:SecurityLevel>
    <medcom:Linking>
     <medcom:FlowID>AMRRMD</medcom:FlowID>
     <medcom:MessageID>AGQ5ZW</medcom:MessageID>
    </medcom:Linking>
    <medcom:RequestPriority>RUTINE</medcom:RequestPriority>
   </medcom:Header>
  </soap:Header>
  <soap:Body/>
</soap:Envelope>
```

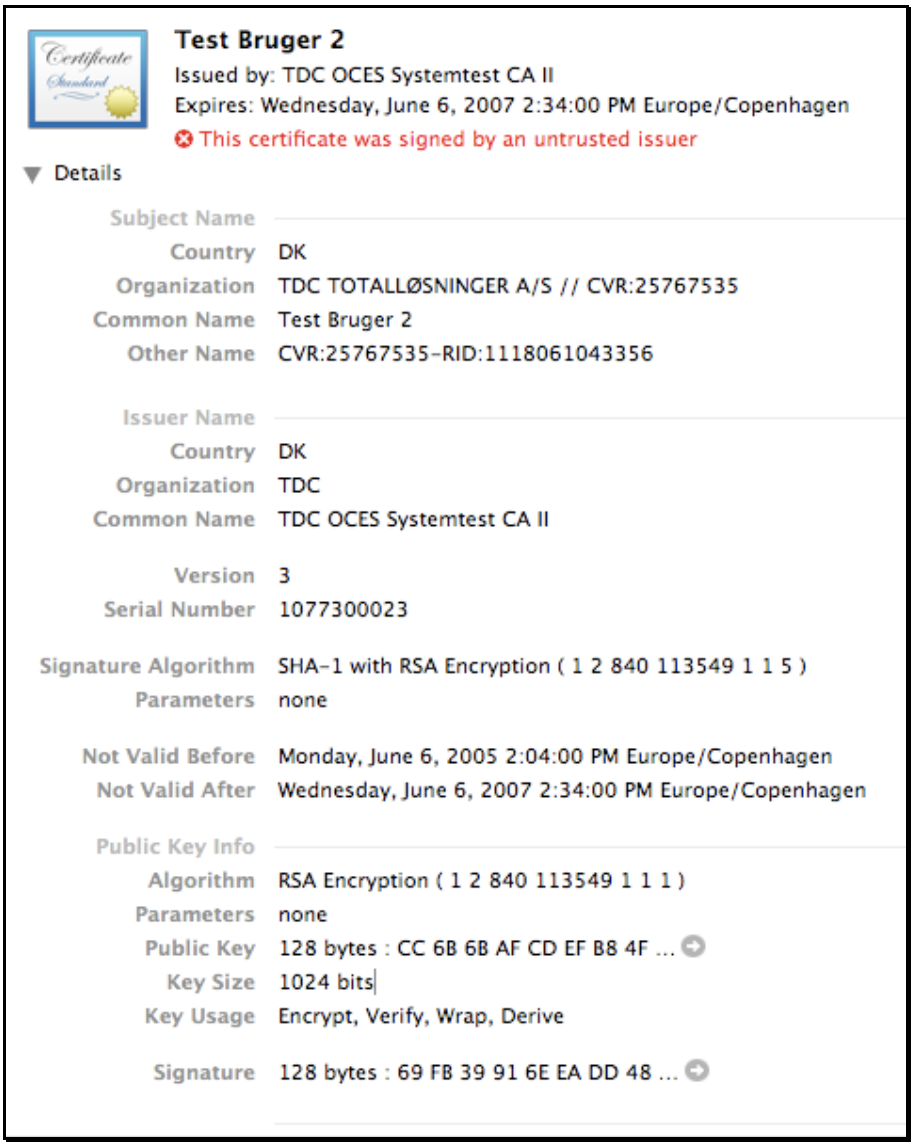
#7: Valider signaturen

For at validere signaturen skal man stort set gøre det samme, som når man skaber den. Modtageren skal gøre følgende:

1) Valider <ds:Reference>
    a. Udpeg det XML, hvis signatur der skal valideres (id-kortet).
    b. Transformer id-kortet ved at fjerne den indlejrede signatur fra XML (enveloped transform).
    c. Transformer id-kortet ved at "kanonisere det" med C14N-algoritmen.
    d. Beregn et SHA-1-"fingeraftryk" (digest) af det transfomerede id-kort.
    e. Base64-dekod <ds:DigestValue> og sammenlign med det beregnede fingeraftryk. De skal være ens.
2) Valider <ds:SignatureValue>
    a. Kanoniser <ds:SignedInfo-elementet med C14N-algoritmen.
    b. Beregn et SHA-1-fingeraftryk af det kanoniserede <ds:SignedInfo>-element.
    c. Base64-dekod OCES-certifikatet i <ds:KeyInfo> og find den offentlige nøgle.
    d. Dekrypter værdien af <ds:SignatureValue> med den offentlige nøgle.
    e. Sammenlign det beregnede "fingeraftryk" med det dekrypterede. De skal være ens.

#8: Valider certifikatet

Før eller efter signaturen er valideret (det er underordnet hvornår), skal certifikatets gyldighed tjekkes.OCES-certifikater indeholder information om, hvem der har udstedt dem, hvornår de er blevet udstedt og til hvem. Certifikatet indeholder også en række tekniske værdier. Eksemplet nedenfor viser information fra et TDC test-MOCES-certifikat, som er udstedt til brugeren "Test Bruger 2" fra organisationen "TDC TOTALLØSNINGER A/S" af certifikat-autoriteten (CA) "TDC OCES Systemtest CA II":



Før signaturen tjekkes, skal man først sikre sig, at certifikatet er gyldigt. Det gøres ved at undersøge følgende punkter:

1) **Er certifikatet udløbet**? I figuren ovenfor er certifikatets gyldighed angivet i felterne "Not Valid Before" og "Not Valid After". Hvis dags dato ikke er inden for denne periode, er certifikatet ikke gyldigt.
2) **Er certifikatet et validt OCES**? Certifikater kan udstedes af mange forskellige autoriteter, herunder Thawte, Verisign, mfl. Certifikatet ovenfor er udstedt af TDC's test-CA-facilitet, mens et produktionscertifikat vil være udstedt af TDC OCES CA. CA har lavet en digital signatur, der er angivet i feltet "Signature" nederst. Denne signatur valideres med CA's offentlige nøgle fra dennes certifikat, som forventes at være installeret på forhånd.
3) **Er certifikatet spærret**? Brugere, der forlader en organisation eller bliver forment adgang, kan blive spærret af en lokal certifikatautoritet (fra den organisation, der

står trykt i certifikatet). CA publicerer en liste over spærrede certifikater (CRL), som kan downloades og efterfølgende tjekkes op mod det medsendte certifikat. Alternativt kan man kalde en såkaldt OCSP- (Online Certificate Status Protocol) service for at høre, om certifikatet er blevet spærret. TDC understøtter begge metoder.

Bilag 2: Id-kortet - Sådan bruges SAML-standarden

I Den Gode Webservices requestbeskeder og i visse responsebeskeder findes oplysninger om bruger og system, herunder sikkerhedsoplysninger. Oplysningerne indlejres i en såkaldt SAML Assertion, der standardiserer denne type information.

I Den Gode Webservice kaldes denne SAML Assertion for beskedens id-kort, fordi det angiver afsenderens identitet i bred forstand. Id-kortet er en pendant til det fysiske id-kort, man anvender i mange virksomheder, og som dels identificerer indehaveren, dels giver denne adgang til forskellige afdelinger.

Ud over oplysninger om brugeren og systemet indeholder id-kortet akkreditiver, der bruges til autentifikation. Én type akkreditiv er det velkendte brugernavn og password, mens et andet er en OCES digital signatur.

Id-kortet består ud over akkreditiverne af følgende overordnede afsnit:

1. **saml:Subject**, som er SAML's måde at angive den bruger eller det system, som id-kortets andre attributter henfører til.
2. **saml:Conditions**, der i Den Gode Webservice benyttes til at angive id-kortets gyldighed, som er 24 timer efter udstedelsestidspunktet. Gyldigheden kan benyttes af webserviceudbydere, når det skal afgøres, om de kan have tillid til id-kortet.
3. **IDCardData**, der indeholder oplysninger om selve kortet, f.eks. hvornår det er udstedt og af hvem
4. **UserLog**, som rummer oplysninger om den bruger, id-kortet identificerer. Hvis id-kortet identificerer et system og ikke en enkelt bruger, udgår UserLog-elementet.
5. **SystemLog**, hvor oplysninger om afsendersystemet indsættes.

Denne figur illustrerer id-kortets overordnede struktur:

```
<saml:Assertion … id="IDCard">
  <saml:Subject>
    …
  </saml:Subject>
  <saml:Conditions NotBefore="2006-01-05T07:53:00.00"
                   NotOnOrAfter="2006-01-06T07:53:00.000"/>
  <saml:AttributeStatement id="IDCardData">
    …
  </saml:AttributeStatement>
  <saml:AttributeStatement id="UserLog">
    …
  </saml:AttributeStatement>
  <saml:AttributeStatement id="SystemLog">
    …
  </saml:AttributeStatement>
</saml:Assertion>
```

Kortoplysninger: IDCardData

Ethvert id-kort har et unikt id. Det angives som et løbenummer i attributten IDCardID og benytter en bestemt version af specifikationen, der angives i IDCardVersion. Versionsnummeret gør det muligt at udvide id-kortets datasæt i en senere udgave af Den Gode Webservice.

Id-kortet kan være udstedt til en medarbejder eller et system. Det angives i IDCardType som enten ”user” eller ”system”.

Afhængig af hvor stærke akkreditiver der blev anvendt, da id-kortet blev udstedt, kan dets AuthenticationLevel være fra 1 til 4, hvor 4 er stærkest. Hvis der blev anvendt et OCES-certifikat til autentifikation, indeholder OCESCertHash en hashværdi af det certifikat, der lå til grund. Se senere for en dybdegående beskrivelse af autentifikationsniveauer og akkreditiver.

Figuren nedenfor viser et IDCardData-element:

```
<saml:AttributeStatement id="IDCardData">
  <saml:Attribute Name="sosi:IDCardID">
    <saml:AttributeValue>1234</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:IDCardVersion">
    <saml:AttributeValue>1.0</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:IDCardType">
    <saml:AttributeValue>user</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:AuthenticationLevel">
    <saml:AttributeValue>3</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:OCESCertHash">
    <saml:AttributeValue>ALiLaerBquie1/t6ykRKqLZe13Y=</saml:AttributeValue>
  </saml:Attribute>
</saml:AttributeStatement>
```

Data om id-kortet, IDCardData

Brugeroplysninger: UserLog

Et id-korts vigtigste funktion er at levere oplysninger om dets indehaver ved identifikation, adgangskontrol, logning mv. Den sektion af id-kortet, der indeholder information om medarbejderen, findes i UserLog-elementet.

I UserLog findes oplysninger om personens navn, CPR-nummer og e-mail-adresse. Desuden indeholder elementet en rolle, der på sigt forventes at komme fra en national klassifikation. Brugeren er sundhedsfaglig (og har som sådan en autorisationskode fra Sundhedsstyrelsen) og kommer fra en sundhedsfaglig organisation, der angives med navn og en ”CareProviderID”-kode. Koden kan være et CVR-nummer, et P-nummer, en SKS-kode, et ydernummer, kommunenummer, lokationsnummer eller andet, som angivet i attributtens type.

Eksemplet nedenfor viser en UserLog for en fiktiv praktiserende læge, hvis organisation er angivet med et ydernummer:

```
<saml:AttributeStatement id="UserLog">
  <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
    <saml:AttributeValue>1903991234</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserGivenName">
    <saml:AttributeValue>Jens</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserSurName">
    <saml:AttributeValue>Hansen</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserEMailAddress">
    <saml:AttributeValue>jh@nomail.dk</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserRole">
    <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserOccupation">
    <saml:AttributeValue>Overlæge</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:AuthorizationCode">
    <saml:AttributeValue>1234</saml:AttributeValue>
  </saml:Attribute>
</saml:AttributeStatement>
```

Medarbejderinformationer

Systemoplysninger: SystemLog

Alle id-kort, uanset om de gælder for en medarbejder eller en applikation, skal anvendes gennem et klientsystem, som en evt. medarbejder så benytter. Oplysninger om dette system findes i SystemLog:

```
<saml:AttributeStatement id=" SystemLog">
  <saml:Attribute Name="medcom:ITSystemName">
    <saml:AttributeValue>LægeSystemet 3.0</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:CareProviderID"
                  NameFormat="urn:medcom:names:careprovider:ynumber">
    <saml:AttributeValue>123456</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:CareProviderName">
    <saml:AttributeValue>Hansens Lægepraksis</saml:AttributeValue>
  </saml:Attribute>
</saml:AttributeStatement>
```

Oplysninger om afsendersystemet

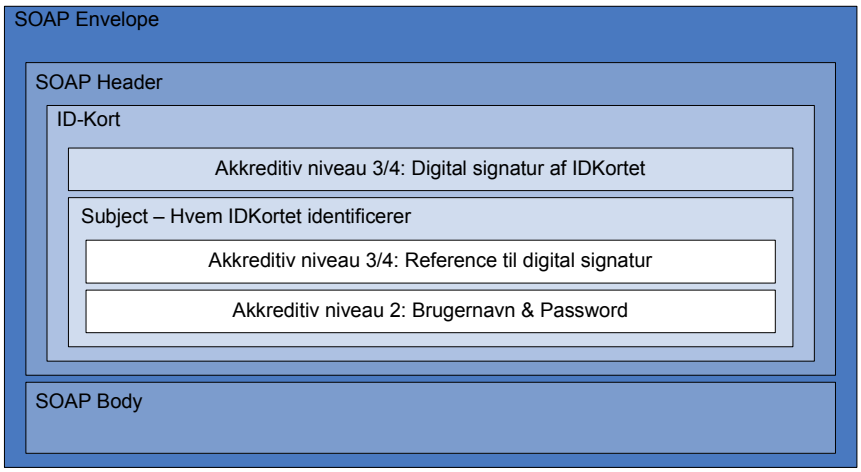
I klassiske autentifikationsscenarier sendes akkreditiverne kun, når der skal logges på, og der sendes ikke egentlige oplysninger om brugeren og systemet, som det er tilfældet i Den Gode Webservice. Den part, der foretager autentifikationen, kan så selv associere aftageren med flere oplysninger, f.eks. baseret på et bagvedliggende brugerkatalog eller lignende.

I Den Gode Webservice er den gængse autentifikationsmodel udvidet en anelse, idet der sammen med akkreditiver også sendes egentlige oplysninger om slutbrugeren, og det system vedkommende kom fra. Det er så op til webserviceudbyderen og/eller

identitetsservicen at validere disse påstande eller at stole på dem. Modellen gør det altså muligt at kommunikere et fælles sæt oplysninger med forskellige grader af pålidelighed.

I det mest pålidelige scenario har en klient eller en identitetsservice tjekket digitalt signerede brugerinformationer mod bagvedliggende centrale registre. I det mindst pålidelige scenario er der ingen akkreditiver, og modtageren må stole blindt på afsenderens oplysninger i id-kortet.

Figuren nedenfor viser id-kortet og de mulige akkreditiver, der kan indsættes i og sammen med det:



Id-kortet og akkreditiver i en DGWS-besked

Akkreditiv Niveau 2: Brugernavn og password. Angiver et brugernavn og password, som webserviceudbyder eller identitetsservice kan verificere. Dette akkreditiv er mindre stærkt end digitale signaturer, men bedre end ingen (niveau 1).

Akkreditiv Niveau 3 og 4: OCES digital signatur. Når der er behov for en høj grad af tillid til brugerens identitet, anvendes digital signatur i stedet. Det kan enten være VOCES (niveau 3) eller MOCES (niveau 4).

Niveau 1: Ingen akkreditiver

På niveau 1 medsendes ingen akkreditiver i id-kortet, som derfor udelukkende indeholder oplysninger om kort, medarbejder og system, men ingen mulighed for at verificere identiteten.

Niveau 2: Brugernavn og password

Når der anvendes brugernavn og password til autentifikation, indlejres denne information som et webservice security (wsse) UsernameToken-element inde i subject-elementet. Webserviceudbyder anvender brugernavn og password til at verificere brugerens identitet, og informationen lagres derfor i ”SubjectConfirmationData”, som vist på figuren nedenfor:

```
<saml:Assertion … id="IDCard">
  <saml:Subject>
    <saml:NameID
="HTTP://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/CPR_PersonCivilRegistrati
onIdentifier.xsd">1903701234</saml:NameID>
```

```
    <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-
key</saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
        <wsse:UsernameToken>
          <wsse:Username>epmui01</wsse:Username>
          <wsse:Password>dfh1241</wsse:Password>
        </wsse:UsernameToken>
      </saml:SubjectConfirmationData>
    </saml:SubjectConfirmation>
  </saml:Subject>
  …
</saml:Assertion>
```

Brugernavn og password som akkreditiver

Brugernavn angives i wsse:Username, mens password angives i wsse:Password. For begge felter gælder det, at informationen altid lagres i klar tekst.

I figuren ovenfor findes ud over brugernavn og password et NameID, der angiver den bruger, id-kortet identificerer. I dette tilfælde er brugeren angivet som CPR-nummeret på medarbejderen, ” 1903701234”, og i OIO-formatet for CPR-numre.

Niveau 3 og 4: Digital signatur

På niveau 3 og 4 anvendes digitale signaturer til at verificere et system eller en bruger. OCES-standarden definerer VOCES-certifikatet til identifikation af virksomheder og MOCES til identifikation af medarbejdere. Det er disse to OCES-certifikattyper, Den Gode Webservice anvender: Niveau 3 anvender VOCES, mens niveau 4 anvender MOCES.

På niveau 3 og 4 er akkreditivet en digital signatur af id-kortet, der er dannet med den private nøgle, som kun afsenderen har adgang til. Signaturen er i de fleste tilfælde selvindeholdt, dvs. at det certifikat, der hører til, også er indlejret i signaturen, så det er nemt at validere. Hvis der anvendes en identitetsservice, kan man dog vælge at indlejre certifikatets serienummer i stedet for, hvis man vurderer, at identitetesservicens certifikat er kendt af alle i føderationen.

De digitale signaturer indlejres i selve id-kortet efter kort-, bruger- og systemoplysningerne, som vist på figuren nedenfor. Bemærk, at signaturer, attributter mv. er forkortet med ”...” af hensyn til læsbarheden:

```
<saml:Assertion … id="IDCard">
  …
  <ds:Signature wsu:id="OCESSignature">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod
           Algorithm="HTTP://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod
           Algorithm="HTTP://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <ds:Reference URI="#IDCard">
        <ds:DigestMethod Algorithm="HTTP://www.w3.org/2000/09/xmldsig#sha1"/>
        <ds:DigestValue>ALiLaerBquie1/t6ykRKqLZe13Y=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>/jOaG4RVdBKBKpB5q2…</ds:SignatureValue>
```

```
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:X509Certificate>
          /wMwDQYJKoZIhvcNAQ…
        </ds:X509Certificate>
      </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
  <saml:Subject>
    <saml:NameID>1903701234</saml:NameID>
    <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>
        urn:oasis:names:tc:SAML:2.0:cm:holder-of-key
      </saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
        <ds:KeyInfo>
          <ds:KeyName>OCESSignature</ds:KeyName>
        </ds:KeyInfo>
      </saml:SubjectConfirmationData>
    </saml:SubjectConfirmation>
  </saml:Subject>
</saml:Assertion>
```

Digitale signaturer som akkreditiver

Det indlejrede saml:Subject-element indeholder en reference til signaturen, hvor værdien af ”ds:KeyName”-elementet, ”OCESSignature”, matcher id’et på ds:Signature-elementet.

Autentificerede id-kort

Normalt skaber klienten selv id-kortet og sender det og egne akkreditiver med i hvert eneste kald. Hvis der er behov for niveau 3- eller 4-autentifikation, er det altså klientens digitale signaturer med MOCES og/eller VOCES, der medsendes.

I SingleSignon-scenariet forholder sagen sig lidt anderledes. Her vil identitetsservicen validere id-kortet, fjerne de gamle akkreditiver og selv underskrive det med sin egen private VOCES-nøgle, inden det returneres til klienten.

Et autentificeret id-kort vil altså aldrig indeholde brugernavn og password eller MOCES-signatur, men altid en VOCES-signatur fra den identitetsudbyder, der foretog autentifikationen. Figuren nedenfor viser, hvordan id-kortet ser ud efter autentifikationen:

```
<saml:Assertion … id="IDCard">
  <saml:Subject>
    <saml:NameID
Format="HTTP://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/CPR_PersonCivilRegi
strationIdentifier.xsd">1903701234</saml:NameID>
    <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>
        urn:oasis:names:tc:SAML:2.0:cm:holder-of-key
      </saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
        <ds:KeyInfo>
          <ds:KeyName>OCESSignature</ds:KeyName>
        </ds:KeyInfo>
      </saml:SubjectConfirmationData>
```

```
    </saml:SubjectConfirmation>
  </saml:Subject>
  ...
  <ds:Signature wsu:id="OCESSignature">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod
           Algorithm="HTTP://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod
           Algorithm="HTTP://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <ds:Reference URI="#IDCard">
        <ds:DigestMethod Algorithm="HTTP://www.w3.org/2000/09/xmldsig#sha1"/>
        <ds:DigestValue>ALiLaerBquie1/t6ykRKqLZe13Y=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>/jOaG4RVdBKBKpB5q2…</ds:SignatureValue>
    <ds:KeyInfo>
      <ds:X509Data>
        <ds:KeyName>2F9B7C21</ds:KeyName>
      </ds:X509Data>
    </ds:KeyInfo>
  </ds:Signature>
</saml:Assertion>
```

SingleSignon id-kort

I autentificerede id-kort kan man i Den Gode Webservice nøjes med at indsætte en reference til det certifikat, man har anvendt til at validere underskriften med, i KeyInfo/KeyName-feltet. I eksemplet ovenfor angives værdien ”2F9B7C21” for KeyName, som er serienummeret på identitetsservicens certifikat. Certifikatet antages at være kendt af alle klienter.

Bilag 3: Usecase-eksempler

Nedenfor er der eksempler på, hvordan man kan bruge Den Gode Webservice. Først er der et uddybet eksempel på en webservice for laboratoriesvar og derefter kortere eksempler på, hvordan man ellers kan anvende Den Gode Webservice. Eksemplerne er kun fiktive, og de demonstrerer dermed ikke faktiske webservices.

Eksempel 1: Laboratoriesvar-webservice

I dag sendes næsten alle laboratoriesvar direkte til rekvirenten ved hjælp af MedComs standarder for laboratoriesvar. I visse tilfælde kan det imidlertid være relevant for speciallæger og andre sundhedsfaglige personer at have adgang til laboratoriesvar, som andre har rekvireret. Det kan de få, hvis laboratoriet udbyder en webservice for laboratoriesvar.

Sikkerhedslog

Udbyderen af laboratoriesvar er dataansvarlig for laboratoriesystemet og skal derfor føre og kontrollere en sikkerhedslog, der viser, hvilke brugere der har set hvilke oplysninger om hvilke patienter på hvilket tidspunkt.

Derfor skal klientsystemet i hver requestmeddelelse fremsende brugeren id, navn og organisation, f.eks.: 140793-1566, overlæge Hans Hansen, Bispebjerg Hospital.

På baggrund af brugeroplysningerne skal laboratoriet føre en log-liste, der for hver responsemeddelelse indeholder følgende oplysninger:

- responsemeddelelsens navn (”GetLabReportsResponse”), meddelelse-id, flow-id og afsendelsestidspunkt (”created”)
- brugerens id, navn og organisation
- patientens CPR-nummer og navn.
- liste over hvert laboratoriesvars dato og id.

Laboratoriesystemet skal på opfordring kunne finde de konkrete laboratoriesvar, der blev sendt til brugeren, på listen.

Samtykke

Inden brugeren får adgang til patientens laboratorieresultater, skal brugeren imidlertid have patientens samtykke til at se oplysningerne.

Brugeren skal derfor først i webservicedialogen vælge et af tre typer samtykke: ”samtykke”, ”bevidstløs” eller ”patienten er i aktuel behandling”. Ved de to sidste skal begrundelsen uddybes i et tekstfelt, f.eks. sådan

```
<medcom:PatientConsent>
<medcom:PatientConsentCode>I_BEHANDLING</medcom:PatientConsentCode>
<medcom:PatientConsentRemark>Patienten har brækket benet på ferie her i området. Derfor har jeg ikke
set  patienten før.</medcom:PatientConsentRemark>
</medcom:PatientConsent>
```

Samtykket skrives i forespørgslens XML-element <PatientConsentCode> på formen ”SAMTYKKE_GIVET”, ”PATIENT_BEVISTLOES” eller ”I_BEHANDLING”.

Hvis ingen af de tre typer samtykker afkrydses, returneres en faultmeddelelse med id ”No_Patient_Consent” og teksten ”Patientsamtykke foreligger ikke. Indhent patientsamtykke og fremsend forespørgslen igen”.

Webserviceflow
Laboratoriesvar-webservicen vil typisk bestå af to kald: ”FindLabReports” og ”GetLabReports”:

- FindLabReports: Klientsystemet søger ved hjælp af patientens CPR-nummer alle laboratoriesvar hos laboratoriet.

- GetLabReports: Klientsystemet udvælger de/det ønskede laboratoriesvar og modtager svaret fra laboratoriet. Selve laboratoriesvarene er indlejret i body-elementet og er kodet som almindelige MedCom XML-laboratoriesvar.

FindLabReports-forespørgslen kan typisk indeholde fire forespørgselsparametre, der også vil kunne anvendes ved mange andre søgninger: ”CPR-nummer”, ”start- og slutdato” og ”maks. X seneste sager”.

XML-koden indsættes i FindLabReports-forespørgslen i SOAP body (evt. sammen med anden information) og kan have følgende syntaks:

```
<medcom:RequestParameters>
<medcom:RequestCaseID>2208700289</medcom:RequestCaseID>
<medcom:RequestStartDateTime>2004-01-01T00:00:00</medcom:RequestStartDateTime>
<medcom:RequestEndDateTime>2005-01-01T00:00:00</medcom:RequestEndDateTime>
<medcom:RequestLatestCases>10</medcom:RequestLatestCases>
</medcom:RequestParameters>
```

Denne søgning vil for det pågældende CPR-nummer (RequestCaseID) resultere i en responsemeddelelse der indeholder de seneste 10 laboratoriesvar, der er produceret i perioden 1/1 2004 til 1/1 2005.

De benyttede felter anvendes sådan:

RequestCaseID	Repeateable	RequestCaseID er det data, der benyttes for at finde den aktuelle "case", f.eks. et sagsID ("LÆ221" for læ-blanket LÆ221) eller et CPR-nummer ("1405402199" for en patient) (CPR-nummer). RequestCaseID-elementet kan gentages.
RequestStartDateTime	dateTime	"Fra"-dato og klokkeslæt til brug ved yderligere afgrænsning af søgninger. Angives på UTC-formen YYYY-MM-DDTHH:MMZ. Kan udelades.
RequestEndDateTime	dateTime	"Til"-dato og klokkeslæt til brug ved yderligere afgrænsning af søgninger. Angives på UTC-formen YYYY-MM-DDTHH:MMZ. Kan udelades.
RequestLatestCases	string	Kode, der benyttes til yderligere afgrænsning til de f.eks. 10 nyeste cases. Kan udelades.

Når laboratoriesvarene er fundet, returneres svarene i GetLabReports-responsemeddelelsen, der ved fremsendelse af MedCom-meddelelser bør have denne form:

<soap:body>
<GetLabReportsResponse>
<Patient>
<CivilRegistrationNumber>0108624884</CivilRegistrationNumber>
<PersonGivenName>Jane</PersonGivenName>
<PersonSurnameName>Svendsen</PersonSurnameName>
</Patient>
<Result>
<Sent.Date.Time>20000114T12:47</Sent.Date.Time>
<Letter.Identifier>80901082504854</Letter.Identifier>
<Letter.TypeCode>XRPT01</Letter.TypeCode>
<Sender.OrganisationName>Hillerød Sygehus</Sender.OrganisationName>
<Sender.DepartmentName>Klinisk Kemisk Afdeling</Sender.DepartmentName>
<MedComMessage>
<Emessage xmlns="http://rep.oio.dk/medcom.dk/xml/schemas/2004/06/01/">**MedCom XRPT01 biokemisk laboratoriesvar**</Emessage>
</MedComMessage>
</Result>
<Result>
<!--Next MedCom laboratory report -->
</Result>
</GetLabReportsResponse>
</soap:body>

Hvor

- Hver responsemeddelelse kun omfatter én patient. Patientens CPR-nummer og navn nævnes først i body-elementet.
- Hvert laboratorieresultat starter med data, der kan bruges til at lave en ”liste” over alle medsendte laboratoriesvar, dvs. det originale afsendelsestidspunkt, meddelelses-id, brevtype og afsendende organisation.
- XML-elementet <MedComMessage> indeholder MedComs <Emessage>-tag, der er rodelement i alle MedCom XML-meddelelser.

Eksempel 2: Kreditkortbetaling

Hvis en udbudt webservice tilbydes mod betaling, kan Den Gode Webservice benyttes til at overføre kreditkortoplysninger til webserviceudbyderen. Udbyderen kan herefter sørge for, at beløbet kan overføres fra kundens bankkonto.

XML-koden indsættes i SOAP-body (evt. sammen med anden information) og har følgende syntaks:

```
<medcom:Payment>
<medcom:CardNumber>9876876543219876</medcom:CardNumber>
<medcom:CardValidThru>1005</medcom:CardValidThru>
<medcom:CardName>HANS H HANSEN</medcom:CardName>
<medcom:CardScurityCode>987</medcom:CardScurityCode>
</medcom:Payment>
```

Hvor de benyttede felter anvendes sådan:

CardNumber	string	Nummer på kreditkort eller kontonummer (inkl. registeringsnummer), der skal betale for benyttelse af webservicen
CardValidThru	string	Kortets udløbsmåned på formen MMYY
CardName	string	Navn på kortet (med store bogstaver)
CardSecurityCode	string	Trecifret sikkerhedskode på bagsiden af kortet (ikke pinkoden)

Eksempel 3: Henvisning og andre meddelelser

Et sygehus kan udbyde en webservice til at modtage sygehus- og røntgenhenvisninger. Henvisningerne fremsendes i Den Gode Webservice til en central webserviceserver på sygehuset.

Webservicen kan bestå af ét enkelt kald (”SendCase”) hvor:

- Requestmeddelelsens body-del indeholder den fremsendte henvisning og
  - patient-id og navn angives separat
  - meddelelsens dato, id, type og afsender angives for hver meddelelse
  - selve MedCom-meddelelsen er en helt almindelig MedCom XML-meddelelse.

- Responsemeddelelsen alene fungerer som positiv eller negativ kvittering, det vil sige
  - ved positiv kvittering angives dette i ”FlowStatus”, og body-delen er tom.
  - ved negativ kvittering angives dette i ”FlowStatus”, og body-delen indeholder en evt. fault-besked.

Se i øvrigt eksemplet med webservice-laboratoriesvar.

Eksempel 4: Onlinekommunikation med en central database

Det forventes, at sundhedssektoren fremover vil anvende centralt lagrede data i stor stil, f.eks. i medicinoplysninger, vandrejournaler og nationale patientindeks.

På medicinalområdet forventes det, at patientens opdaterede medicinkort på sigt ligger på en central server, hvor borgeren har adgang til medicinkortet via www.sundhed.dk. Lægesystemer, sygehussystemer og omsorgssystemer opdaterer deres lokale kopi af medicinkortet ved at bruge en webservice til at hente de opdaterede data fra den centrale server. Efter brug returneres de opdaterede data til serveren.

I hovedtræk vil kommunikationen mellem et klientsystem og den centrale server omfatte to funktioner: ”GetUpdate” og ”UpdateCase”:

- I ”GetUpdate” henter klientsystemet via Den Gode Webservice det centrale medicinkort (eller dele heraf) ned i eget it-system. Inden udlevering opdaterer webservice originalens løbenummer.

- ”UpdateCase”. Lægen indfører ordinationsændringerne i medicinkortet og returnerer det opdaterede XML-dokument til PEM-serveren. Opdateringen lykkes kun, hvis ingen andre brugere har ændret i de centrale data i mellemtiden. Hvis et nyt løbenummer viser, at dette er tilfældet, må klientsystemet starte forfra.

Hvis der er kommunikationsproblemer, skal det være muligt for klientsystemet at arbejde midlertidigt på den lokale kopi.

Eksempel 5: DGWS-body som ”billet”

En ”ticket”, eller en ”billet”, udstedes af et ”billet-kontor” og giver adgang til bestemte patientoplysninger.

For eksempel forventes det i Tyskland, at lægesystemerne lagrer en recept-billet på patientens fysiske sundheds-id-kort. Den indlejrede billet er et signeret og unikt id på den aktuelle recept.

Når patienten efterfølgende henvender sig på et apotek, kan apoteket indlæse recept-billetten og ved hjælp af denne hente recepten på en central server – et recept-hotel.

Bilag 4: Datalister

I datalisten gengives alle de værdibærende elementer i Den Gode Webservice-request/-response, i samme rækkefølge som variablene forekommer i XML-listen. XML Elementer, der ikke i sig selv er informationsbærende, er udeladt.

Skemaets første felt "Tag / Attribut" angiver navnet på det beskrevne tag eller den beskrevne attribut på pseudo Xpath-form (se http://www.w3.org/TR/xpath). Det betyder følgende:

- Tag-navne angives direkte med det namespace, de hører til (dog uden den fulde sti), f.eks. ds:Signature for Signature-elementet fra ds-namespacet (XML Signature).
- Attributter angives med et @ og med navnet på det tag, de tilhører, sat foran, f.eks. saml:Assertion@Version for Version-attributten på tagget saml:Assertion.
- Af hensyn til læsbarheden af saml Attribute-elementer angives en yderligere kompaktnotation, f.eks. [@Name='sosi:IDCardType'] for værdien af det AttributeValue-element, der er indlejret i det Attribute-element, hvis Name-attribut har værdien IDCardType. I eksemplet nedenfor ville dette svare til værdien "system":

  ```
  <saml:Attribute Name="sosi:IDCardType">
    <saml:AttributeValue>system</saml:AttributeValue>
  </saml:Attribute>
  ```

  Den fulde relative XPath-syntaks vil i dette tilfælde være
  //saml:Attribute[@Name="sosi:IDCardType"]/saml:AttributeValue/text()

Skemaets "type" felt angiver en XML Schema-type eller en enumeration. Følgende typer anvendes:

- "string" angiver, at dataindholdet skal være en streng. Reserverede XML-styrekartekterer må ikke forekomme. Se http://www.w3.org/TR/xmlschema11-2/#string
- "integer" angiver, at dataindholdet er et positivt hel-tal. Se http://www.w3.org/TR/xmlschema11-2/#integer
- "dateTime" angiver, at data er en dato og et klokkeslæt efter lokal dansk tid og formatet YYYY-MM-DDTHH:MM:SS, f.eks. 2006-05-28T23:59:00 for 28. maj 2006 kl. 23:59:00. Den Gode Webservice kræver, at webserviceklienter og webserviceudbydere synkroniserer urene efter en global anerkendt tidsserver og benytter lokal dansk tid som tidsangivelse, der også følger sommer og vintertid. Se http://www.w3.org/TR/xmlschema11-2/#dateTime
- "anyType" angiver, at elementet kan indeholde et vilkårligt indlejret XML-dokument.
- "ID" bruges til at navngive et XML-element i dokumentet og er af typen http://www.w3.org/TR/xmlschema11-2/#ID, hvilket betyder, at det er en string, hvor kolon ":" ikke er tilladt.
- "base64binary" angiver en base64-kodet binær streng af data, f.eks. et digitalt certifikat eller en signatur. Se http://www.w3.org/TR/xmlschema11-

2/#base64Binary. Base64-algortimen konverterer binære data til et begrænset sæt af 64 ASCII-tegn, nemlig A-Z, a-z, 0-9 samt symbolerne "+", "/" og "=".
- "ENUM" angiver, at der skal benyttes én af de valgmuligheder, der fremgår af enumerationslisten.

Kolonnen "betingelse" anvendes til at beskrive, i hvilke tilfælde et element skal medtages, og i hvilke det skal undlades. Hvis der *ikke* er en betingelse på elementet, er det altid lovligt at medtage. Hvis der *er* en betingelse på elementet, skal det kun medtages, hvis betingelsen er opfyldt. F.eks. skal elementet ds:Signature kun medtages, hvis sikkerhedsniveau 3 eller 4 anvendes, dvs. hvor der er digital signatur på id-kortet (angivet som "Niveau 3/4").

Nogle elementer kan forekomme flere gange, nogle er valgfri, og andre skal altid medtages. Dette angives med kolonnen "Antal", hvor følgende gælder:

- 1 betyder, at elementet *altid* skal forekomme, hvis betingelsen er opfyldt.
- 0..1 betyder, at elementet kan forekomme 0 eller 1 gang, hvis betingelsen er opfyldt.
- 0..n betyder, at elementet kan forekomme 0 eller et vilkårligt antal gange, hvis betingelsen er opfyldt.

Endelig angives en beskrivelse af elementet i den sidste kolonne.

Request-dataliste

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
soap:Envelope@xmlns:saml	string		1	Det officielle namespace for SAML-sikkerhedsstandarden. SAML benyttes som den standardmekanisme, der transporterer oplysninger om DGWS-brugere. Saml 2.0-dokumentation findes på http://docs.oasis-open.org/security/saml/v2.0/saml-2.0-os.zip
soap:Envelope@xmlns:wsse	string		1	Det officielle namespace for Webservice Security-standarden. Angiver bestemte XML-tags for brug af signering, token og kryptering. Dokumentationen findes på "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0" og på "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
soap:Envelope@xmlns:ds	string		1	Det officielle namespace for XML Signature-standarden. Dokumentation findes på http://www.w3.org/TR/xmldsig-core/
soap:Envelope@xmlns:medcom	string		1	Det lokale namespace for de XML-elementer, der indgår i DGWS. Dokumentationen findes på "http://www.medcom.dk"
soap:Envelope@xmlns:sosi	string		1	Det lokale namespace for de XML-elementer, der indgår i DGWS. Dokumentationen findes på "http://www.sosi.dk"

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
soap:Envelope@xmlns:soap	string		1	Det officielle namespace for soap version 1.1. Dokumentationen findes på http://schemas.xmlsoap.org/soap/envelope/"
soap:Envelope@xmlns:wsu	string		1	Det officielle namespace for en tilføjelse (utility) til soapstandarden, der bl.a.definerer "Id" på XML-tag-niveau. Dokumentaionen findes på adressen "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
soap:Header			1	soap header
wsse:Security			1	wsse:Securitydata gentages nøjagtigt i efterfølgende requests i samme WS-session, da serveren gennemgår autorisationsprocedurerne ved hver ny request. Dog opdateres afsendelsestidspunktet "Created" i efterfølgende requests
wsu:Created	dateTime		1	Dato og klokkeslæt for påbegyndelse af generering af soap-meddelelsen. Skal ifølge wsse-standarden genereres så tæt på afsendelsestidspunktet som muligt
saml:Assertion@ID	ID		1	Den del af DGWS, der indeholder oplysninger om afsenderen og bevis for dennes identitet (bruger, system, signatur, username/password etc.)
saml:Assertion@IssueInstant	dateTime		1	Det tidspunkt, hvor id-kortet blev skabt. Hvis id-kortet indeholder en digital signatur, angiver IssueInstant det tidspunkt, hvor beskeden blev underskrevet (dvs. lige inden)
saml:Assertion@Version	string		1	SAML versions-id. DGWS benytter p.t. 2.0
saml:Issuer	string		1	Navn på den organisation, der har udstedt id-kortet (eller underskrevet det)
saml:ConfirmationMethod	ENUM		1	Angiver, hvordan oplysningerne kan godtgøres, f.eks. ved at indehaveren fremviser en nøgle (brugernavn/password, signatur). DGWS bruger kun "urn:oasis:names:tc:SAML:2.0:cm:holder-of-key"
ds:KeyName	string	Niveau 3/4	1	På niveau 3/4: Reference til det id, som id-kortets digitale signatur har
		Niveau 2	1	Brugernavn og password. Benyttes kun på niveau 2. Må ikke anvendes på de andre niveauer!
wsse:Username	string	Niveau 2	1	Brugerens adgangsnavn ved login på webserviceudbydersystemet
wsse:Password	string	Niveau 2	1	Brugerens password ved login på webserviceudbydersystemet
saml:Conditions@NotBefore	dateTime		1	Tidspunkt for id-kortets oprettelse. Id-kortet er ugyldigt før dette tidspunkt. Dette tidspunkt benyttes af webservicen ved beregning af timeout
saml:Conditions@NotOnOrAfter	dateTime		1	Tidspunkt for id-kortets udløb. Sættes til NotBefore + 24 timer. Efter dette tidspunkt er id-kortet ugyldigt

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
[@id='IDCardData']			1	Oplysninger om selve id-kortet
[@Name='sosi:IDCardID']	string		1	Et unikt id for dette id-kort. To id-kort må aldrig anvende samme id
[@Name='sosi:IDCardVersion']	string		1	Angiver den version af id-kort-formatet, som dette id-kort er lavet ud fra. P.t. findes kun 1.0
[@Name='sosi:IDCardType']	ENUM		1	Angiver om id-kortet identificerer en medarbejder ("user") eller et it-system ("system")
[@Name='sosi:AuthenticationLevel']	ENUM		1	Det sikkerhedsniveau, som dette id-kort blev udstedt til. Lovlige værdier er 1 = ingen autentifikation, 2 = brugernavn/password, 3 = VOCES-signatur, 4 = MOCES-signatur. DGWS tillader også niveau 5 med digital signatur på hele kuverten, men dette angives IKKE i id-kortet – kun i medcom:SecurityLevel feltet.
[@Name='sosi:OCESCertHash']	base64binary	Niveau 3/4	1	SHA-1 hashværdi af det certifikat, der blev anvendt til autentifikation
[@id='UserLog']			1	Oplysninger om brugeren til brug ved tildeling af rettigheder og ved generering af log-fil hos webserviceserveren. Rettigheder kan være tildelt ud fra CPR-nummer, rolle, f.x. "læge", it-system eller brugers organisation
[@Name='medcom:UserCivilRegistrationNumber']	string		1	Brugers CPR-nummer eller et erstatnings-CPR-nummer
[@Name='medcom:UserGivenName']	string		0..1	Brugerens for- og mellemnavne, f.eks. Hans H.
[@Name='medcom:UserSurName']	string		0..1	Brugerens efternavn f.eks. Hansen
[@Name='medcom:UserEmailAddress']	string		0..1	Brugerens e-mail-adresse
[@Name='medcom:UserRole']	string		1	Brugers rolle (Fx "læge" eller "Ansat på OUH") i forbindelse med rettighedstildeling. En bruger kan have flere roller
[@Name='medcom:UserOccupation']	string		0..1	Brugers stilling, f.eks. overlæge
[@Name='medcom:UserAuthorizationCode']	string		0..1	Brugerens autorisationskode fra Sundhedsstyrelsen Autorisationsregister
[@Name='medcom:CareProviderID']@NameFormat	ENUM		1	Den type id, som identificerer brugerens organisation. Se enumerationslisten for valide værdier.
[@id='SystemLog']			1	Oplysninger om det it-system, som denne DGWS-besked blev sendt fra
[@Name='medcom:ITSystemName']	string		1	Navnet på det it-system, som denne DGWS-besked kom fra
[@Name='medcom:CareProviderID']	string		1	Unikt id for den organisation, brugeren optræder for. Formatet af id'et angives med NameFormat
[@Name='medcom:CareProviderName']	string		0..1	Navn på brugerens organisation
ds:Signature[@id=OCESSignature]		Niveau 3/4	1	Digital signatur af id-kortet. Anvendes til autentifikation på niveau 3/4 og til at beskytte integriteten af id-kortet (det kan ikke ændres, uden at signaturen bliver ugyldig)

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
ds:CanonicalizationMethod@Algorithm	ENUM	Niveau 3/4	1	Angiver den metode, der er benyttet til at "normalisere" det XML, der skal signeres (SignedInfo-elementet). Kanoniseringen konverterer XML-dokumetet til et standardtegnsæt og -format, der sikrer, at både afsender og modtager ser dokumentet på nøjagtig samme måde, inden dokumentet signeres. WSSE-standarden anbefaler "Exclusive XML Canonisering", der er dokumenteret på W3C, se http://www.w3.org/TR/2001/REC-xml-c14n-20010315
ds:SignatureMethod@Algorithm	ENUM	Niveau 3/4	1	Angiver den metode, der benyttes ved signering af SignedInfo-elementet. Først kanoniseres SignedInfo, derpå beregnes et digest (en "tværsum) vha. SHA-1- metoden (Secure Hash se Se http://www.w3.org/2000/09/xmldsig#sha). Til sidst bruges den private nøgle til at kryptere digesten, som gemmes i SignatureValue-feltet
ds:Reference@URI	String	Niveau 3/4	1	Reference peger på det XML-element, der skal signeres. " IDCard" id-attributten på hele id-kortet (Assertion-elementet). I Den Gode Webservice signeres normalt kun id-kortet, men det er også muligt at underskrive hele konvolutten (sikkerhedsniveau 5). # angiver at det, der refereres til, findes indlejret i samme XML-dokument
ds:DigestMethod@Algorithm	ENUM	Niveau 3/4	1	Den metode "SHA-1", der benyttes til at generere "digesten" (tværsummen - chekcifret-hashværdien- fingeraftrykket) for den XML, der er signeret (id-kortet i dette tilfælde). Se http://www.w3.org/2000/09/xmldsig#sha1
ds:DigestValue	base64binary	Niveau 3/4	1	Digesten eller hash-værdien, der er den værdi, der er resultatet af at beregne en SHA-1-digest på objektet. Feltet skal base64binary-kodes
ds:SignatureValue	base64binary	Niveau 3/4	1	Den digitale signatur. Signaturen er de bytes, der er fremkommet ved at signere SignedInfo-elementet ved hjælp af den private nøgle. Modtageren dekrypterer denne signatur med den offentlige nøgle fra certifikatet, beregner selv værdien på samme måde som afsenderen og sammenligner de to værdier. Er de ens, er det et bevis for, at dokumentet virkelig er underskrevet af den, der kan identificeres ved certifikatet, og i øvrigt er uændret undervejs
ds:X509Certificate	base64binary	Niveau 3/4	1	KeyInfo (nøgle-info). Indholder det certifikat, hvis indlejrede offentlige nøgle kan dekryptere SignatureValue
ds:Subject			1	Identifikation af det subjekt (medarbejder eller system) der identificeres ved id-kortets SAML-assertion.
ds:NameID@Format	ENUM		1	Angiver formatet på id'et. Se enumerationslisten for valide værdier.

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
ds:NameID	String		1	Id på det subjekt, som id-kortet identificerer. Af typen angivet ved Format-attributten.
ds:Signature[@id=OCESSignature2]		Niveau 5	1	Digital signatur af hele kuverten. Anvendes til autentifikation på niveau 5 og til at beskytte integriteten af hele beskeden (intet kan ændres, uden at signaturen bliver ugyldig)
ds:CanonicalizationMethod@Algorithm	string	Niveau 5	1	Angiver den metode, der er benyttet til at "normalisere" det XML, der skal signeres (SignedInfo-elementet). Kanoniseringen konverterer XML-dokumetet til et standardtegnsæt og -format, der sikrer, at både afsender og modtager ser dokumentet på nøjagtig samme måde, inden dokumentet signeres. WSSE-standarden anbefaler "Exclusive XML Canonisering", der er dokumenteret på W3C, se http://www.w3.org/TR/2001/REC-xml-c14n-20010315
ds:SignatureMethod@Algorithm	string	Niveau 5	1	Angiver den metode, der benyttes ved signering af SignedInfo-elementet. Først kanoniseres SignedInfo, derpå beregnes et digest (en "tværsum) vha. SHA-1- metoden (Secure Hash se http://www.w3.org/2000/09/xmldsig#sha). Til sidst bruges den private nøgle til at kryptere digesten, som gemmes i SignatureValue feltet
ds:Reference@URI	string	Niveau 5	1	Reference peger på det XML-element, der skal signeres. "Envelope" er id-attributten på hele kuverten (Envelope-elementet). # angiver, at det, der refereres til, findes indlejret i samme XML-dokument
ds:DigestMethod@Algorithm	string	Niveau 5	1	Den metode "SHA-1", der benyttes til at generere "digesten" (tværsummen - chekcifret-hashværdien- fingeraftrykket) for den XML, der er signeret (Envelope-elementet i dette tilfælde). Se http://www.w3.org/2000/09/xmldsig#sha1
ds:DigestValue	base64binary	Niveau 5	1	Digesten eller hash-værdien, der er en resulterende værdi af at beregne en SHA-1 digest på objektet. Feltet skal base64binary-kodes
ds:SignatureValue	base64binary	Niveau 5	1	Den digitale signatur. Signaturen er de bytes, der er fremkommet ved at signere SignedInfo-elementet ved hjælp af den private nøgle. Modtageren dekrypterer denne signatur med den offentlige nøgle fra certifikatet, beregner selv værdien på samme måde som afsenderen og sammenligner de to værdier. Er de ens, er det er bevis for, at dokumentet virkelig er underskrevet af den, der kan identificeres ved certifikatet, og i øvrigt er uændret undervejs
ds:X509Certificate	base64binary	Niveau 5	1	KeyInfo (nøgle-info). Indholder det certifikat, hvis indlejrede offentlige nøgle kan dekryptere SignatureValue
medcom:SecurityLevel	ENUM		1	Angiver det sikkerhedsniveau, som hele MedCom-kuverten er underlagt. Valide værdier er 1-5

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
Medcom:TimeOut	ENUM		0	
medcom:FlowID	string		1	Unikt id for en session (et antal beskeder indgår i samme workflow). Alle request- og response-beskeder i samme session har samme FlowID
medcom:MessageID	string		1	Unikt id for denne DGWS-kuvert. Må ikke genbruges af andre DGWS-kuverter. Bruges til at identificere dubletter ved gensendelse
medcom:Priority	ENUM		1	Klientens angivelse af WS-prioritet. Angives med én af følgende tre værdier: "AKUT", "HASTER" eller "RUTINE". Bruges som hint til webserviceudbyder og klientsystem om, hvordan beskeden skal prioriteres (om muligt)
medcom:RequireNonRepudiationReceipt	ENUM		0..1	Angiver, hvorvidt klientsystemet ønsker en digital signatur på response-beskeden. Hvis værdien er "yes", skal webserviceudbyderen underskrive hele responsekuverten med sin digitale signatur om muligt. Hvis webserviceudbyderen ikke kan håndtere digital signatur, sendes fejlkoden ”nonrepudiation_not_supported”
soap:Body	anyType		1	I DGWS body-elementet indlejres webservice-specifikke request-beskeder og input-parametre.

Response-dataliste

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
soap:Envelope@xmlns:saml	string		1	Det officielle namespace for SAML-sikkerhedsstandarden. SAML benyttes som den standardmekanisme, der transporterer oplysninger om DGWS-brugere. Saml 2.0-dokumentation findes på http://docs.oasis-open.org/security/saml/v2.0/saml-2.0-os.zip
soap:Envelope@xmlns:wsse	string		1	Det officielle namespace for Webservice Security-standarden. Angiver bestemte XML-tags for brug af signering, token og kryptering. Dokumentaionen findes på "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0" og på "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
soap:Envelope@xmlns:ds	string		1	Det officielle namespace for XML Signature-standarden. Dokumentation findes på http://www.w3.org/TR/xmldsig-core/
soap:Envelope@xmlns:medcom	string		1	Det lokale namespace for de XML-elementer, der indgår i Den Gode Webservice. Dokumentationen findes på "http://www.medcom.dk"
soap:Envelope@xmlns:sosi	string		1	Det lokale namespace for de XML-elementer, der indgår i Den Gode Webservice. Dokumentationen findes på "http://www.sosi.dk"
soap:Envelope@xmlns:soap	string		1	Det officielle namespace for soap version 1.1. Dokumentationen findes på http://schemas.xmlsoap.org/soap/envelope/"
soap:Envelope@xmlns:wsu	string		1	Det officielle namespace for en tilføjelse (utility) til soap-standarden, der bl.a.definerer "Id" på XML-tag-niveau. Dokumentaionen findes på adressen "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
wsse:Security			1	wsse:Security-data gentages nøjagtigt i efterfølgende requests i samme WS-session, da serveren gennemgår autorisations-procedurerne ved hver ny request. Dog opdateres afsendelsestidspunktet "Created" i efterfølgende requests
wsu:Created	dateTime		1	Dato og klokkeslæt for påbegyndelse af generering af soap-meddelelsen
saml:Assertion@ID	ID		1	Den del af DGWS, der indeholder oplysninger om afsenderen og bevis for dennes identitet (bruger, system, signatur, username/password etc.)
saml:Assertion@IssueInstant	dateTime		1	Det tidspunkt, hvor id-kortet blev skabt. Hvis id-kortet indeholder en digital signatur, angiver IssueInstant det tidspunkt, hvor beskeden blev underskrevet (dvs. lige inden)
saml:Assertion@Version	string		1	SAML-versionsi-d. DGWS benytter p.t. "2.0"

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
saml:Issuer	string		1	Navn på den organisation, der har udstedt id-kortet (eller underskrevet det)
saml:ConfirmationMethod	string		1	Angiver, hvordan oplysningerne kan godtgøres, f.eks. ved at indehaveren fremviser en nøgle (brugernavn/password, signatur). DGWS bruger kun "urn:oasis:names:tc:SAML:2.0:cm:holder-of-key"
ds:KeyName	string	Niveau 3/4	1	På niveau 3/4: Reference til det id, som id-kortets digitale signatur har
saml:Conditions@NotBefore	dateTime		1	Tidspunkt for id-kortets oprettelse. Id-kortet er ugyldigt før dette tidspunkt
saml:Conditions@NotOnOrAfter	dateTime		1	Tidspunkt for id-kortets udløb. Sættes til NotBefore + 24 timer. Efter dette tidspunkt er id-kortet ugyldigt
[@id='IDCardData']			1	Oplysninger om selve id-kortet
[@Name='sosi:IDCardID']	string		1	Et unikt id for dette id-kort. To id-kort må aldrig anvende samme id
[@Name='sosi:IDCardVersion']	string		1	Angiver den version af id-kort-formatet, som dette id-kort er lavet ud fra. P.t. findes kun 1.0
[@Name='sosi:IDCardType']	ENUM		1	Angiver om id-kortet identificerer en medarbejder ("user") eller et it-system ("system")
[@Name='sosi:AuthenticationLevel']	ENUM		1	Det sikkerhedsniveau, som dette id-kort blev udstedt til. Lovlige værdier er 1 = ingen autentifikation, 2 = brugernavn/password, 3 = VOCES signatur, 4 = MOCES-signatur. DGWS tillader også niveau 5 med digital signatur på hele kuverten, men dette angives IKKE i id-kortet – kun i medcom:SecurityLevel-feltet
[@Name='sosi:OCESCertHash']	base64binary	Niveau 3/4	1	SHA-1-hashværdi af det certifikat, der blev anvendt til autentifikation
[@id='medcom:SystemLog']			1	Oplysninger om det it-system, som denne DGWS-besked blev sendt fra
[@Name='medcom:ITSystemName']	string		1	Navnet på det it-system, som denne DGWS-besked kom fra
[@Name='medcom:CareProviderID']	string		1	Unikt id for den organisation, brugeren optræder for. Formatet af id'et angives med NameFormat
[@Name='medcom:CareProviderName']	string		0..1	Navn på brugerens organisation
ds:Signature[@id=OCESSignature]		Niveau 3/4	1	Digital signatur af id-kortet. Anvendes til autentifikation på niveau 3/4 og til at beskytte integriteten af id-kortet (det kan ikke ændres, uden at signaturen bliver ugyldig)
ds:CanonicalizationMethod@Algorithm	ENUM	Niveau 3/4	1	Angiver den metode, der er benyttet til at "normalisere" det XML, der skal signeres (SignedInfo-elementet). Kanoniseringen konverterer XML-dokumetet til et standardtegnsæt og -format, der sikrer, at både afsender og modtager ser dokumentet på nøjagtig samme måde, inden dokumentet signeres. WSSE-standarden anbefaler "Exclusive XML Canonisering", der er

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
				dokumenteret på W3C, se http://www.w3.org/TR/2001/REC-xml-c14n-20010315.
ds:SignatureMethod@Algorithm	ENUM	Niveau 3/4	1	Angiver den metode, der benyttes ved signering af SignedInfo-elementet. Først kanoniseres SignedInfo, derpå beregnes et digest (en "tværsum) vha. SHA-1- metoden (Secure Hash se Se http://www.w3.org/2000/09/xmldsig#sha). Sluttelig bruges den private nøgle til at kryptere digesten, som gemmes i SignatureValue-feltet
ds:Reference@URI	string	Niveau 3/4	1	Reference peger på det XML-element, der skal signeres. " IDCard" id-attributten på hele id-kortet (Assertion-elementet). I Den Gode Webservice signeres normalt kun id-kortet, men det er også muligt at underskrive hele konvolutten (sikkerhedsniveau 5). # angiver, at det, der refereres til, findes indlejret i samme XML-dokument
ds:DigestMethod@Algorithm	ENUM	Niveau 3/4	1	Den metode "SHA-1", der benyttes til at generere "digesten" (tværsummen - chekcifret-hashværdien- fingeraftrykket) for den XML, der er signeret (id-kortet i dette tilfælde). Se http://www.w3.org/2000/09/xmldsig#sha1
ds:DigestValue	base64binary	Niveau 3/4	1	Digesten eller hash-værdien, der er en resulterende værdi af at beregne en SHA-1-digest på objektet. Feltet skal base64binary-kodes
ds:SignatureValue	base64binary	Niveau 3/4	1	Den digitale signatur. Signaturen er de bytes, der er fremkommet ved at signere SignedInfo-elementet ved hjælp af den private nøgle. Modtageren dekrypterer denne signatur med den offentlige nøgle fra certifikatet, beregner selv værdien på samme måde som afsenderen og sammenligner de to værdier. Er de ens, er det et bevis for, at dokumentet virkelig er underskrevet af den, der kan identificeres ved certifikatet, og i øvrigt er uændret undervejs
ds:X509Certificate	base64binary	Niveau 3/4	1	KeyInfo (nøgle-info). Indholder det certifikat hvis indlejrede offentlige nøgle kan dekryptere SignatureValue
ds:Subject			1	Identifikation af det subjekt (medarbejder eller system), der identificeres ved id-kortets SAML-assertion

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
ds:NameID@Format	ENUM		1	Angiver formatet på id'et. Se enumerationslisten for valide værdier.
ds:NameID	string		1	Id på det subjekt, som id-kortet identificerer. Af typen angivet ved Format-attributten.
ds:Signature[@id=OCESSignature2]		Niveau 5	1	Digital signatur af hele kuverten. Anvendes til autentifikation på niveau 5 og til at beskytte integriteten af hele beskeden (intet kan ændres, uden at signaturen bliver ugyldig)
ds:CanonicalizationMethod@Algorithm	ENUM	Niveau 5	1	Angiver den metode, der er benyttet til at "normalisere" det XML, der skal signeres (SignedInfo-elementet). Kanoniseringen konverterer XML-dokumetet til et standardtegnsæt og -format, der sikrer, at både afsender og modtager ser dokumentet på nøjagtig samme måde, inden dokumentet signeres. WSSE-standarden anbefaler "Exclusive XML Canonisering", der er dokumenteret på W3C, se http://www.w3.org/TR/2001/REC-xml-c14n-20010315
ds:SignatureMethod@Algorithm	ENUM	Niveau 5	1	Angiver den metode, der benyttes ved signering af SignedInfo-elementet. Først kanoniseres SignedInfo, derpå beregnes et digest (en "tværsum) vha. SHA-1 metoden (Secure Hash se Se http://www.w3.org/2000/09/xmldsig#sha). Sluttelig bruges den private nøgle til at kryptere digesten, som gemmes i SignatureValue-feltet
ds:Reference@URI	string	Niveau 5	1	Reference peger på det XML-element, der skal signeres. "Envelope" er id-attributten på hele kuverten (Envelope-elementet). # angiver, at det, der refereres til, findes indlejret i samme XML-dokument
ds:DigestMethod@Algorithm	ENUM	Niveau 5	1	Den metode "SHA-1", der benyttes til at generere "digesten" (tværsummen - chekcifret-hashværdien- fingeraftrykket) for den XML, der er signeret (Envelope-elementet i dette tilfælde). Se http://www.w3.org/2000/09/xmldsig#sha1
ds:DigestValue	base64binary	Niveau 5	1	Digesten eller hash-værdien, der er en resulterende værdi af at beregne en SHA-1-digest på objektet. Feltet skal base64binary-kodes

Tag / Attribut	Type	Beting-else	Antal	Beskrivelse
ds:SignatureValue	base64binary	Niveau 5	1	Den digitale signatur. Signaturen er de bytes, der er fremkommet ved at signere SignedInfo-elementet ved brug af den private nøgle. Modtageren dekrypterer denne signatur med den offentlige nøgle fra certifikatet, beregner selv værdien på samme måde som afsenderen og sammenligner de to værdier. Er de ens, er det et bevis for, at dokumentet virkelig er underskrevet af den, der kan identificeres ved certifikatet, og i øvrigt er uændret undervejs
ds:X509Certificate	base64binary	Niveau 5	1	KeyInfo (nøgle-info). Indholder det certifikat, hvis indlejrede offentlige nøgle kan dekryptere SignatureValue
medcom:SecurityLevel	ENUM		1	Angiver det sikkerhedsniveau, som hele MedCom-kuverten er underlagt. Valide værdier er 1-5
medcom:FlowID	string		1	Unikt id for en session (et antal beskeder indgår i samme workflow). Alle request- og responsebeskeder i samme session har samme FlowID
medcom:MessageID	string		1	Unikt id for denne DGWS-kuvert. Må ikke genbruges af andre DGWS-kuverter. Bruges til at identificere dubletter ved gensendelse
medcom:InResponseToMessageID	string		1	MessageID for den tilhørende requestbesked kopieres til dette felt
medcom:FlowStatus	ENUM		1	Status, der angiver, hvordan kaldet forløb. Se listen over fejlkoder
soap:Body	any		1	I DGWS body-elementet indlejres webservicespecifikke requestbeskeder og inputparametre
faultcode	string		0..1	Angiver at der er sket en fejl på serveren og har altid værdien ”Server”
detail	string		0..1	Angiver en specifik fejlkode fra enumerationslisten eller en brugspecifik kode.
faultstring	string		0..1	En tekst, der kan læses af alle, der beskriver, hvorfor fejlen opstod

Bilag 5: Enumerationsliste

Request metanavn	Koder	Kodebetydning
[@Name='sosi:IDCardType']	user	Id-kortet repræsenterer en medarbejder
[@Name='sosi:IDCardType']	system	Id-kortet repræsenterer et system
[@Name='sosi:AuthenticationLevel']	1	Id-kortet indeholder ingen akkreditiver
[@Name='sosi:AuthenticationLevel']	2	Id-kortet indeholder brugernavn/password
[@Name='sosi:AuthenticationLevel']	3	Id-kortet indeholder en VOCES-signatur
[@Name='sosi:AuthenticationLevel']	4	Id-kortet indeholder en MOCES-signatur
medcom:Priority	AKUT	Behandling af forespørgslen skal ske så hurtigt som muligt
medcom:Priority	HASTER	Behandling af forespørgslen haster
medcom:Priority	RUTINE	Behandling af forespørgslen skal ske efter de almindelige procedurer
medcom:RequireNonRepudiationReceipt	yes	Klientsystemet forlanger, at webserviceudbyderen underskriver responsemeddelelsen med en VOCES-signatur
medcom:RequireNonRepudiationReceipt	no	Der skal ikke være nogen signatur på svaret. Alternativt kan feltet helt udelades fra XML
ds:CanonicalizationMethod@Algorithm	http://www.w3.org/TR/2001/REC-xml-c14n-20010315	Anvend kanoninseringsalgoritmen "C14N omit comments". Tillades i DGWS, men er ikke så robust som exclusive C14N
ds:CanonicalizationMethod@Algorithm	http://www.w3.org/2001/10/xml-exc-c14n#	Anvend kanoninseringsalgoritmen "Exclusive C14N omit comments". Anbefales
ds:SignatureMethod@Algorithm	http://www.w3.org/2000/09/xmldsig#rsa-sha1	Anvend RSA-SHA1 som signeringsalgoritme.
ds:DigestMethod@Algorithm	http://www.w3.org/2000/09/xmldsig#sha1	Anvend SHA1 (secure hash) som metoden, der beregner hashværdier.
ds:NameID@Format	http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/CPR_PersonCivilRegistrationIdentifier.xsd	Feltet indeholder et CPR-nummer på OIO-format - det vil sige uden bindestreg
ds:NameID@Format	urn:medcom:names:careprovider:ynumber	Feltet indeholder et sygesikringsydernummer
ds:NameID@Format	urn:medcom:names:careprovider:pnumber	Feltet indeholder et kommunalt P-nummer
ds:NameID@Format	urn:medcom:names:careprovider:skscode	Feltet indeholder en SKS-sygehusafdelingskode
ds:NameID@Format	urn:medcom:names:careprovider:cvrnumber	Feltet indeholder et CVR-nummer
saml:ConfirmationMethod	urn:oasis:names:tc:SAML:2.0:cm:holder-of-key	Feltet indeholder et OCES PID (person identifier number)-nummer. Nummeret kan af TDC udskiftes med det tilsvarende CPR-nummer
[@Name='medcom:CareProviderID']/@NameFormat	urn:medcom:names:careprovider:ynumber	Feltet indeholder et unikt id-nummer af anden type end de ovennævnte
[@Name='medcom:CareProviderID']/@NameFormat	urn:medcom:names:careprovider:pnumber	Feltet indeholder et P-nummer, der er produktionsenhedsnummer fra CVR-registeret, der tildeles et CVR-nummer for hver fysisk

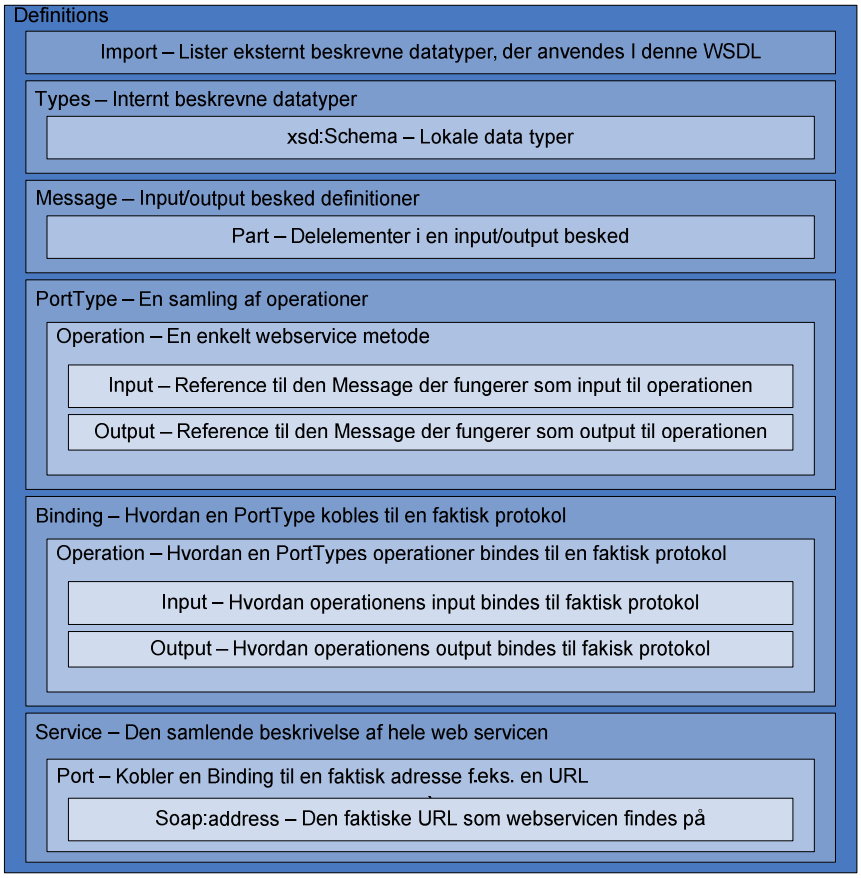
Request metanavn	Koder	Kodebetydning
		enhed
[@Name='medcom:CareProvider ID']/@NameFormat	urn:medcom:names:careprovider:skscode	Feltet indeholder en SKS-sygehusafdelingskode
[@Name='medcom:CareProvider ID']/@NameFormat	urn:medcom:names:careprovider:cvrnumber	Feltet indeholder et CVR-nummer
medcom:FlowStatus	flow_runnig	Webserviceudbyderen har modtaget og behandler den modtagne request. Klientsystemet skal kalde webserveren senere for at hente en efterfølgende request
medcom:FlowStatus	flow_finalized_succesfully	Webservicesession er afsluttet med succes. Klientsystemet behøver ikke nødvendigvis at kalde webserveren igen. Hvis klientsystemet benytter sidste kald i en session flere gange, vil der blive returneret flere ”færdigbehandlet”-kvitteringer
medcom:FaultCode	syntax_error	Den sendte besked indeholdt data, der ikke blev forstået af serveren
medcom:FaultCode	missing_required_header	Der mangler en eller flere obligatoriske DGWS-headere i den medsendte besked, f.eks. id-kort, som altid skal være der
medcom:FaultCode	security_level_failed	Autentifikation eller autorisationsfejl. Forkert valgt sikkerhedsniveau
medcom:FaultCode	invalid_username_password	Autentifikation eller autorisationsfejl. Fejl i username/password
medcom:FaultCode	invalid_signature	Autentifikation eller autorisationsfejl. Fejl i digital OCES-signatur enten på id-kortet eller på hele kuverten
medcom:FaultCode	invalid_idcard	Autentifikation eller autorisationsfejl. Fejl i SOSI id-kort, f.eks. at CPR-nummeret ikke matcher det, der kan slås op via certifikatet
medcom:FaultCode	invalid_certificate	Autentifikation eller autorisationsfejl. Certifikat er ikke OCES, det er spærret eller udløbet.
medcom:FaultCode	expired_idcard	Autentifikation eller autorisationsfejl. SOSI ID udløbet eller for gammelt for denne webserviceudbyder
medcom:FaultCode	not_authorized	Brugeren har ikke rettigheder til at udføre denne webservice
medcom:FaultCode	illegal_http_method	Bruges, hvis en klient sender alle andre HTTP Methods end ”GET” og ”POST”. Bruges også, hvis en server ikke udstiller webservice-implementation via ”GET”
medcom:FaultCode	nonrepudiation_not_supported	Returneres, hvis klienten har sat RequireNonRepudiationReceipt til ”yes”, men webserviceudbyderen ikke understøtter digital signering
HTTP Status	200 OK	Meddelelsen er modtaget
HTTP Status	500_Internal_Server_Error	Meddelelsen kunne ikke modtages
medcom:SecurityLevel	1	DGWS-kuverten indeholder et id-kort uden akkreditiver
medcom:SecurityLevel	2	DGWS-kuverten indeholder et id-kort med brugernavn/password
medcom:SecurityLevel	3	DGWS-kuverten indeholder et id-kort med en VOCES-signatur
medcom:SecurityLevel	4	DGWS-kuverten indeholder et id-kort med en

Request metanavn	Koder	Kodebetydning
		MOCES-signatur
medcom:SecurityLevel	5	Hele DGWS-kuverten er digitalt signeret
medcom:TimeOut	5	Webserviceudbyder kræver fornyet brugerautentifikation ved hvert nyt http-kald. Id-kortet udløber efter 5 minutter
medcom:TimeOut	30	Webserviceudbyder kræver fornyet brugerautentifikation, når id-kortet er 30 minutter gammelt
medcom:TimeOut	480	Webserviceudbyder kræver fornyet brugerautentifikation, når id-kortet er 8 timer (480 minutter) gammelt
medcom:TimeOut	1440	Webserviceudbyder kræver fornyet brugerautentifikation, når id-kortet er 24 timer (1440 minutter) gammelt
medcom:TimeOut	undbounded	Webserviceudbyder kræver ikke fornyet brugerautentifikation.

BILAG 7: WSDL For Den Gode Webservice

WSDL er en forkortelse for Web Services Description Language. Et WSDL-dokument er et XML-dokument, som beskriver en eller flere SOAP-beskeder, og hvordan disse beskeder udveksles. WSDL bruger et XML-skema til at beskrive de XML-beskeder og datatype-definitioner, der indgår i kommunikationen.

Ud over at beskrive de beskeder webservicen bruger, angiver WSDL'en også, hvor servicen kan tilgås, og hvilken protokol der skal bruges for at kommunikere med servicen, f.eks. HTTP. Figuren nedenfor viser strukturen af en WSDL for DGWS-webservices:



Figur 1: Strukturen af WSDL for DGWS-webservices

WSDL-dokumentet indeholder følgende centrale elementer:

- **Import**
  Hvordan eksterne datatyper defineres via XML-skema importeres ind i denne WSDL-fil med henblik på genbrug. Disse datatyper benyttes efterfølgende til at definere de beskeder, som webservice anvender.
- **Types**
  Som et alternativ til at importere eksterne datatyper kan man vælge at definere dem direkte i WSDL-filen. Dette element indeholder datatype definitioner, som har

relevans for de beskeder, der bruges til kommunikation med webservicen. Indholdet i Types-sektionen er et eller flere XML-skemaer.

- **Message**
  Beskriver de beskeder, en webservice benytter sig af som input og output. Messages består af en eller flere logiske dele (parts). Hver del er associeret med en datatype fra "Types"-sektionen – dvs. et XML-skema-element.
- **Operation**
  Angiver hvilke operationer webservicen udstiller, hvilke beskeder (messages) der indgår i operationerne, og om disse er beskeder er input- eller output-beskeder.
- **PortType**
  En PortType er en abstrakt gruppering af en eller flere operationer.
- **Binding**
  En binding definerer format- og protokoldetaljer for operationer og beskeder.
- **Port**
  En Port-sektion definerer et individuelt endpoint – en adresse, f.eks. på internettet – med hvilken man tilknytter en binding.
- **Service**
  Angiver adressen på webserviceudbyderen.

Eksemplet nedenfor viser en WSDL-fil for webservicen PersonWebService, der kan kaldes på adressen http://medcom.dk/services/sample/personwebservice, som har en operation, hentNavn, der benytter SOAP over HTTP-bindingen. Operationen kræver, at der sendes en inputbesked, hentNavnSoapIn, og vil efter udførsel returnere en outputbesked, hentNavnSoapOut. Inputbeskeden anvender en lokal type kaldet hentNavnRequest, som udelukkende specificerer et personnummer. Outputbeskeden returnerer svaret i den lokale type hentNavnResponse med "navn" som eneste værdi.

Med andre ord kan man med denne webservice finde ud af en persons navn, hvis man kender CPR-nummeret:

```
<?xml version="1.0" encoding="utf-8"?>
<definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
   xmlxmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
   xmlxmlns:s="http://www.w3.org/2001/xmlSchema"
   xmlxmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
   xmlns:tns="http://medcom.dk/Personinfo" xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
   targetNamespace="http://medcom.dk/Personinfo" name="Personinformation"
   xmlns="http://schemas.xmlsoap.org/wsdl/">
   <types>
       <s:schema elementFormDefault="qualified" targetNamespace="http://medcom.dk/Personinfo">
           <s:element name="hentNavnRequest">
               <s:complexType>
                   <s:sequence>
                       <s:element name="Personnummer" type="s:string"/>
                   </s:sequence>
               </s:complexType>
           </s:element>
           <s:element name="hentNavnResponse">
               <s:complexType>
                   <s:sequence>
                       <s:element name="Navn" type="s:string"/>
                   </s:sequence>
               </s:complexType>
           </s:element>
       </s:schema>
   </types>
   <message name="hentNavnSoapIn">
```

```
        <part name="parameters" element="tns:hentNavnRequest"/>
    </message>
    <message name="hentNavnSoapOut">
        <part name="parameters" element="tns:hentNavnResponse"/>
    </message>
    <portType name="PersoninformationSoap">
        <operation name="hentNavn">
            <input message="tns:hentNavnSoapIn"/>
            <output message="tns:hentNavnSoapOut"/>
            <fault name="CprNummerIkkeFundet" message="tns:CprIkkeFundetFault"/>
        </operation>
    </portType>
    <binding name="PersoninformationSoap" type="tns:PersoninformationSoap">
        <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
        <operation name="hentNavn">
            <soap:operation soapAction="http://medcom.dk/Personinfo/hentNavn" style="document"/>
            <input><soap:body use="literal"/></input>
            <output><soap:body use="literal"/></output>
            <fault><soap:body use="literal"/></fault>
        </operation>
    </binding>
    <service name="PersonWebService">
        <port name="PersoninformationPort" binding="tns:PersoninformationSoap">
            <soap:address location="http://medcom.dk/services/sample/personwebservice"/>
        </port>
    </service>
</definitions>
```

Figur 2: Eksempel på WSDL

En klient kan nu danne en SOAP-besked på baggrund af den definerede WSDL og spørge på personen med CPR-nummer 1212121212's navn:

Request message XML

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <hentNavn XMLns="http://medcom.dk/Personinfo">
      <Personnummer>1212121212</Personnummer>
    </hentNavn>
  </soap:Body>
</soap:Envelope>
```

Figur 3: Eksempel på request

Og efterfølgende få et svar fra webservicen om, at vedkommende hedder Jens Hansen:

Response message XML

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope
:xsi="http://www.w3.org/2001/XMLSchema-instance"
:xsd="http://www.w3.org/2001/XMLSchema"
:soap="http://schemas.xmlsoap.org/soap/envelope/" >
 <soap:Body>
   <hentNavnResponse xmlns="http://medcom.dk/Personinfo">
     <Navn>Jens Hansen</Navn>
   </hentNavnResponse>
 </soap:Body>
</soap:Envelope>
```

Figur 4: Eksempel på response

WSDL-skabelon

Den Gode Webservices kompatible webservices overholder nedenstående skabelon i dokumentation af snitfladen. I skabelonen er variable værdier vist med **fed** skrift, og alle andre elementer skal tages for pålydende.

Det er værd at bemærke, at DGWS webservices anvender document-literal style og HTTP binding samt en fast SOAP:Header. DGWS erklærer desuden mulige fejl med faultelementet ”dgwsfault”, som specificerer indholdet af detail elementet for soap:Fault. I DGWS er detail elementet altid en medcom:FaultCode, der indeholder fejlkoden fra enumerationslisten eller en webservicespecifik kode.

DGWS WSDL stiller krav om at der skal være en medcom header og en wsse header i kuverten. Elementerne er indlejret i tråd med WS-I Basic Profile, men da webservice verdenen forventes fremover at beskrive krav til intrastruktur i form af WS-Policy specifikationer kan en senere version af DGWS ændre på denne praksis.

```
<?xml version="1.0"?>
<definitions name="... wsdl navn ..." targetNamespace="... skema url ..."
xmlns:tns="http://www.medcom.dk/dglws/1.0/wsdl" xmlns:myns="... skema url ..."
xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns="http://schemas.xmlsoap.org/wsdl/">
 <types>
  <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <xsd:import namespace="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd"
schemaLocation="http://www.medcom.dk/schemas/dgws/1.0/medcom.xsd"/>
  </xsd:schema>
  <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <xsd:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd" schemaLocation="http://www.medcom.dk/schemas/dgws/1.0//wsse.xsd"/>
  </xsd:schema>
  ... egne xml skemaer her ...
 </types>
 <message name="... input message navn ...">
  <part name="header_wsse" element="wsse:Security"/>
  <part name="header_medcom" element="medcom:Header"/>
  <part name="parameters" element="... input datatype ..."/>
 </message>
 <message name="... output message navn ...">
  <part name="header_wsse" element="wsse:Security"/>
  <part name="header_medcom" element="medcom:Header"/>
  <part name="parameters" element="... output datatype ..."/>
 </message>
 <message name="DgwsFaultMessage">
  <part name="fault_medcom" element="medcom:FaultCode"/>
 </message>
  ... flere message elementer ...
 <portType name="... port type navn ...">
   <operation name="... operations navn ...">
     <input message="... input message navn ..."/>
     <output message="... output message navn ..."/>
     <fault name="dgwsfault" message="tns:DgwsFaultMessage"/>
   </operation>
   ... flere operation elementer ...
 </portType>
 <binding name="... soap binding navn ..." type="... port type navn ...">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="... operations navn ...">
   <soap:operation soapAction="... soapaction http header værdi ..." style="document"/>
   <input>
    <soap:header message="... input message navn ..." part="header_wsse" use="literal"/>
    <soap:header message="... input message navn ..." part="header_securitylevel" use="literal"/>
    <soap:body use="literal" parts="parameters"/>
   </input>
   <output>
```

```
    <soap:header message="... output message navn ..." part="header_wsse" use="literal"/>
    <soap:header message="... output message navn ..." part="header_securitylevel" use="literal"/>
    <soap:body use="literal" parts="parameters"/>
   </output>
   <fault name="dgwsfault">
    <soap:fault name="dgwsfault"/>
   </fault>
  </operation>
 </binding>
 <service name="... webservice navn ...">
   <port name="... port navn ..." binding="... soap binding navn ...">
     <soap:address location="... webservice url ..."/>
   </port>
 </service>
</definitions>
```

De variable værdier i skabelonen er beskrevet nedenfor:

Variabel	Værdi
wsdl navn	Navn på WSDL-fil
skema namespace	Namespace-angivelse (f.eks. ”myns:”) for eget skema
skema url	URL, hvor dette skema kan findes
xml skema	Indlejret XML-skema
input message navn	Navnet på den message, der indeholder inputdata til webservicen
input data type	XML-skema-type fra et importeret eller indlejret skema for inputmessage
output message navn	Navnet på den message, der indeholder outputdata til webservicen
output data type	XML-skema-type fra et importeret eller indlejret skema for outputmessage
flere message elementer	Der er ingen begrænsninger på antallet af message-elementer
port type navn	Navnet på en porttype (abstrakt gruppering af operationer)
operations navn	Navnet på en enkelt webserviceoperation
input message navn	Reference til ”input message navn” for den message, der er input til operationen
output message navn	Reference til ”output message navn” for den message, der er output af operationen
flere operation elementer	Der er ingen begrænsninger på antallet af messageelementer
soap binding navn	Navn på den HTTP SOAP-binding, der anvendes. Bruges til senere reference
soapaction http header værdi	I HTTP SOAP-bindingen skal HTTP-headeren SOAPAction sættes til værdien af dette felt.
webservice navn	Navnet på hele denne webservice
port navn	Navnet på den port, der definerer, hvilke operationer (porttype) over hvilken protokol (binding) denne webservice udstiller. Der kan generelt være flere porte, men i DGWS anvendes kun HTTP-bindingen, så derfor er der kun én alligevel.
webservice url	Den HTTP(S)-URL, som webservicen kan findes på

Eksempel: Den Gode Labreport WSDL

Nedenstående WSDL beskriver servicen LabReportsService, der giver mulighed for at hente laboratoriesvar. Servicen er dokumenteret i "Den Gode Laboratorie Webservice". Bemærk! Den URL, der er angivet i <service>-elementet under <soap:address>, er ikke valid og skal erstattes med det faktiske endpoint, som en implementation af servicen udstilles på.

```
<?xml version="1.0"?>
<definitions name="DGLWS" targetNamespace="http://www.medcom.dk/dglws/1.0/wsdl"
xmlns:tns="http://www.medcom.dk/dglws/1.0/wsdl" xmlns:dglws="http://www.medcom.dk/dglws/1.0/xsd"
xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns="http://schemas.xmlsoap.org/wsdl/">
 <types>
  <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <xsd:import namespace="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd"
schemaLocation="medcom.xsd"/>
  </xsd:schema>
  <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <xsd:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd" schemaLocation="wsse.xsd"/>
  </xsd:schema>
  <xsd:schema targetNamespace="http://www.medcom.dk/dglws/1.0/xsd"
xmlns:tns="http://www.medcom.dk/dglws/1.0/xsd" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
   <xsd:element name="FindLabReportsRequest" type="dglws:ServiceRequest"/>
   <xsd:element name="GetLabReportsRequest" type="dglws:ServiceRequest"/>
   <xsd:element name="FindLabReportsResponse" type="dglws:ServiceResponse"/>
   <xsd:element name="GetLabReportsRespone" type="dglws:ServiceResponse"/>
   <xsd:complexType name="ServiceRequest">
    <xsd:sequence>
     <xsd:element ref="dglws:PatientConsent"/>
     <xsd:element ref="dglws:RequestParameters"/>
    </xsd:sequence>
   </xsd:complexType>
   <xsd:complexType name="ServiceResponse">
    <xsd:sequence>
     <xsd:element ref="dglws:Patient" maxOccurs="1"/>
     <xsd:element ref="dglws:Result" maxOccurs="unbounded"/>
    </xsd:sequence>
   </xsd:complexType>
   <xsd:element name="PatientConsent">
    <xsd:complexType>
     <xsd:sequence>
      <xsd:element ref="dglws:PatientConsentCode"/>
      <xsd:element ref="dglws:PatientConsentRemark"/>
     </xsd:sequence>
    </xsd:complexType>
   </xsd:element>
   <xsd:element name="PatientConsentCode">
    <xsd:simpleType>
     <xsd:restriction base="xsd:string">
      <xsd:enumeration value="SAMTYKKE_GIVET"/>
      <xsd:enumeration value="PATIENT_BEVISTLOES"/>
      <xsd:enumeration value="I_BEHANDLING"/>
     </xsd:restriction>
    </xsd:simpleType>
   </xsd:element>
   <xsd:element name="PatientConsentRemark" type="xsd:string"/>
   <xsd:element name="RequestParameters">
    <xsd:complexType>
     <xsd:sequence>
      <xsd:element ref="dglws:RequestCaseID"/>
      <xsd:element ref="dglws:RequestStartDateTime"/>
      <xsd:element ref="dglws:RequestEndDateTime"/>
      <xsd:element ref="dglws:RequestLatestCases"/>
     </xsd:sequence>
    </xsd:complexType>
   </xsd:element>
```

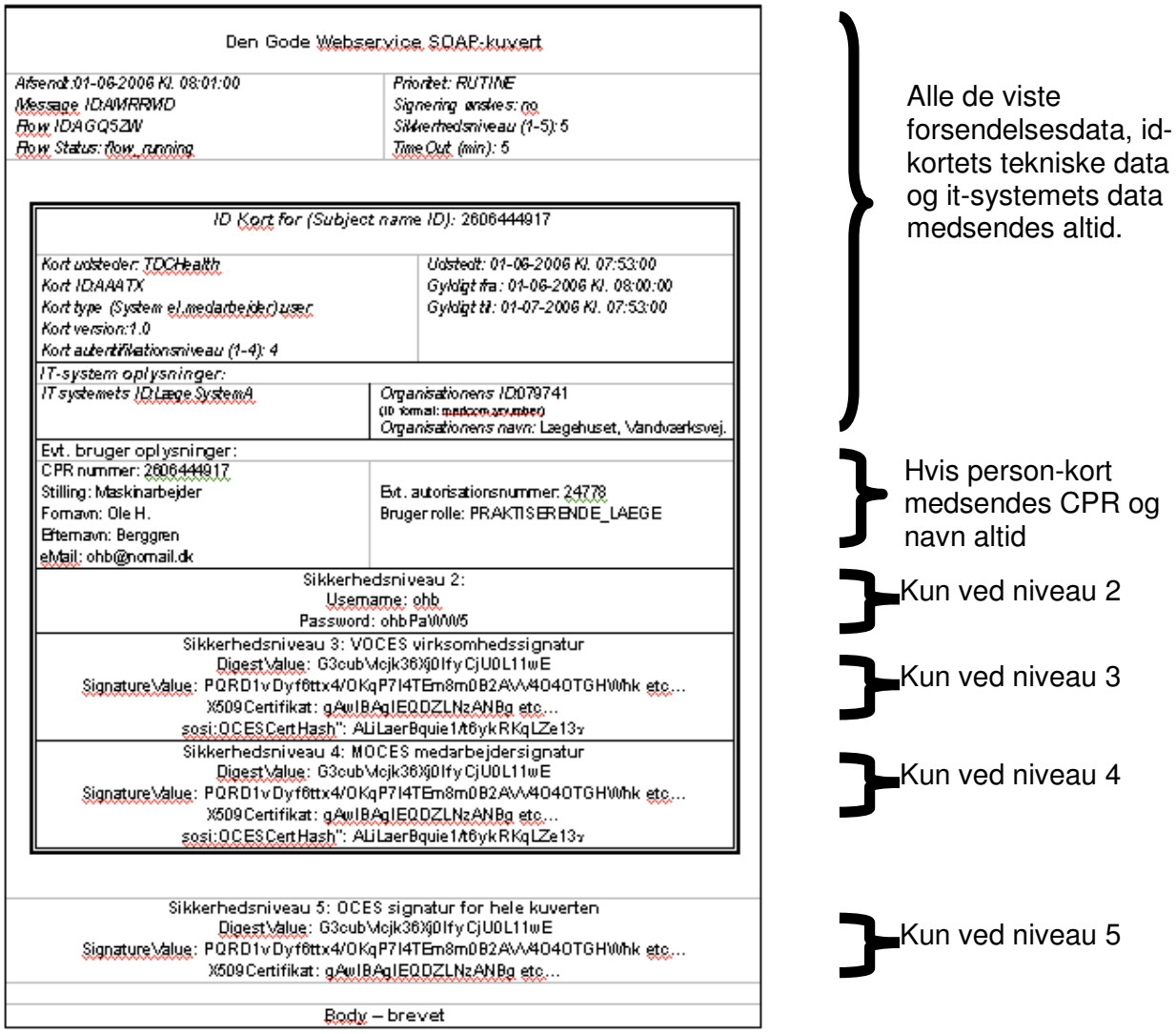
```
  <xsd:element name="RequestCaseID" type="xsd:string"/>
  <xsd:element name="RequestStartDateTime" type="xsd:dateTime"/>
  <xsd:element name="RequestEndDateTime" type="xsd:dateTime"/>
  <xsd:element name="RequestLatestCases" type="xsd:string"/>
  <xsd:element name="Patient">
   <xsd:complexType>
    <xsd:sequence>
     <xsd:element ref="dglws:CivilRegistrationNumber"/>
     <xsd:element ref="dglws:PersonSurnameName"/>
     <xsd:element ref="dglws:PersonGivenName"/>
    </xsd:sequence>
   </xsd:complexType>
  </xsd:element>
  <xsd:element name="CivilRegistrationNumber" type="xsd:string"/>
  <xsd:element name="PersonSurnameName" type="xsd:string"/>
  <xsd:element name="PersonGivenName" type="xsd:string"/>
  <xsd:element name="Result">
   <xsd:complexType>
    <xsd:sequence>
     <xsd:element ref="dglws:Sent.Date.Time"/>
     <xsd:element ref="dglws:Letter.Identifier"/>
     <xsd:element ref="dglws:Letter.TypeCode"/>
     <xsd:element ref="dglws:Sender.OrganisationName"/>
     <xsd:element ref="dglws:Sender.DepartmentName"/>
     <xsd:element ref="dglws:MedComLaboratoryReport" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
   </xsd:complexType>
  </xsd:element>
  <xsd:element name="Sent.Date.Time" type="xsd:dateTime"/>
  <xsd:element name="Letter.Identifier" type="xsd:string"/>
  <xsd:element name="Letter.TypeCode" type="xsd:string"/>
  <xsd:element name="Sender.OrganisationName" type="xsd:string"/>
  <xsd:element name="Sender.DepartmentName" type="xsd:string"/>
  <xsd:element name="MedComLaboratoryReport" type="xsd:anyType"/>
 </xsd:schema>
</types>
<message name="FindLabReportsRequestMessage">
 <part name="header_wsse" element="wsse:Security"/>
 <part name="header_medcom" element="medcom:Header"/>
 <part name="parameters" element="dglws:FindLabReportsRequest"/>
</message>
<message name="FindLabReportsResponseMessage">
 <part name="header_wsse" element="wsse:Security"/>
 <part name="header_medcom" element="medcom:Header"/>
 <part name="parameters" element="dglws:FindLabReportsResponse"/>
</message>
<message name="GetLabReportsRequestMessage">
 <part name="header_wsse" element="wsse:Security"/>
 <part name="header_medcom" element="medcom:Header"/>
 <part name="parameters" element="dglws:GetLabReportsRequest"/>
</message>
<message name="GetLabReportsResponseMessage">
 <part name="header_wsse" element="wsse:Security"/>
 <part name="header_medcom" element="medcom:Header"/>
 <part name="parameters" element="dglws:GetLabReportsResponse"/>
</message>
<message name="DgwsFaultMessage">
 <part name="fault_medcom" element="medcom:FaultCode"/>
</message>
<portType name="LabReportsPortType">
 <operation name="FindLabReports">
  <input message="tns:FindLabReportsRequestMessage"/>
  <output message="tns:FindLabReportsResponseMessage"/>
  <fault name="dgwsfault" message="tns:DgwsFaultMessage"/>
 </operation>
 <operation name="GetLabReports">
  <input message="tns:GetLabReportsRequestMessage"/>
  <output message="tns:GetLabReportsResponseMessage"/>
  <fault name="dgwsfault" message="tns:DgwsFaultMessage"/>
 </operation>
</portType>
<binding name="LabReportsSoapBinding" type="tns:LabReportsPortType">
 <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
 <operation name="FindLabReports">
```

```
    <soap:operation soapAction="http://medcom.com/DGLWS/FindLabReports" style="document"/>
    <input>
     <soap:header message="tns:FindLabReportsRequestMessage" part="header_wsse" use="literal"/>
     <soap:header message="tns:FindLabReportsRequestMessage" part="header_medcom" use="literal"/>
     <soap:body use="literal" parts="parameters"/>
    </input>
    <output>
     <soap:header message="tns:FindLabReportsResponseMessage" part="header_wsse" use="literal"/>
     <soap:header message="tns:FindLabReportsResponseMessage" part="header_medcom" use="literal"/>
     <soap:body use="literal" parts="parameters"/>
    </output>
    <fault name="dgwsfault">
     <soap:fault name="dgwsfault"/>
    </fault>
   </operation>
   <operation name="GetLabReports">
    <soap:operation soapAction="http://medcom.com/DGLWS/GetLabReports" style="document"/>
    <input>
     <soap:header message="tns:GetLabReportsRequestMessage" part="header_wsse" use="literal"/>
     <soap:header message="tns:GetLabReportsRequestMessage" part="header_medcom" use="literal"/>
     <soap:body use="literal" parts="parameters"/>
    </input>
    <output>
     <soap:header message="tns:GetLabReportsResponseMessage" part="header_wsse" use="literal"/>
     <soap:header message="tns:GetLabReportsResponseMessage" part="header_medcom" use="literal"/>
     <soap:body use="literal" parts="parameters"/>
    </output>
    <fault name="dgwsfault">
     <soap:fault name="dgwsfault"/>
    </fault>
   </operation>
  </binding>
  <service name="LabReportsService">
   <port name="LabReportsPort" binding="tns:LabReportsSoapBinding">
    <soap:address location="http://medcom.com/dglws/labreportsws"/>
   </port>
  </service>
</definitions>
```

Bilag 7: XML-liste for Den Gode Webservice

Nedenstående blanket illustrerer, hvor meget dataindhold der maksimalt kan kommunikeres i SOAP-headeren i Den Gode Webservice.

Blanketten er opbygget på samme måde som XML-listen. Til højre for blanketten er der en oversigt over, hvilke data der medsendes på de forskellige sikkerhedsniveauer.



XML-liste

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd" wsu:id="Envelope">
 <soap:Header>
  <wsse:Security>
```

Afsendelsestidspunkt

```
<wsu:Timestamp>
 <wsu:Created>2006-06-01T08:01:00</wsu:Created>
</wsu:Timestamp>
```

Id-kort

```
<saml:Assertion IssueInstant="2006-06-01T07:53:00" Version="2.0" id="IDCard">
 <saml:Issuer>LægeSystemA</saml:Issuer>
 <saml:Subject>
  <saml:NameID Format="medcom:cprnumber">2606444917</saml:NameID>
```

Autentifikationsniveau

```
   <saml:SubjectConfirmation>
    <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-key</saml:ConfirmationMethod>
    <saml:SubjectConfirmationData>
     <ds:KeyInfo>
      <ds:KeyName>OCESSignature</ds:KeyName>
     </ds:KeyInfo>
    </saml:SubjectConfirmationData>
   </saml:SubjectConfirmation>
```

Kort-data

```
  </saml:Subject>
  <saml:Conditions NotBefore="2006-06-01T08:00:00" NotOnOrAfter="2006-07-01T07:53:00"/>
  <saml:AttributeStatement id="IDCardData">
   <saml:Attribute Name="sosi:IDCardID">
    <saml:AttributeValue>AAATX</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="sosi:IDCardVersion">
    <saml:AttributeValue>1.0</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="sosi:IDCardType">
    <saml:AttributeValue>user</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="sosi:AuthenticationLevel">
    <saml:AttributeValue>4</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="sosi:OCESCertHash">
    <saml:AttributeValue>ALiLaerBquie1/t6ykRKqLZe13Y=</saml:AttributeValue>
   </saml:Attribute>
  </saml:AttributeStatement>
```

Bruger-data

```
  <saml:AttributeStatement id="UserLog">
   <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
    <saml:AttributeValue>2606444917</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:UserGivenName">
    <saml:AttributeValue>Ole H.</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:UserSurName">
    <saml:AttributeValue>Berggren</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:UserEmailAddress">
    <saml:AttributeValue>ohb@nomail.dk</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:UserRole">
    <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:UserOccupation">
    <saml:AttributeValue>Maskinarbejder</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:UserAuthorizationCode">
    <saml:AttributeValue>24778</saml:AttributeValue>
   </saml:Attribute>
  </saml:AttributeStatement>
  <saml:AttributeStatement id="SystemLog">
   <saml:Attribute Name="medcom:ITSystemName">
    <saml:AttributeValue>LægeSystemA</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:ynumber">
    <saml:AttributeValue>079741</saml:AttributeValue>
   </saml:Attribute>
   <saml:Attribute Name="medcom:CareProviderName">
    <saml:AttributeValue>Lægehuset, Vandværksvej</saml:AttributeValue>
```

```
  </saml:Attribute>
 </saml:AttributeStatement>
```

Digital signatur for id-kortet

```
 <ds:Signature id="OCESSignature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
   <ds:Reference URI="#IDCard">
    <ds:Transforms>
     <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
     <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>
  PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8G
  qMJh6gh/cA2mgKJ9bqiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2d
  ochINlFg=</ds:SignatureValue>
  <ds:KeyInfo>
   <ds:X509Data>
    <ds:X509Certificate>
    gAwIBAgIEQDZLNzANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UE
    NFUyBTeXN0ZW10ZXN0IENBIElJMB4XDTA1MDYwNjEyMDQw
    MDYwNjEyMzQwMFowfTELMAkGA1UEBhMCREsxLzAtBgNVBAoUJlREQyBUT1RBTEzYU05J
    MgLy8gQ1ZSOjI1NzY3NTM1MT0wFAYDVQQDEw1UZXN0IEJydWdlciAyMCUGA1UEBRMe
    ZSOjI1NzY3NTM1LVJJRDoxMTE4MDYxMDQzMzU2MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
    uvze+4T1i0inhmvafWB2d81q3AG7ds06eGy+eLjQYumaY5EViSv4qyNwmnV6Y1sVi3LpD/
    /wr7+DBanwBUEXnlzRY4No4U3DrDAjvl4NKjdv/Dkg1pMfUwmaIYkQoLTWHe8bCf
     VPXtovQ12CLO7uydoBzTQIDAQABo4ICzTCCAskwDgYDVR0PAQH/BAQDAgP4MCsGA1UdEAQkMCKA
    MTIwNDAwWoEPMjAwNzA2MDYxMjM0MDBaMEYGCCsGAQUFBwEBBDowODA2BggrBgEF
    cDovL3Rlc3Qub2NzcC5jZXJ0aWZpa2F0LmRrL29jc3Avc3RhdHVzMIIBAwYDVR0g
    MIH4MIH1BgkpAQEBAQEBAQIwgecwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cuY2VydGlmaWth
    kay9yZXBvc2l0b3J5MIGzBggrBgEFBQcCAjCBpjAKFgNUREMwAwIBARqBl1REQyBUZXN0IENl
    EgdWRzdGVkZXMgdW5kZXIgT0lEIDEuMS4xLjEuMS4xLjEu
    xLjIuIFREQyBUZXN0IENlcnRpZmljYXRlcyBmcm9tIHRoaXMgQ0EgYXJlIGlzc3VlZCB1bmRl
    xLjEuMS4xLjEuMS4xLjEuMi4wGgYJYIZIAYb4QgENBA0WC2VtcGxveWVlV2ViMCAG
    UdEQQZMBeBFXN1cHBvcnRAY2VydGlmaWthdC5kazCBlgYDVR0fBIGOMIGLMFagVKBSpFAwTjEL
    UEBhMCREsxDDAKBgNVBAoTA1REQzEiMCAGA1UEAxMZVERDIE9DRVMgU3lzdGVtdGVzdCBD
    UEAxMEQ1JMMjAxoC+gLYYraHR0cDovL3Rlc3QuY3JsLm9jZXMuY2VydGlmaWth
    kay9vY2VzLmNybDAfBgNVHSMEGDAWgBQcmAlHGkw4uRDFBClb8fROgGrMfjAdBgNVHQ4EFgQU
    pQWIRbZKFhWkcHOi1bgdX4YwCQYDVR0TBAIwADAZBgkqhkiG9n0HQQAEDDAKGwRWNy4xAwID
    w0BAQUFAAOBgQBp+zmRburdSGirxmMWFFcT4NaP3W+XRPqY3iCiZuW2FcBrTtHy
    uFrjbQHg9RznxAgHIpzu/txQsSqv+m76Ki8zB2+r0fwlYrABvcloPUfRF6pRksYtYNXsnGS
    Re1147c9K315hXG3QMmuU+rBFyvRGkWx0wlf3lOrLg==</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
 </ds:Signature>
</saml:Assertion>
```

Digital signatur for hele SOAP-meddelelsen (niveau 5)

```
 <ds:Signature id="OCESSignature2">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
   <ds:Reference URI="Envelope">
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>
  PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8G
  qMJh6gh/cA2mgKJ9bqiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2d
  ochINlFg=</ds:SignatureValue>
  <ds:KeyInfo>
   <ds:X509Data>
    <ds:X509Certificate>
    gAwIBAgIEQDZLNzANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UE
    NFUyBTeXN0ZW10ZXN0IENBIElJMB4XDTA1MDYwNjEyMDQw
    MDYwNjEyMzQwMFowfTELMAkGA1UEBhMCREsxLzAtBgNVBAoUJlREQyBUT1RBTEzYU05J
```

```
MgLy8gQ1ZSOjI1NzY3NTM1MT0wFAYDVQQDEw1UZXN0IEJydWdlciAyMCUGA1UEBRMe
ZSOjI1NzY3NTM1LVJJRDoxMTE4MDYxMDQzMzU2MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
uvze+4T1i0inhmvafWB2d81q3AG7ds06eGy+eLjQYumaY5EViSv4qyNwmnV6Y1sVi3LpD/
/wr7+DBanwBUEXnlzRY4No4U3DrDAjvl4NKjdv/Dkg1pMfUwmaIYkQoLTWHe8bCf
 VPXtovQ12CLO7uydoBzTQIDAQABo4ICzTCCAskwDgYDVR0PAQH/BAQDAgP4MCsGA1UdEAQkMCKA
MTIwNDAwWoEPMjAwNzA2MDYxMjM0MDBaMEYGCCsGAQUFBwEBBDowODA2BggrBgEF
cDovL3Rlc3Qub2NzcC5jZXJ0aWZpa2F0LmRrL29jc3Avc3RhdHVzMIIBAwYDVR0g
MIH4MIH1BgkpAQEBAQEBAQIwgecwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cuY2VydGlmaWth
kay9yZXBvc2l0b3J5MIGzBggrBgEFBQcCAjCBpjAKFgNUREMwAwIBARqBl1REQyBUZXN0IENl
EgdWRzdGVkZXMgdW5kZXIgT0lEIDEuMS4xLjEuMS4xLjEu
xLjIuIFREQyBUZXN0IENlcnRpZmljYXRlcyBmcm9tIHRoaXMgQ0EgYXJlIGlzc3VlZCB1bmRl
xLjEuMS4xLjEuMS4xLjEuMi4wGgYJYIZIAYb4QgENBA0WC2VtcGxveWVlV2ViMCAG
UdEQQZMBeBFXN1cHBvcnRAY2VydGlmaWthdC5kazCBlgYDVR0fBIGOMIGLMFagVKBSpFAwTjEL
UEBhMCREsxDDAKBgNVBAoTA1REQzEiMCAGA1UEAxMZVERDIE9DRVMgU3lzdGVtdGVzdCBD
UEAxMEQ1JMMjAxoC+gLYYraHR0cDovL3Rlc3QuY3JsLm9jZXMuY2VydGlmaWth
kay9vY2VzLmNybDAfBgNVHSMEGDAWgBQcmAlHGkw4uRDFBClb8fROgGrMfjAdBgNVHQ4EFgQU
pQWIRbZKFhWkcHOi1bgdX4YwCQYDVR0TBAIwADAZBgkqhkiG9n0HQQAEDDAKGwRWNy4xAwID
w0BAQUFAAOBgQBp+zmRburdSGirxmMWFFcT4NaP3W+XRPqY3iCiZuW2FcBrTtHy
uFrjbQHg9RznxAgHlpzu/txQsSqv+m76Ki8zB2+r0fwlYrABvcloPUfRF6pRksYtYNXsnGS
Re1147c9K315hXG3QMmuU+rBFyvRGkWx0wlf3lOrLg==</ds:X509Certificate>
   </ds:X509Data>
  </ds:KeyInfo>
 </ds:Signature>
</wsse:Security>
```

MedCom-sikkerhedsdata

```
<medcom:Header>
 <medcom:SecurityLevel>5</medcom:SecurityLevel>
 <medcom:TimeOut>480</medcom:TimeOut>
 <medcom:Linking>
  <medcom:FlowID>AMRRMD</medcom:FlowID>
  <medcom:MessageID>AGQ5ZW</medcom:MessageID>
 </medcom:Linking>
 <medcom:Priority>RUTINE</medcom:Priority>
<medcom:Header>
</soap:Header>
<soap:Body/>
</soap:Envelope>
```

Ved niveau 2 vil elementet <saml:SubjectConfirmation> have følgende indhold:

```
<saml:SubjectConfirmation>
  <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-key</saml:ConfirmationMethod>
  <saml:SubjectConfirmationData>
    <wsse:UsernameToken>
      <wsse:Username>Ole H. Berggren</wsse:Username>
      <wsse:Password>ohbPaWW5</wsse:Password>
    </wsse:UsernameToken>
  </saml:SubjectConfirmationData>
</saml:SubjectConfirmation>
```

Bilag 8: Testeksempler

I dette bilag findes valide testeksempler på DGWS-konvolutter for samtlige sikkerhedsniveauer.

Request-niveau 1

På det laveste sikkerhedsniveau er der ingen akkreditiver indlejret i kuverten:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig" xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-
1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:id="Envelope">
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2006-06-01T08:01:00</wsu:Created>
   </wsu:Timestamp>
   <saml:Assertion id="IDCard" IssueInstant="2006-06-01T07:53:00" Version="2.0">
    <saml:Issuer>LægeSystemA</saml:Issuer>
    <saml:Subject>
     <saml:NameID Format="medcom:cprnumber">2606444917</saml:NameID>
    </saml:Subject>
    <saml:Conditions NotBefore="2006-06-01T08:00:00" NotOnOrAfter="2006-07-01T07:53:00"/>
    <saml:AttributeStatement id="IDCardData">
     <saml:Attribute Name="sosi:IDCardID">
      <saml:AttributeValue>AAATX</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardVersion">
      <saml:AttributeValue>1.0</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardType">
      <saml:AttributeValue>user</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:AuthenticationLevel">
      <saml:AttributeValue>1</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <saml:AttributeStatement id="UserLog">
     <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
      <saml:AttributeValue>2606444917</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserGivenName">
      <saml:AttributeValue>Ole H.</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserSurName">
      <saml:AttributeValue>Berggren</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserEmailAddress">
      <saml:AttributeValue>ohb@nomail.dk</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserRole">
      <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserOccupation">
      <saml:AttributeValue>Maskinarbejder</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserAuthorizationCode">
      <saml:AttributeValue>24778</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <saml:AttributeStatement id="SystemLog">
     <saml:Attribute Name="medcom:ITSystemName">
      <saml:AttributeValue>LægeSystemA</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:ynumber">
```

```
      <saml:AttributeValue>079741</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:CareProviderName">
      <saml:AttributeValue>Lægehuset, Vandværksvej</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
   </saml:Assertion>
  </wsse:Security>
  <medcom:Header>
   <medcom:SecurityLevel>1</medcom:SecurityLevel>
   <medcom:TimeOut>1440</medcom:TimeOut>
   <medcom:Linking>
    <medcom:FlowID>AMRRMD</medcom:FlowID>
    <medcom:MessageID>AGQ5ZW</medcom:MessageID>
   </medcom:Linking>
   <medcom:Priority>RUTINE</medcom:Priority>
  </medcom:Header>
 </soap:Header>
 <soap:Body/>
</soap:Envelope>
```

Request-niveau 2

Nedenstående eksempel viser en forespørgsel på niveau 2, dvs. hvor der anvendes brugernavn og password som akkreditiver:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig" xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-
1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:id="Envelope">
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2006-06-01T08:01:00</wsu:Created>
   </wsu:Timestamp>
   <saml:Assertion id="IDCard" IssueInstant="2006-06-01T07:53:00" Version="2.0">
    <saml:Issuer>LægeSystemA</saml:Issuer>
    <saml:Subject>
     <saml:NameID Format="medcom:cprnumber">2606444917</saml:NameID>
     <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-
key</saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
       <wsse:UsernameToken>
        <wsse:Username>Ole H..Berggren</wsse:Username>
        <wsse:Password>ohbPaWW5</wsse:Password>
       </wsse:UsernameToken>
      </saml:SubjectConfirmationData>
     </saml:SubjectConfirmhation>
    </saml:Subject>
    <saml:Conditions NotBefore="2006-06-01T08:00:00" NotOnOrAfter="2006-07-01T07:53:00"/>
    <saml:AttributeStatement id="IDCardData">
     <saml:Attribute Name="sosi:IDCardID">
      <saml:AttributeValue>AAATX</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardVersion">
      <saml:AttributeValue>1.0</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardType">
      <saml:AttributeValue>user</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:AuthenticationLevel">
      <saml:AttributeValue>2</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <saml:AttributeStatement id="UserLog">
     <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
      <saml:AttributeValue>2606444917</saml:AttributeValue>
```

```
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserGivenName">
       <saml:AttributeValue>Ole H.</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserSurName">
       <saml:AttributeValue>Berggren</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserEmailAddress">
       <saml:AttributeValue>ohb@nomail.dk</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserRole">
       <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserOccupation">
       <saml:AttributeValue>Maskinarbejder</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:UserAuthorizationCode">
       <saml:AttributeValue>24778</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:ynumber">
       <saml:AttributeValue>079741</saml:AttributeValue>
      </saml:Attribute>
      <saml:Attribute Name="medcom:CareProviderName">
       <saml:AttributeValue>Lægehuset, Vandværksvej</saml:AttributeValue>
      </saml:Attribute>
     </saml:AttributeStatement>
     <saml:AttributeStatement id="SystemLog">
      <saml:Attribute Name="medcom:ITSystemName">
       <saml:AttributeValue>LægeSystemA</saml:AttributeValue>
      </saml:Attribute>
     </saml:AttributeStatement>
    </saml:Assertion>
   </wsse:Security>
   <medcom:Header>
    <medcom:SecurityLevel>2</medcom:SecurityLevel>
    <medcom:TimeOut>1440</medcom:TimeOut>
    <medcom:Linking>
     <medcom:FlowID>AMRRMD</medcom:FlowID>
     <medcom:MessageID>AGQ5ZW</medcom:MessageID>
    </medcom:Linking>
    <medcom:Priority>RUTINE</medcom:Priority>
   </medcom:Header>
  </soap:Header>
  <soap:Body/>
</soap:Envelope>
```

Request-niveau 3 og 4

Nedenstående eksempel viser en forespørgsel på niveau 3 eller 4, dvs. hvor der anvendes en digital signatur skabt med et VOCES- eller MOCES-certifikat. I dette tilfælde er der tale om en MOCES-signatur.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-
1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:id="Envelope">
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2006-06-01T08:01:00</wsu:Created>
   </wsu:Timestamp>
   <saml:Assertion IssueInstant="2006-06-01T07:53:00" Version="2.0" id="IDCard">
    <saml:Issuer>TDCHealth</saml:Issuer>
    <saml:Subject>
     <saml:NameID Format="medcom:cprnumber">2606444917</saml:NameID>
     <saml:SubjectConfirmation>
```

```
      <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-
key</saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
       <ds:KeyInfo>
        <ds:KeyName>OCESSignature</ds:KeyName>
       </ds:KeyInfo>
      </saml:SubjectConfirmationData>
     </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Conditions NotBefore="2006-06-01T08:00:00" NotOnOrAfter="2006-07-01T07:53:00"/>
    <saml:AttributeStatement id="IDCardData">
     <saml:Attribute Name="sosi:IDCardID">
      <saml:AttributeValue>AAATX</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardVersion">
      <saml:AttributeValue>1.0</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:IDCardType">
      <saml:AttributeValue>user</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:AuthenticationLevel">
      <saml:AttributeValue>4</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="sosi:OCESCertHash">
      <saml:AttributeValue>ALiLaerBquie1/t6ykRKqLZe13Y=</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <saml:AttributeStatement id="UserLog">
     <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
      <saml:AttributeValue>2606444917</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserGivenName">
      <saml:AttributeValue>Ole H.</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserSurName">
      <saml:AttributeValue>Berggren</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserEmailAddress">
      <saml:AttributeValue>ohb@nomail.dk</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserRole">
      <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserOccupation">
      <saml:AttributeValue>Maskinarbejder</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:UserAuthorizationCode">
      <saml:AttributeValue>24778</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <saml:AttributeStatement id="SystemLog">
     <saml:Attribute Name="medcom:ITSystemName">
      <saml:AttributeValue>LægeSystemA</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:ynumber">
      <saml:AttributeValue>079741</saml:AttributeValue>
     </saml:Attribute>
     <saml:Attribute Name="medcom:CareProviderName">
      <saml:AttributeValue>Lægehuset, Vandværksvej</saml:AttributeValue>
     </saml:Attribute>
    </saml:AttributeStatement>
    <ds:Signature id="OCESSignature">
     <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <ds:Reference URI="#IDCard">
       <ds:Transforms>
        <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
        <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
       </ds:Transforms>
       <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
       <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
      </ds:Reference>
     </ds:SignedInfo>
```

```
      <ds:SignatureValue>
      PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8G
      qMJh6gh/cA2mgKJ9bqiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2d
      ochINlFg=</ds:SignatureValue>
      <ds:KeyInfo>
       <ds:X509Data>
        <ds:X509Certificate>
        gAwIBAgIEQDZLNzANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UE
        NFUyBTeXN0ZW10ZXN0IENBIElJMB4XDTA1MDYwNjEyMDQw
        MDYwNjEyMzQwMFowfTELMAkGA1UEBhMCREsxLzAtBgNVBAoUJlREQyBUT1RBTEzYU05J
        MgLy8gQ1ZSOjI1NzY3NTM1MT0wFAYDVQQDEw1UZXN0IEJydWdlciAyMCUGA1UEBRMe
        ZSOjI1NzY3NTM1LVJJRDoxMTE4MDYxMDQzMzU2MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
        uvze+4T1i0inhmvafWB2d81q3AG7ds06eGy+eLjQYumaY5EViSv4qyNwmnV6Y1sVi3LpD/
        /wr7+DBanwBUEXnlzRY4No4U3DrDAjvl4NKjdv/Dkg1pMfUwmaIYkQoLTWHe8bCf
         VPXtovQ12CLO7uydoBzTQIDAQABo4ICzTCCAskwDgYDVR0PAQH/BAQDAgP4MCsGA1UdEAQkMCKA
        MTIwNDAwWoEPMjAwNzA2MDYxMjM0MDBaMEYGCCsGAQUFBwEBBDowODA2BggrBgEF
        cDovL3Rlc3Qub2NzcC5jZXJ0aWZpa2F0LmRrL29jc3Avc3RhdHVzMIIBAwYDVR0g
        MIH4MIH1BgkpAQEBAQEBAQIwgecwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cuY2VydGlmaWth
        kay9yZXBvc2l0b3J5MIGzBggrBgEFBQcCAjCBpjAKFgNUREMwAwIBARqBl1REQyBUZXN0IENl
        EgdWRzdGVkZXMgdW5kZXIgT0lEIDEuMS4xLjEuMS4xLjEu
        xLjIuIFREQyBUZXN0IENlcnRpZmljYXRlcyBmcm9tIHRoaXMgQ0EgYXJlIGlzc3VlZCB1bmRl
        xLjEuMS4xLjEuMS4xLjEuMi4wGgYJYIZIAYb4QgENBA0WC2VtcGxveWVlV2ViMCAG
        UdEQQZMBeBFXN1cHBvcnRAY2VydGlmaWthdC5kazCBlgYDVR0fBIGOMIGLMFagVKBSpFAwTjEL
        UEBhMCREsxDDAKBgNVBAoTA1REQzEiMCAGA1UEAxMZVERDIE9DRVMgU3lzdGVtdGVzdCBD
        UEAxMEQ1JMMjAxoC+gLYYraHR0cDovL3Rlc3QuY3JsLm9jZXMuY2VydGlmaWth
        kay9vY2VzLmNybDAfBgNVHSMEGDAWgBQcmA1HGkw4uRDFBClb8fROgGrMfjAdBgNVHQ4EFgQU
        pQWIRbZKFhWkcHOi1bgdX4YwCQYDVR0TBAIwADAZBgkqhkiG9n0HQQAEDDAKGwRWNy4xAwID
        w0BAQUFAAOBgQBp+zmRburdSGirxmMWFFcT4NaP3W+XRPqY3iCiZuW2FcBrTtHy
        uFrjbQHg9RznxAgHIpzu/txQsSqv+m76Ki8zB2+r0fwlYrABvcloPUfRF6pRksYtYNXsnGS
        Re1147c9K315hXG3QMmuU+rBFyvRGkWx0wIf3lOrLg==</ds:X509Certificate>
       </ds:X509Data>
      </ds:KeyInfo>
     </ds:Signature>
    </saml:Assertion>
   </wsse:Security>
   <medcom:Header>
    <medcom:SecurityLevel>4</medcom:SecurityLevel>
    <medcom:Linking>
     <medcom:FlowID>AMRRMD</medcom:FlowID>
     <medcom:MessageID>AGQ5ZW</medcom:MessageID>
    </medcom:Linking>
    <medcom:Priority>RUTINE</medcom:Priority>
   </medcom:Header>
  </soap:Header>
  <soap:Body/>
</soap:Envelope>
```

Request-niveau 5

Dette request indeholder en digital signatur på hele kuverten, dvs. sikkerhedsniveau 5, og et digitalt signeret id-kort med autentifikationsniveau 4.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-
1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:id="Envelope">
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2006-06-01T08:01:00</wsu:Created>
   </wsu:Timestamp>
   <saml:Assertion IssueInstant="2006-06-01T07:53:00" Version="2.0" id="IDCard">
    <saml:Issuer>LægeSystemA</saml:Issuer>
    <saml:Subject>
     <saml:NameID Format="medcom:cprnumber">2606444917</saml:NameID>
     <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-
key</saml:ConfirmationMethod>
      <saml:SubjectConfirmationData>
```

```
    <ds:KeyInfo>
     <ds:KeyName>OCESSignature</ds:KeyName>
    </ds:KeyInfo>
   </saml:SubjectConfirmationData>
  </saml:SubjectConfirmation>
 </saml:Subject>
 <saml:Conditions NotBefore="2006-06-01T08:00:00" NotOnOrAfter="2006-07-01T07:53:00"/>
 <saml:AttributeStatement id="IDCardData">
  <saml:Attribute Name="sosi:IDCardID">
   <saml:AttributeValue>AAATX</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:IDCardVersion">
   <saml:AttributeValue>1.0</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:IDCardType">
   <saml:AttributeValue>user</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:AuthenticationLevel">
   <saml:AttributeValue>4</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="sosi:OCESCertHash">
   <saml:AttributeValue>ALiLaerBquie1/t6ykRKqLZe13Y=</saml:AttributeValue>
  </saml:Attribute>
 </saml:AttributeStatement>
 <saml:AttributeStatement id="UserLog">
  <saml:Attribute Name="medcom:UserCivilRegistrationNumber">
   <saml:AttributeValue>2606444917</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserGivenName">
   <saml:AttributeValue>Ole H.</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserSurName">
   <saml:AttributeValue>Berggren</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserEmailAddress">
   <saml:AttributeValue>ohb@nomail.dk</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserRole">
   <saml:AttributeValue>PRAKTISERENDE_LAEGE</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserOccupation">
   <saml:AttributeValue>Maskinarbejder</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:UserAuthorizationCode">
   <saml:AttributeValue>24778</saml:AttributeValue>
  </saml:Attribute>
 </saml:AttributeStatement>
 <saml:AttributeStatement id="SystemLog">
  <saml:Attribute Name="medcom:ITSystemName">
   <saml:AttributeValue>LægeSystemA</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:ynumber">
   <saml:AttributeValue>079741</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="medcom:CareProviderName">
   <saml:AttributeValue>Lægehuset, Vandværksvej</saml:AttributeValue>
  </saml:Attribute>
 </saml:AttributeStatement>
 <ds:Signature id="OCESSignature">
  <ds:SignedInfo>
   <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
   <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
   <ds:Reference URI="#IDCard">
    <ds:Transforms>
     <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
     <ds:Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
   </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>
  PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8G
  qMJh6gh/cA2mgKJ9bqiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2d
```

```
    ochINlFg=</ds:SignatureValue>
    <ds:KeyInfo>
     <ds:X509Data>
      <ds:X509Certificate>
      gAwIBAgIEQDZLNzANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UE
      NFUyBTeXN0ZW10ZXN0IENBIElJMB4XDTA1MDYwNjEyMDQw
      MDYwNjEyMzQwMFowfTELMAkGA1UEBhMCREsxLzAtBgNVBAoUJlREQyBUT1RBTEzYU05J
      MgLy8gQ1ZSOjI1NzY3NTM1MT0wFAYDVQQDEw1UZXN0IEJydWdlciAyMCUGA1UEBRMe
      ZSOjI1NzY3NTM1LVJJRDoxMTE4MDYxMDQzMzU2MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
      uvze+4T1i0inhmvafWB2d81q3AG7ds06eGy+eLjQYumaY5EViSv4qyNwmnV6Y1sVi3LpD/
      /wr7+DBanwBUEXnlzRY4No4U3DrDAjvl4NKjdv/Dkg1pMfUwmaIYkQoLTWHe8bCf
       VPXtovQ12CLO7uydoBzTQIDAQABo4ICzTCCAskwDgYDVR0PAQH/BAQDAgP4MCsGA1UdEAQkMCKA
      MTIwNDAwWoEPMjAwNzA2MDYxMjM0MDBaMEYGCCsGAQUFBwEBBDowODA2BggrBgEF
      cDovL3Rlc3Qub2NzcC5jZXJ0aWZpa2F0LmRrL29jc3Avc3RhdHVzMIIBAwYDVR0g
      MIH4MIH1BgkpAQEBAQEBAQIwgecwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cuY2VydGlmaWth
      kay9yZXBvc2l0b3J5MIGzBggrBgEFBQcCAjCBpjAKFgNUREMwAwIBARqBl1REQyBUZXN0IENl
      EgdWRzdGVkZXMgdW5kZXIgT0lEIDEuMS4xLjEuMS4xLjEu
      xLjIuIFREQyBUZXN0IENlcnRpZmljYXRlcyBmcm9tIHRoaXMgQ0EgYXJlIGlzc3VlZCB1bmRl
      xLjEuMS4xLjEuMS4xLjEuMi4wGgYJYIZIAYb4QgENBA0WC2VtcGxveWVlV2ViMCAG
      UdEQQZMBeBFXN1cHBvcnRAY2VydGlmaWthdC5kazCBlgYDVR0fBIGOMIGLMFagVKBSpFAwTjEL
      UEBhMCREsxDDAKBgNVBAoTA1REQzEiMCAGA1UEAxMZVERDIE9DRVMgU3lzdGVtdGVzdCBD
      UEAxMEQ1JMMjAxoC+gLYYraHR0cDovL3Rlc3QuY3JsLm9jZXMuY2VydGlmaWth
      kay9vY2VzLmNybDAfBgNVHSMEGDAWgBQcmAlHGkw4uRDFBC1b8fROgGrMfjAdBgNVHQ4EFgQU
      pQWIRbZKFhWkcHOi1bgdX4YwCQYDVR0TBAIwADAZBgkqhkiG9n0HQQAEDDAKGwRWNy4xAwID
      w0BAQUFAAOBgQBp+zmRburdSGirxmMWFFcT4NaP3W+XRPqY3iCiZuW2FcBrTtHy
      uFrjbQHg9RznxAgHIpzu/txQsSqv+m76Ki8zB2+r0fwlYrABvcloPUfRF6pRksYtYNXsnGS
      Re1147c9K315hXG3QMmuU+rBFyvRGkWx0wIf3lOrLg==</ds:X509Certificate>
     </ds:X509Data>
    </ds:KeyInfo>
   </ds:Signature>
  </saml:Assertion>
  <ds:Signature id="OCESSignature2">
   <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
    <ds:Reference URI="Envelope">
     <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
     <ds:DigestValue>G3cubVicjk36Xj0IfyCjU0L11wE=</ds:DigestValue>
    </ds:Reference>
   </ds:SignedInfo>
   <ds:SignatureValue>
   PQRD1vDyf6ttx4/OKqP7I4TEm8m0B2AVV4O4OTGHWhkU9j9PvLQBIx+JdOYKGynzMRTJ8G
   qMJh6gh/cA2mgKJ9bqiNRVedxuw4/QnTYz0Yw/8kSO4X7MjdA7/pn0OwIDGCxkw3y4wJGLRR2d
   ochINlFg=</ds:SignatureValue>
   <ds:KeyInfo>
    <ds:X509Data>
     <ds:X509Certificate>
     gAwIBAgIEQDZLNzANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UE
     NFUyBTeXN0ZW10ZXN0IENBIElJMB4XDTA1MDYwNjEyMDQw
     MDYwNjEyMzQwMFowfTELMAkGA1UEBhMCREsxLzAtBgNVBAoUJlREQyBUT1RBTEzYU05J
     MgLy8gQ1ZSOjI1NzY3NTM1MT0wFAYDVQQDEw1UZXN0IEJydWdlciAyMCUGA1UEBRMe
     ZSOjI1NzY3NTM1LVJJRDoxMTE4MDYxMDQzMzU2MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
     uvze+4T1i0inhmvafWB2d81q3AG7ds06eGy+eLjQYumaY5EViSv4qyNwmnV6Y1sVi3LpD/
     /wr7+DBanwBUEXnlzRY4No4U3DrDAjvl4NKjdv/Dkg1pMfUwmaIYkQoLTWHe8bCf
      VPXtovQ12CLO7uydoBzTQIDAQABo4ICzTCCAskwDgYDVR0PAQH/BAQDAgP4MCsGA1UdEAQkMCKA
     MTIwNDAwWoEPMjAwNzA2MDYxMjM0MDBaMEYGCCsGAQUFBwEBBDowODA2BggrBgEF
     cDovL3Rlc3Qub2NzcC5jZXJ0aWZpa2F0LmRrL29jc3Avc3RhdHVzMIIBAwYDVR0g
     MIH4MIH1BgkpAQEBAQEBAQIwgecwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cuY2VydGlmaWth
     kay9yZXBvc2l0b3J5MIGzBggrBgEFBQcCAjCBpjAKFgNUREMwAwIBARqBl1REQyBUZXN0IENl
     EgdWRzdGVkZXMgdW5kZXIgT0lEIDEuMS4xLjEuMS4xLjEu
     xLjIuIFREQyBUZXN0IENlcnRpZmljYXRlcyBmcm9tIHRoaXMgQ0EgYXJlIGlzc3VlZCB1bmRl
     xLjEuMS4xLjEuMS4xLjEuMi4wGgYJYIZIAYb4QgENBA0WC2VtcGxveWVlV2ViMCAG
     UdEQQZMBeBFXN1cHBvcnRAY2VydGlmaWthdC5kazCBlgYDVR0fBIGOMIGLMFagVKBSpFAwTjEL
     UEBhMCREsxDDAKBgNVBAoTA1REQzEiMCAGA1UEAxMZVERDIE9DRVMgU3lzdGVtdGVzdCBD
     UEAxMEQ1JMMjAxoC+gLYYraHR0cDovL3Rlc3QuY3JsLm9jZXMuY2VydGlmaWth
     kay9vY2VzLmNybDAfBgNVHSMEGDAWgBQcmAlHGkw4uRDFBC1b8fROgGrMfjAdBgNVHQ4EFgQU
     pQWIRbZKFhWkcHOi1bgdX4YwCQYDVR0TBAIwADAZBgkqhkiG9n0HQQAEDDAKGwRWNy4xAwID
     w0BAQUFAAOBgQBp+zmRburdSGirxmMWFFcT4NaP3W+XRPqY3iCiZuW2FcBrTtHy
     uFrjbQHg9RznxAgHIpzu/txQsSqv+m76Ki8zB2+r0fwlYrABvcloPUfRF6pRksYtYNXsnGS
     Re1147c9K315hXG3QMmuU+rBFyvRGkWx0wIf3lOrLg==</ds:X509Certificate>
    </ds:X509Data>
   </ds:KeyInfo>
  </ds:Signature>
 </wsse:Security>
```

```
  <medcom:Header>
   <medcom:SecurityLevel>5</medcom:SecurityLevel>
   <medcom:TimeOut>480</medcom:TimeOut>
   <medcom:Linking>
    <medcom:FlowID>AMRRMD</medcom:FlowID>
    <medcom:MessageID>AGQ5ZW</medcom:MessageID>
   </medcom:Linking>
   <medcom:Priority>RUTINE</medcom:Priority>
  </medcom:Header>
 </soap:Header>
 <soap:Body/>
</soap:Envelope>
```

Response OK

Dette response indeholder et gyldigt svar fra en webserviceudbyder:

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-
1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:id="Envelope">
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2005-08-24T10:03:46</wsu:Created>
   </wsu:Timestamp>
  </wsse:Security>
  <medcom:Linking>
   <medcom:FlowID>AMRRMD</medcom:FlowID>
   <medcom:MessageID>AB76AF</medcom:MessageID>
   <medcom:InResponseToMessageID>AGQ5ZW</medcom:InResponseToMessageID>
  </medcom:Linking>
  <medcom:FlowStatus>flow_finalized_succesfully</medcom:FlowStatus>
 </soap:Header>
 <soap:Body>
  <Emessage xmlns="http://rep.oio.dk/medcom.dk/xml/schemas/2004/06/01/"> MedCom XRPT01
  laboratoriesvar </Emessage>
 </soap:Body>
</soap:Envelope>
```

Response Fejlet

Dette eksempel viser en fejlet besked, hvor webserviceudbyderen har konstateret, at der manglede input til webservicen, og returnerer en webservicespecifik fejlkode ”missing_input”:

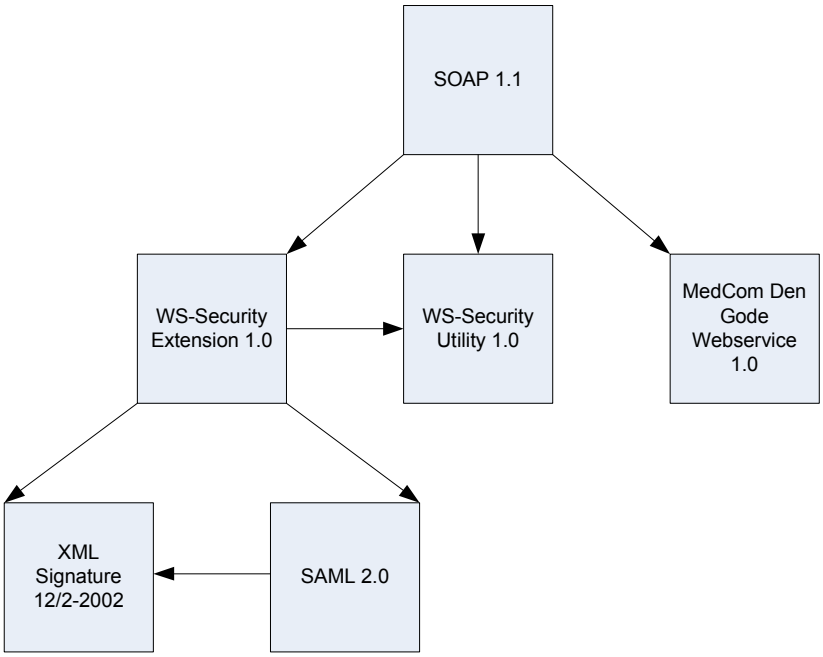
```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-
1.0.xsd" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" id="Envelope">
 <soap:Header>
  <wsse:Security>
   <wsu:Timestamp>
    <wsu:Created>2005-08-24T10:03:46</wsu:Created>
   </wsu:Timestamp>
  </wsse:Security>
  <medcom:Header>
   <medcom:Linking>
    <medcom:FlowID>AMRRMD</medcom:FlowID>
    <medcom:MessageID>AB76AF</medcom:MessageID>
    <medcom:InResponseToMessageID>AGQ5ZW</medcom:InResponseToMessageID>
   </medcom:Linking>
   <medcom:FlowStatus>processing_problem</medcom:FlowStatus>
  </medcom:Header>
```

```
 </soap:Header>
 <soap:Body>
  <soap:Fault>
   <faultcode>Server</faultcode>
   <detail>
     <medcom:FaultCode>invalid_idcard</medcom:FaultCode>
   </detail>
   <faultstring>Id-kort version 3.0 supporteres ikke. Anvend version 2.0 i stedet.</faultstring>
  </soap:Fault>
 </soap:Body>
</soap:Envelope>
```

Bilag 9: XML-skema for Den Gode Webservice

DGWS Formatet bygger på en række internationale standarder, der er defineret ved XML-skemaer. Disse standarder er karakteriseret ved at være meget fleksible, og de tillader ofte mere end én måde at angive den samme information på. Dette strammer DGWS op på ved at definere, præcis hvilke elementer fra de anvendte standarder der må forekomme hvor og hvornår.

Figuren nedenfor illustrerer de anvendte XML-skemaer, og hvordan de anvender hinanden indbyrdes i DGWS sammenhæng.



Figur 5: XML-skema-sammenhænge for DGWS

Dette bilag definerer "skræddersyede" versioner af ovenstående skemaer, der præcist angiver hvilke tags man skal forholde sig til i relation til DGWS. Bemærk venligst, at pånær MedCom skemaet er der altså tale om uofficielle versioner af skemaerne, der med fordel kan benyttes i en udviklingssituation til at checke om den skabte SOAP XML overholder DGWS formatet til punkt og prikke.

SOAP 1.1

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
targetNamespace="http://schemas.xmlsoap.org/soap/envelope/" elementFormDefault="qualified">
 <xs:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd" schemaLocation="wsse.xsd"/>
 <xs:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" schemaLocation="wsu.xsd"/>
 <xs:import namespace="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd" schemaLocation="medcom.xsd"/>
 <xs:element name="Envelope">
  <xs:complexType>
```

```
    <xs:sequence>
     <xs:element ref="soap:Header"/>
     <xs:element ref="soap:Body"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:NCName" use="required"/>
   </xs:complexType>
  </xs:element>
  <xs:element name="Header">
   <xs:complexType>
    <xs:sequence>
     <xs:element ref="wsse:Security" minOccurs="1" maxOccurs="1"/>
     <xs:element ref="medcom:Header" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
   </xs:complexType>
  </xs:element>
  <xs:element name="Body" type="xs:anyType"/>
</xs:schema>
```

WS-Security Extension 1.0

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
targetNamespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
 <xs:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" schemaLocation="wsu.xsd"/>
 <xs:import namespace="http://www.w3.org/2000/09/xmldsig#" schemaLocation="ds.xsd"/>
 <xs:import namespace="urn:oasis:names:tc:SAML:2.0:assertion" schemaLocation="saml.xsd"/>
 <xs:element name="Security">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="wsu:Timestamp" minOccurs="1" maxOccurs="1"/>
    <xs:element ref="saml:Assertion" minOccurs="0" maxOccurs="1"/>
    <xs:element ref="ds:Signature" minOccurs="0" maxOccurs="1"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="UsernameToken">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="wsse:Username" minOccurs="1" maxOccurs="1"/>
    <xs:element ref="wsse:Password" minOccurs="1" maxOccurs="1"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="Username" type="xs:NCName"/>
 <xs:element name="Password" type="xs:NCName"/>
</xs:schema>
```

WS-Security Utility 1.0

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
targetNamespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
elementFormDefault="qualified">
 <xs:attribute name="id" type="xs:NCName"/>
 <xs:element name="Timestamp">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="wsu:Created"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="Created" type="xs:dateTime"/>
</xs:schema>
```

XML Signature af 12/2-2002

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
targetNamespace="http://www.w3.org/2000/09/xmldsig#" elementFormDefault="qualified">
 <xs:element name="Signature">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="ds:SignedInfo"/>
    <xs:element ref="ds:SignatureValue"/>
    <xs:element ref="ds:KeyInfo"/>
   </xs:sequence>
   <xs:attribute name="id" type="xs:NCName" use="required"/>
  </xs:complexType>
 </xs:element>
 <xs:element name="SignedInfo">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="ds:CanonicalizationMethod"/>
    <xs:element ref="ds:SignatureMethod"/>
    <xs:element ref="ds:Reference" maxOccurs="unbounded"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="CanonicalizationMethod">
  <xs:complexType>
   <xs:attribute name="Algorithm" use="required">
    <xs:simpleType>
     <xs:restriction base="xs:anyURI">
      <xs:enumeration value="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <xs:enumeration value="http://www.w3.org/2001/10/xml-exc-c14n#"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:attribute>
  </xs:complexType>
 </xs:element>
 <xs:element name="SignatureMethod">
  <xs:complexType>
   <xs:attribute name="Algorithm" use="required">
    <xs:simpleType>
     <xs:restriction base="xs:anyURI">
      <xs:enumeration value="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:attribute>
  </xs:complexType>
 </xs:element>
 <xs:element name="Reference">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="ds:Transforms" minOccurs="0"/>
    <xs:element ref="ds:DigestMethod"/>
    <xs:element ref="ds:DigestValue"/>
   </xs:sequence>
   <xs:attribute name="URI" use="required"/>
  </xs:complexType>
 </xs:element>
 <xs:element name="Transforms">
  <xs:complexType>
   <xs:sequence maxOccurs="unbounded">
    <xs:element name="Transform">
     <xs:complexType>
      <xs:attribute name="Algorithm" use="required"/>
     </xs:complexType>
    </xs:element>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="DigestMethod">
  <xs:complexType>
   <xs:attribute name="Algorithm" use="required">
    <xs:simpleType>
     <xs:restriction base="xs:anyURI">
      <xs:enumeration value="http://www.w3.org/2000/09/xmldsig#sha1"/>
```

```
      </xs:restriction>
     </xs:simpleType>
    </xs:attribute>
   </xs:complexType>
  </xs:element>
  <xs:element name="DigestValue" type="xs:base64Binary"/>
  <xs:element name="SignatureValue" type="xs:base64Binary"/>
  <xs:element name="KeyInfo">
   <xs:complexType>
    <xs:choice>
     <xs:element ref="ds:KeyName"/>
     <xs:sequence>
      <xs:element ref="ds:X509Data"/>
     </xs:sequence>
    </xs:choice>
   </xs:complexType>
  </xs:element>
  <xs:element name="KeyName" type="xs:NMTOKEN"/>
  <xs:element name="X509Data">
   <xs:complexType>
    <xs:sequence>
     <xs:element ref="ds:X509Certificate"/>
    </xs:sequence>
   </xs:complexType>
  </xs:element>
  <xs:element name="X509Certificate" type="xs:base64Binary"/>
</xs:schema>
```

SAML 2.0

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:wsse="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
targetNamespace="urn:oasis:names:tc:SAML:2.0:assertion" elementFormDefault="qualified">
 <xs:import namespace="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd" schemaLocation="wsse.xsd"/>
 <xs:import namespace="http://www.w3.org/2000/09/xmldsig#" schemaLocation="ds.xsd"/>
 <xs:element name="Assertion">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="saml:Issuer" maxOccurs="1" minOccurs="1"/>
    <xs:element ref="saml:Subject" maxOccurs="1" minOccurs="1"/>
    <xs:element ref="saml:Conditions" minOccurs="1"/>
    <xs:element ref="saml:AttributeStatement" minOccurs="1" maxOccurs="3"/>
    <xs:element ref="ds:Signature" minOccurs="0" maxOccurs="1"/>
   </xs:sequence>
   <xs:attribute name="IssueInstant" type="xs:dateTime" use="required"/>
   <xs:attribute name="Version" type="xs:decimal" use="required"/>
   <xs:attribute name="id" type="xs:NCName" use="required"/>
  </xs:complexType>
 </xs:element>
 <xs:element name="Issuer" type="xs:NCName"/>
 <xs:element name="Subject">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="saml:NameID"/>
    <xs:element ref="saml:SubjectConfirmation" minOccurs="0"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="NameID">
  <xs:complexType>
   <xs:simpleContent>
    <xs:extension base="xs:integer">
     <xs:attribute name="Format" type="saml:SubjectIdentifierType" use="required"/>
    </xs:extension>
   </xs:simpleContent>
  </xs:complexType>
 </xs:element>
 <xs:element name="SubjectConfirmation">
  <xs:complexType>
```

```
  <xs:sequence>
   <xs:element ref="saml:ConfirmationMethod"/>
   <xs:element ref="saml:SubjectConfirmationData"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>
<xs:element name="ConfirmationMethod">
 <xs:simpleType>
  <xs:restriction base="xs:anyURI">
   <xs:enumeration value="urn:oasis:names:tc:SAML:2.0:cm:holder-of-key"/>
  </xs:restriction>
 </xs:simpleType>
</xs:element>
<xs:element name="SubjectConfirmationData">
 <xs:complexType>
  <xs:choice>
   <xs:element ref="ds:KeyInfo"/>
   <xs:element ref="wsse:UsernameToken"/>
  </xs:choice>
 </xs:complexType>
</xs:element>
<xs:element name="Conditions">
 <xs:complexType>
  <xs:attribute name="NotBefore" type="xs:dateTime" use="required"/>
  <xs:attribute name="NotOnOrAfter" type="xs:dateTime" use="required"/>
 </xs:complexType>
</xs:element>
<xs:element name="AttributeStatement">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="saml:Attribute" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" use="required">
   <xs:simpleType>
    <xs:restriction base="xs:NCName">
     <xs:enumeration value="IDCardData"/>
     <xs:enumeration value="UserLog"/>
     <xs:enumeration value="SystemLog"/>
    </xs:restriction>
   </xs:simpleType>
  </xs:attribute>
 </xs:complexType>
</xs:element>
<xs:element name="Attribute">
 <xs:complexType>
  <xs:sequence>
   <xs:element ref="saml:AttributeValue"/>
  </xs:sequence>
  <xs:attribute name="Name" use="required">
   <xs:simpleType>
    <xs:restriction base="xs:NMTOKEN">
     <xs:enumeration value="sosi:IDCardID"/>
     <xs:enumeration value="sosi:IDCardVersion"/>
     <xs:enumeration value="sosi:IDCardType"/>
     <xs:enumeration value="sosi:AuthenticationLevel"/>
     <xs:enumeration value="sosi:OCESCertHash"/>
     <xs:enumeration value="medcom:UserCivilRegistrationNumber"/>
     <xs:enumeration value="medcom:UserGivenName"/>
     <xs:enumeration value="medcom:UserSurName"/>
     <xs:enumeration value="medcom:UserEmailAddress"/>
     <xs:enumeration value="medcom:UserRole"/>
     <xs:enumeration value="medcom:UserOccupation"/>
     <xs:enumeration value="medcom:UserAuthorizationCode"/>
     <xs:enumeration value="medcom:CareProviderID"/>
     <xs:enumeration value="medcom:CareProviderName"/>
     <xs:enumeration value="medcom:ITSystemName"/>
    </xs:restriction>
   </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="NameFormat" type="saml:SubjectIdentifierType"/>
 </xs:complexType>
</xs:element>
<xs:element name="AttributeValue" type="xs:string"/>
<xs:simpleType name="SubjectIdentifierType">
```

```
  <xs:restriction base="xs:anyURI">
   <xs:enumeration value="medcom:cprnumber"/>
   <xs:enumeration value="medcom:ynumber"/>
   <xs:enumeration value="medcom:pnumber"/>
   <xs:enumeration value="medcom:skscode"/>
   <xs:enumeration value="medcom:cvrnumber"/>
   <xs:enumeration value="medcom:communalnumber"/>
   <xs:enumeration value="medcom:locationnumber"/>
   <xs:enumeration value="medcom:other"/>
  </xs:restriction>
 </xs:simpleType>
</xs:schema>
```

MedCom Den Gode Webservice 1.0

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:medcom="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd"
targetNamespace="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd" elementFormDefault="qualified">
 <xs:element name="Linking">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="medcom:FlowID"/>
    <xs:element ref="medcom:MessageID" minOccurs="0"/>
    <xs:element ref="medcom:InResponseToMessageID" minOccurs="0"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="FlowID" type="xs:string"/>
 <xs:element name="MessageID" type="xs:string"/>
 <xs:element name="InResponseToMessageID" type="xs:string"/>
 <xs:element name="Priority">
  <xs:simpleType>
   <xs:restriction base="xs:string">
    <xs:enumeration value="AKUT"/>
    <xs:enumeration value="HASTER"/>
    <xs:enumeration value="RUTINE"/>
   </xs:restriction>
  </xs:simpleType>
 </xs:element>
 <xs:element name="RequireNonRepudiationReceipt">
  <xs:simpleType>
   <xs:restriction base="xs:string">
    <xs:enumeration value="yes"/>
    <xs:enumeration value="no"/>
   </xs:restriction>
  </xs:simpleType>
 </xs:element>
 <xs:element name="FlowStatus">
  <xs:simpleType>
   <xs:restriction base="xs:string">
    <xs:enumeration value="flow_running"/>
    <xs:enumeration value="flow_finalized_succesfully"/>
    <xs:enumeration value="syntax_error"/>
    <xs:enumeration value="missing_required_header"/>
    <xs:enumeration value="security_level_failed"/>
    <xs:enumeration value="invalid_username_password"/>
    <xs:enumeration value="invalid_signature"/>
    <xs:enumeration value="invalid_idcard"/>
    <xs:enumeration value="invalid_certificate"/>
    <xs:enumeration value="expired_idcard"/>
    <xs:enumeration value="not_authorized"/>
    <xs:enumeration value="illegal_http_method"/>
    <xs:enumeration value="processing_problem"/>
    <xs:enumeration value="signature_not_supported"/>
   </xs:restriction>
  </xs:simpleType>
 </xs:element>
 <xs:element name="SecurityLevel">
  <xs:simpleType>
   <xs:restriction base="xs:int">
    <xs:enumeration value="1"/>
    <xs:enumeration value="2"/>
```

```
      <xs:enumeration value="3"/>
      <xs:enumeration value="4"/>
      <xs:enumeration value="5"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
   <xs:element name="TimeOut">
    <xs:simpleType>
     <xs:restriction base="xs:string">
      <xs:enumeration value="5"/>
      <xs:enumeration value="30"/>
      <xs:enumeration value="480"/>
      <xs:enumeration value="1440"/>
      <xs:enumeration value="unbound"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:element>
   <xs:element name="Header">
    <xs:complexType>
     <xs:sequence>
      <xs:element ref="medcom:SecurityLevel" minOccurs="0"/>
      <xs:element ref="medcom:TimeOut" minOccurs="0"/>
      <xs:element ref="medcom:Linking"/>
      <xs:element ref="medcom:FlowStatus" minOccurs="0"/>
      <xs:element ref="medcom:Priority" minOccurs="0"/>
      <xs:element ref="medcom:RequireNonRepudiationReceipt" minOccurs="0"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
   <xs:element name="FaultCode" type="xs:string"/>
  </xs:schema>
```