



**Agència Catalana  
de Certificació**

# Guia bàsica per als integradors de PSIS

*Agència Catalana de Certificació (CATCert)*

Referència: D1220  
Versió: 16  
Data: 09/06/2008

---

## Informació general

### Control documental

<b>Projecte:</b>	Integració PSIS (Plataforma de Serveis de Identificació i Signatura)
<b>Entitat de destinació:</b>	Pública
<b>Títol:</b>	Guia bàsica per als integradors de PSIS
<b>Codi de referència:</b>	
<b>Versió:</b>	16
<b>Data:</b>	09/06/2008
<b>Arxiu:</b>	Guia bàsica pels integradors de PSIS_20080527.doc
<b>Eina/es d'edició:</b>	Microsoft Word 2003
<b>Autor/s:</b>	Garcia, David; Navalón, Ramón; Prats Ferrer, Pol; Marcos Cardona, Toni; Josep Riudavets; Joan Mir; Artur Barbeta; Sonia Martinez
<b>Resum:</b>	Descripció de l'ús de la plataforma de Serveis de Identificació i Signatura mitjançant el desenvolupament de clients del servei (WS). Exemples.
<b>Classificació informació – nivell d'accés:</b>	Pública

### Drets d'ús

La present documentació és propietat de l'AGÈNCIA CATALANA DE CERTIFICACIÓ, és confidencial i no podrà ésser objecte de reproducció total o parcial, tractament informàtic ni transmissió de cap forma o per qualsevol mitjà, ja sigui electrònic, mecànic, per fotocòpia, registre o qualsevol altre. Tanmateix, tampoc no podrà ésser objecte de préstec, lloguer o qualsevol forma de cessió d'ús sense el consentiment previ i per escrit de l'AGÈNCIA CATALANA DE CERTIFICACIÓ, titular del dret d'autor (*copyright*). L'incompliment de les limitacions assenyalades per qualsevol persona que tingui accés a la documentació serà perseguida d'acord amb la llei.

### Estat formal

Preparat per:	Revisat per:	Aprovat per:
Nom: Toni Marcos Cardona Data: 15/08/2006	Nom: D.Garcia, R.Navalón, JRiudavets, JMir, Prats Ferrer, Pol  Data: 170806/.. /030107/090608	Nom: Joan Mir Data: 09.02.2007

## Control de versions

Versió	Parts que canvien	Descripció del canvi	Data
1	Versió inicial del document.		03/07/2006
2	Reestructuració global de contingut	Reestructuració de continguts	2/08/2006
3	Segona reestructuració	Segon procés de refinament dels continguts.	14/08/2006
4	Tercera reestructuració	Canvis en els codis	
		Refinament i reestructuració dels apartats relatius a validacions de signatures.	17/08/2006
		Afegits referències a <i>Timestamp Profile</i> .	
		Canvis menors en els codis.	
5	Primera revisió	Revisió global dels continguts amb canvi d'estructuració per a suportar futures funcionalitats.	18/09/2006
		Afegit annex amb els esquemes DSS/XSS	
6	Validacions PDF	Afegit validació signatures a documents PDF	29/09/2006
7	Segona revisió	Refinament dels continguts.	19/10/2006
8	Tercera revisió	Refinament dels continguts	14/12/2006
8.1	Modificacions menors	Identificacions de versions	18/12/2006
9	Modificacions en VBasic	Millores explicatives	03/01/2007
10	Modificacions en Java	Completar exemples de validació de certificats	06/02/2007
10.1	Normalitzacions i correccions	Servidors referenciats normalitzats a psisbeta	09/02/2007
11.	Autenticació. Clients Java, .NET i VB.	Revisió global: millores, correcció errors.	08/05/2007
12	Tot el document	Millores explicatives	03/10/2007
13			
14	Tot el document	Millores explicatives	21/01/2008

16	Tot el document	Eliminació URLs de QUA	09/06/2008
----	-----------------	------------------------	------------

## Glossari

CA	<i>Certificate Authority</i>
CAdES	<i>CMS Advanced Electronic Signatures</i>
CMS	<i>Cryptographic Message Syntax</i>
DSS	<i>Digital Signature Services</i>
JDK	<i>Java Developer Kit</i>
JRE	<i>Java Runtime Environment</i>
OASIS	<i>Organization for the Advancement of Structured Information Standards</i>
PDF	<i>Portable Document Format</i>
PKCS7	<i>Public Key Cryptography Standards</i>
PKI	<i>Public Key Infrastructure</i>
PSIS	Plataforma de Serveis de Identificació i Signatura
RFC	<i>Request For Comments</i>
SOAP	<i>Simple Object Access Protocol</i>
SSL	<i>Secure Sockets Layer</i>
TLS	<i>Transport Layer Security</i>
TSA	<i>Transportation Safety Administration</i>
URI	<i>Uniform Resource Identifier</i>
URL	<i>Uniform Resource Locator</i>
UTF	<i>Universal Transformation Format</i>
VA	Autoritat de Validació
VB6	<i>Visual Basic 6</i>
XSS	<i>eXtended Signature Services (XSS) Profile of the OASIS Digital Signature Service (DSS)</i>
WSDL	<i>Web Service Definition Language</i>
XAdES	<i>XML Advanced Electronic Signatures</i>
XML	<i>Extensible Markup Language</i>
XMLDsig	<i>XML Digital Signatures</i>
XSD	<i>XML Schema Definition</i>

**Índex**

<b>Guia bàsica per als integradors de PSIS</b>	<b>1</b>
Agència Catalana de Certificació (CATCert)	1
<b>Informació general</b>	<b>2</b>
Control documental	2
Drets d'ús	2
Estat formal	2
Control de versions	3
Glossari	5
<b>Figures</b>	<b>8</b>
1. <b>Introducció</b>	<b>10</b>
2. <b>Arquitectura de PSIS</b>	<b>12</b>
3. <b>Prestació de servei</b>	<b>13</b>
4. <b>Missatgeria</b>	<b>14</b>
5. <b>Funcionalitats</b>	<b>22</b>
5.1 Validació de certificats	23
5.2 Validació de signatures en format PKCS#7 / CMS i XML	29
5.3 Validació de signatures XAdES	42
5.4 Validació de signatures PDF	50
5.5 Creació de segells de temps	52
5.6 Validació de segells de temps	59
5.7 Perfil de Timestamp	62
6. <b>Requisits previs</b>	<b>63</b>
6.1 Comunicacions	63
6.2 Autenticació	64
6.3 Software	73
6.4 WSDL	74
7. <b>Creació del client</b>	<b>75</b>
7.1 Java	75
7.2 .NET (C#)	82
7.3 Visual Basic 6	86
8. <b>Creació de la missatgeria</b>	<b>90</b>
8.1 Java	90
8.2 .NET (C#)	100
8.3 Visual Basic 6	110

---

<b>9. Annexes</b>	<b>118</b>
9.1 Referències	118
9.2 Codis de resposta genèrics	118
9.3 Codis de resposta de generació	119
9.4 Codis de resposta de validació	119
9.5 Atributs de consulta d'un certificat	121
9.6 Atributs de consulta d'una signatura	123
9.7 Esquema del protocol DSS i el seu perfil XSS	123

## Figures

Figura 1 Esquema d'invocació de clients a PSIS	10
Figura 2 Esquema d'invocació entre serveis de PSIS	12
Figura 3 Taula amb els diferents perfils de DSS	15
Figura 4 Missatge de validació d'un certificat X509	23
Figura 5 Missatge resposta d'una validació d'un certificat X509	27
Figura 6 Missatge de validació de signatura CMS attached	30
Figura 7 Missatge de validació de signatura CMS detached	30
Figura 8 Missatge de validació de signatura XML attached enveloping	31
Figura 9 Missatge de validació de signatura XML attached enveloped	33
Figura 10 Missatge de validació de signatura XML detached	33
Figura 11 Missatge de sortida per a una validació de signatura	38
Figura 12 Missatge de validació d'una signatura XAdES	44
Figura 13 Missatge de resposta a una validació d'una signatura XAdES	49
Figura 14 Missatge de validació d'un document PDF	50
Figura 15 Taula de compatibilitat entre formats de segells de temps i els continguts a estampar	53
Figura 16 Missatge de creació d'un segell de temps en format XML	54
Figura 17 Missatge de creació d'un segell de temps en format CMS	55
Figura 18 Missatge de resposta de creació d'un segell de temps	58
Figura 19 Missatge de validació d'un segell de temps en format XML	61
Figura 20 Missatge de validació d'un segell de temps en format CMS	61
Figura 21 Missatge de resposta d'una validació d'un segell de temps	62
Figura 22 Captura de pantalla amb una connexió correcta a la plataforma PSIS	64
Figura 23 Imatge de la MMC	67
Figura 24 Contingut del fitxer ant per generar el client Java de PSIS fent servir el fitxer WSDL	78
Figura 25 Contingut del fitxer ant per compilar el client Java generat a partir del fitxer WSDL	81
Figura 26 Exemple de creació de la connexió amb la plataforma PSIS en Java	82
Figura 27 Exemple de creació de la connexió amb la plataforma PSIS en .net	85
Figura 28 Exemple de creació de la connexió amb la plataforma PSIS en VB6	88
Figura 28b Exemple de creació de la connexió amb la plataforma PSIS en VB9	88
Figura 29 Exemple en Java de validació de certificats	91
Figura 30 Exemple en Java de validació de signatura CMS	93
Figura 31 Exemple en Java de validació de signatura XML	94

---

Figura 32 Exemple en Java de validació de signatura XAdES	95
Figura 33 Exemple en Java de validació de documents PDF signats	96
Figura 34 Exemple en Java de creació de segell de temps	98
Figura 35 Exemple en Java de validació de segell de temps	99
Figura 36 Exemple en .net de validació de certificats	101
Figura 37 Exemple en .net de validació de signatura CMS	103
Figura 38 Exemple en .net de validació de signatura XML	104
Figura 39 Exemple en .net de validació de signatura XAdES	105
Figura 40 Exemple en .net de validació de document PDF signat	106
Figura 41 Exemple en .net de creació de segell de temps	108
Figura 42 Exemple en .net de validació de segell de temps	109
Figura 43 Exemple en Visual Basic de validació de certificats	111
Figura 44 Exemple en Visual Basic de signatura CMS	112
Figura 45 Exemple en Visual Basic de signatura XML	113
Figura 46 Exemple en Visual Basic de signatura XAdES	114
Figura 47 Exemple en Visual Basic de document PDF signat	115
Figura 48 Exemple en Visual Basic de creació de segell de temps	116
Figura 49 Exemple en Visual Basic de creació de segell de temps	117

## 1. Introducció

L'objectiu d'aquest document és descriure la invocació de funcionalitats de PSIS fent servir el protocol DSS. Per a agilitzar i simplificar el procés d'integració de clients amb la plataforma PSIS, CATCert proporciona als seus clients un fitxer WSDL (*Web Services Definition Language*) que descriu la missatgeria DSS en un format estandarditzat.

La plataforma PSIS ofereix una sèrie de serveis d'alt nivell que permeten realitzar un gran nombre de funcionalitats relacionades amb la PKI, com ara la verificació de certificats, la creació i verificació de segells de temps i signatures digitals, així com l'arxivat de les mateixes, entre d'altres.

La plataforma PSIS ofereix la possibilitat de poder-se comunicar amb les diverses funcionalitats mitjançant diferents tipus de mecanismes

Aquests mecanismes de connexió als serveis ofertats per PSIS són els següents:

- Peticions codificades segons DSS (protocol per a serveis de signatura digital sobre *web services*), segons el protocol de l'autoritat de validació de CATCert (protocol propietari per a la validació de signatures i certificats en XML sobre *web services*)
- Protocols basats en missatgeria no XML com ara el protocol de creació de segells de temps o RFC 3161.

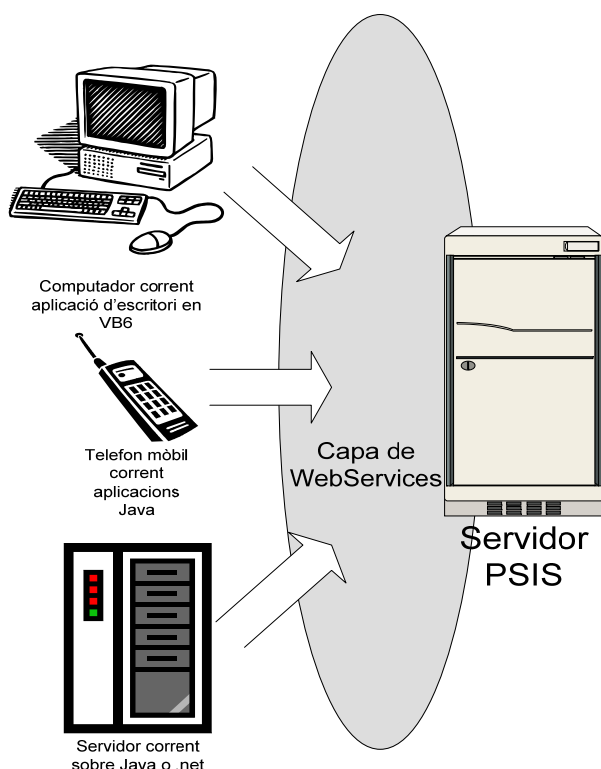


Figura 1 Esquema d'invocació de clients a PSIS

Per tant, aquest document està focalitzat en com els clients poden invocar qualsevol servei ofert per PSIS fent servir el protocol DSS. El document inclou tota la informació necessària per a poder desenvolupar el software (llibries, objectes i classes) que es pot necessitar per tal de poder fer ús dels serveis oferts per la plataforma PSIS.

El document està organitzat en un conjunt d'apartats on es pot trobar tota la informació necessària per a poder fer la integració amb la plataforma PSIS.

- En l'apartat 1, es dóna una visió global dels objectius que es volen assolir en quant a la integració amb els serveis ofertats per PSIS.
- L'arquitectura de la plataforma PSIS es descriu breument en l'apartat 2.
- Dins l'apartat 3 s'introdueix breument, i amb l'ajuda d'un esquema, el sistema de prestació del servei ofertat per la plataforma PSIS per a poder crear una connexió externa amb la mateixa.
- L'apartat 4, amb el suport d'un conjunt d'esquemes, descriu la missatgeria inclosa dins del protocol DSS, la qual permet tenir disponibles les comunicacions per a poder fer les operacions que es poden realitzar amb la plataforma PSIS.
- El conjunt de funcionalitats ofertes per la plataforma PSIS (validació de certificats digitals, validació de signatures digitals, validació de PDF's, validació i creació de segells de temps) es descriuen en l'apartat 5 documentant la missatgeria del protocol DSS que es necessita per tal de poder fer peticions a la plataforma PSIS i posteriorment fer el processament de la resposta i visualitzar els diferents *OptionalInputs* disponibles per a cada funcionalitat.
- Els requisits de connexió a la plataforma PSIS, el procés de creació dels mòduls necessaris per a la connexió, juntament amb exemples d'ús de cadascuna de les funcionalitats descrites en l'apartat anterior, es pot consultar en l'apartat 6. Tota aquesta informació es troba disponible en tres tecnologies diferents: Java, .NET(C#) i Visual Basic 6.
- Per a finalitzar, el document, a l'apartat 7, disposa d'un annex on s'inclouen referències, esquemes i informació que complementen i amplien els conceptes presentats en aquesta documentació.

**NOTA:** Per tal de poder desenvolupar el software necessari per a connectar la plataforma PSIS es fan servir tecnologies que només s'utilitzaran en algun procés descrit al present document. En aquest sentit, cal mencionar que l'objectiu d'aquest document no és el de documentar aquestes tecnologies. Per això, només es presentaran les dades necessàries per a poder fer cada procés, sense fer referència a punts tan diversos com poden ser: instal·lació (ex: Ant, Visual Basic 6, .NET), configuració (ex: configuració de nous projectes Visual Basic 6 amb referències a llibries externes), etc...

## 2. Arquitectura de PSIS

L'arquitectura de la plataforma PSIS va ésser creada fent servir un patró de components orientats a servei. Això vol dir que la plataforma està composta per multitud de serveis que s'orquestren i col·laboren entre sí per a poder donar servei, i que poden ser reutilitzats i recombinats per a donar diferents tipus de serveis.

Aquest model orientat a servei permet, tanmateix, disposar de diferents instàncies del mateix servei amb configuracions diferents. Aquesta *virtualització* del servei possibilita, per exemple, disposar de serveis, com ara diverses instàncies distingibles de serveis de validacions de certificats, que coexisteixen al mateix servidor; però que donat que disposen de configuracions diferents tenen comportaments totalment separats, tot i que el seu codi és idèntic.

Aquesta agrupació de components de grau gruixut (serveis) també permet que la composició dels mateixos no estigui lligada a codi, sinó que la relació *intra-servei* (com ara que un servei de validació de signatura X faci servir un servei de validació de certificats Y) és un paràmetre més a configurar, amb el què aquesta composició pot ésser alterada sense haver-ne de modificar codi.

Un altre avantatge d'aquest tipus d'arquitectura és que els serveis poden disposar de diferents tipus d'implementació (per exemple, un servei de signatura criptogràfica basat en llibreries *software* i un altre basat en dispositius *hardware*), sent cridats, però, per un únic servei que compleix un contracte (o interfície de servei). Això suposa un gran desacoblament entre serveis amb els següents avantatges:

- Coexistència de diferents versions de servei en el mateix servidor assignades a instàncies individuals de serveis.
- Existència de diferents implementacions per al mateix problema. Per exemple, solucions *software* vs. solucions *hardware*.

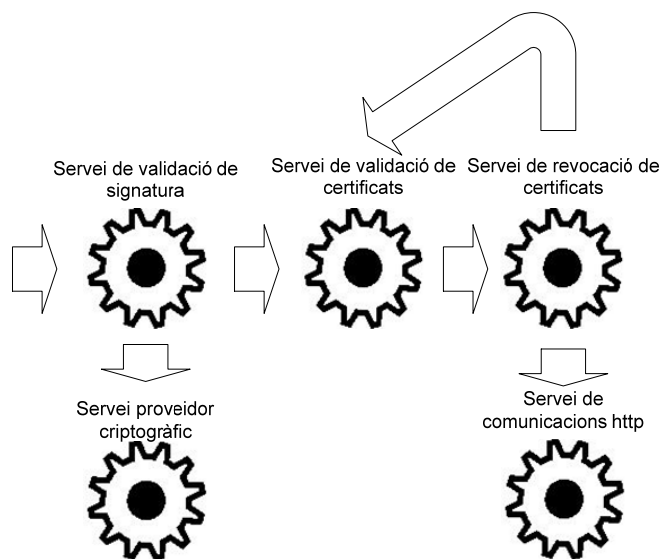


Figura 2 Esquema d'invocació entre serveis de PSIS

---

## 3. Prestació de servei

---

La plataforma PSIS presta els seus serveis en forma de serveis web (o WebServices) que són invocats mitjançant l'intercanvi de fitxers XML entre el client i el servidor que contenen les peticions de servei i les respostes del servidor a aquestes peticions.

Per tal de facilitar la creació de clients d'aquest servei la plataforma disposa d'un fitxer de definició WSDL (Web Services Definition Language) que descriu el servei, tant les estructures dels missatges intercanviats com les adreces dels propis serveis.

L'existència d'aquest fitxer permet que els clients puguin compilar-lo i generar automàticament el codi per a invocar PSIS. Per tal de poder dur a terme aquest procés, els clients només necessiten d'unes llibreries d'invocació de *web services* amb suport per a la invocació de serveis WEB fent servir *document/literal* (la majoria dels clients de *web services* incorporen aquest paradigma d'invocació). Els detalls del procés seran descrits posteriorment a l'apartat 6, ampliant la informació del procés necessari per a cadascuna de les tecnologies tractades.

D'aquesta manera, la complexitat de la lògica a desenvolupar pel client per tal de poder invocar els serveis queda reduïda a la construcció del missatge de petició, que conté les dades de la invocació, i a processar la resposta del servidor. En aquest cas, i donat que es fan servir clients de *web services*, els clients hauran de crear els *stubs* (o objectes generats a partir de la compilació del WSDL) corresponents a l'estructura del missatge i enviar-los (procés que el client del *web service* també encapsula).

Per a il·lustrar el funcionament, procedirem primer a especificar i detallar el format dels missatges a construir per tal de dur a terme les invocacions de servei (sintaxis dels XML's, adjuntant els seus *schemas* corresponents) i després, per a cada funcionalitat de la plataforma PSIS, s'inclouran exemples de com construir aquests missatges per a cada tecnologia concreta.

## 4. Missatgeria

Per a poder fer ús d'aquesta plataforma es requereix, com ja s'ha comentat, del desenvolupament d'uns clients que construïran missatges de petició i extrauran les respostes dels missatges provinents del servidor abstraient el client del procés d'invocació remota que es porta a terme.

Un dels protocols amb què treballa la plataforma és el DSS (*Digital Signature Services*), del consorci d'estandardització OASIS (*Organization for the Advancement of Structured Information Standards*) un protocol per a la prestació de serveis de signatura digital obert i extensible (mitjançant l'ús de perfils).

El protocol DSS Core conté una aproximació generalista als problemes derivats de la provisió de serveis de signatura electrònica. Els perfils de DSS són extensions del DSS Core que aporten més detalls i funcionalitats per a solucionar problemes més concrets.

De perfils es poden trobar diversos, però PSIS dóna suport principalment a XSS (desenvolupat per CATCert i que amplia DSS permetent, entre d'altres, la validació de certificats X509), el perfil XAdES (que permet l'actualització de signatures) o el *Timestamp Profile*, que aporta més control i detalls en l'àmbit dels segells de temps sobre DSS.

També hi ha altres perfils, com el d'arxivat o el de PDF, que són suportats per PSIS i que proporcionen funcionalitats complementàries a les definides en aquesta documentació, com ara, l'arxivat de llarga durada per a signatures i validació de signatures sobre documents PDF, entre d'altres.

Profiles	
<b>DSS</b> Protocol bàsic de creació i validació de signatures	urn:oasis:names:tc:dss:1.0:core:schema
<b>XADES</b> Ampliació de DSS que permet treballar amb signatures avançades XAdES i CAdES	urn:oasis:names:tc:dss:1.0:profiles:XAdES
<b>XSS</b> Ampliació de DSS que permet, entre d'altres validar certificats X509 de clau pública, extreure informació dels mateixos i fer servir polítiques de signatura	urn:oasis:names:tc:dss:1.0:profiles:XSS
<b>TIMESTAMP</b> Defineix restriccions extres sobre la creació i validació de segells de temps via DSS	urn:oasis:names:tc:dss:1.0:profiles:timestamping

<b>ARCHIVE</b> Permet l'arxivat de signatures electròniques	urn:oasis:names:tc:dss:1.0:profiles:archive
<b>COMPOUND</b> Permet l'enviament massiu de peticions DSS i la seva programació en moments diferents del temps	urn:oasis:names:tc:dss:1.0:profiles:compound
<b>DSS_PDF</b> Permet validar documents PDF amb signatures PKCS#7	urn:oasis:names:tc:dss:1.0:profiles:DSS_PDF

**Figura 3 Taula amb els diferents perfils de DSS**

La documentació detallada del protocol i els seus perfils està disponible en el paquet distribuït per CATCert. Els esquemes del protocol DSS i el seu perfil XSS estan inclosos també a l'annex del present document com a informació complementària.

**NOTA:** Totes les descripcions d'estructures / elements que formen part de la missatgeria DSS contenen el nom del document on es pot trobar el detall de la descripció, juntament amb possibles comentaris particulars de la plataforma PSIS.

La missatgeria que intervé a la plataforma PSIS ve definida per l'estàndard DSS i funciona sota el protocol SOAP (*Simple Object Access Protocol*).

SOAP és un protocol estàndard sobre el qual es fonamenta la tecnologia de serveis web (*Web Services*). A diferència d'altres protocols de tipus binari com poden ser COM, COM+ o DCOM, els quals són propis de Microsoft, SOAP es basa en documents de text pla codificats en format XML. L'avantatge principal de codificar en XML és que els missatges són llegibles per éssers humans; però, per contra, aquests documents resultants són, en general, de tamany gran.

SOAP està dissenyat per a funcionar sobre qualsevol protocol d'internet, tot i que l'ús més habitual és sobre HTTP. El fet d'utilitzar HTTP minimitza l'impacte de dispositius com Firewalls i similars, i fa accessible SOAP a pràcticament qualsevol tipologia de comunicació client-servidor.

Els missatges SOAP estan compostats per dos grans blocs funcionals: "Capçalera" (*envelope*) destinat a subministrar dades d'enrutament i "Cos" (*body*), el qual conté les dades del missatge d'usuari. Una explicació més detallada de SOAP no forma part de l'abast d'aquest document.

En aquest apartat es tracten els aspectes més importants involucrats en l'ús del protocol DSS, tot i això, s'adjunta la referència on es pot consultar el document de l'estàndard corresponent per si fos necessària més informació sobre l'apartat concret.

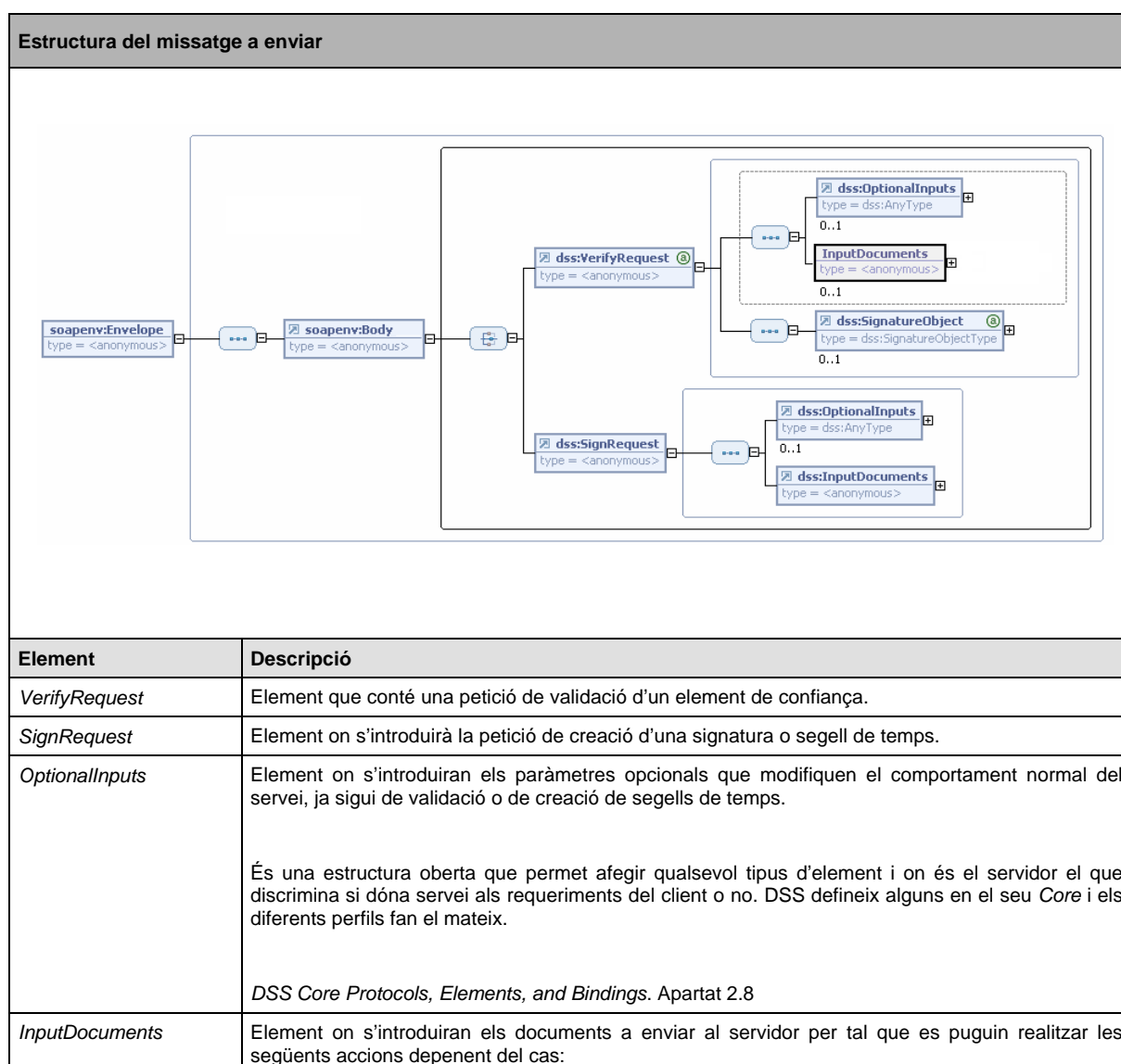
A més, es defineixen una sèrie de prefixos per als diferents espais de noms involucrats. El seu mapeig contra les uri's dels espais de noms és el següent:

- xd: <http://www.w3.org/2000/09/XMLDsig#>
- dss : urn:OASIS:names:tc:dss:1.0:core:schema
- xss : urn:OASIS:names:tc:dss:1.0:profiles:XSS
- pdf: urn:OASIS:names:tc:dss:1.0:profiles:DSS\_PDF

Aquí mostrem els dos tipus bàsics d'estructures que s'utilitzen al protocol DSS, juntament amb els elements bàsics que es faran servir per a compondre els missatges.

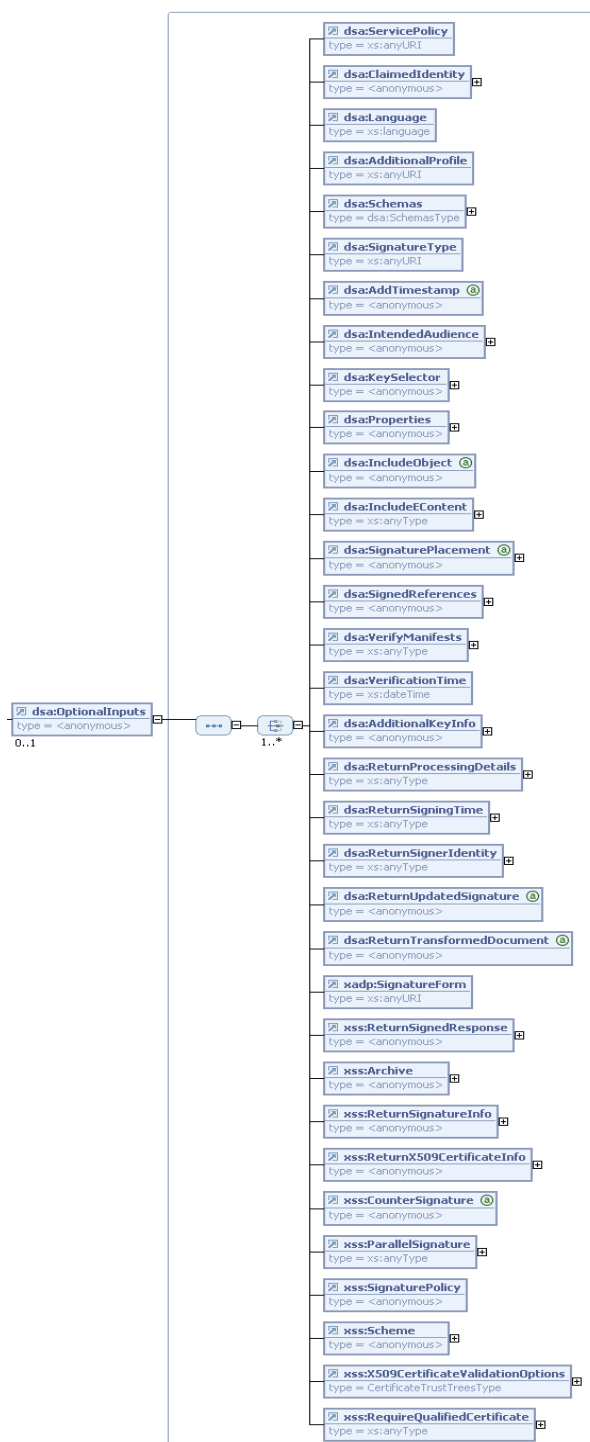
Els dos tipus principals de missatges que defineix DSS són *VerifyRequest* (per a peticions de validació) i *SignRequest* (per a peticions de signatura o estampació de segell de temps).

L'estructura bàsica dels missatges utilitzats per a fer peticions a la plataforma és la següent:



	<ul style="list-style-type: none"> <li>• Documents a signar</li> <li>• Signatures a verificar (quan la signatura estigui continguda dintre del document proporcionat, com ara el cas de les signatures <i>XML Enveloped</i>)</li> <li>• Documents signats (quan la signatura és del tipus <i>detached</i>, es a dir, que va per separat de la signatura)</li> </ul> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 2.4</i></p>
<i>SignatureObject</i>	<p>Element on s'introduiran les signatures digitals a verificar. Aquestes poden ser del tipus:</p> <ul style="list-style-type: none"> <li>• XMLDsig</li> <li>• XAdES</li> <li>• PKCS#7 / CMS</li> <li>• CadES</li> </ul> <p>Altres dades que es poden verificar fent servir aquest element, són:</p> <ul style="list-style-type: none"> <li>• Certificats digitals</li> <li>• Segells de temps</li> </ul> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 2.5</i></p>

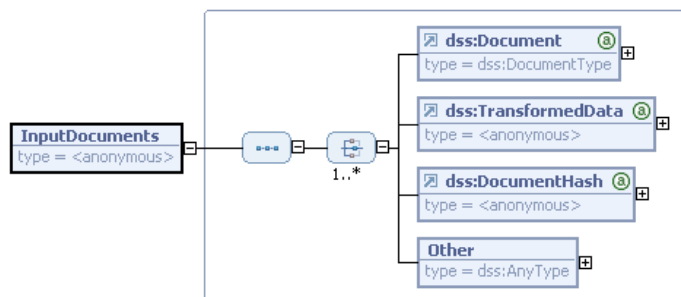
## Estructura d' *OptionalInputs*



Element	Descripció
<i>OptionalInputs</i>	Element que conté el conjunt de <i>OptionalInputs</i> que permetrà configurar l'execució de l'operació al servidor.

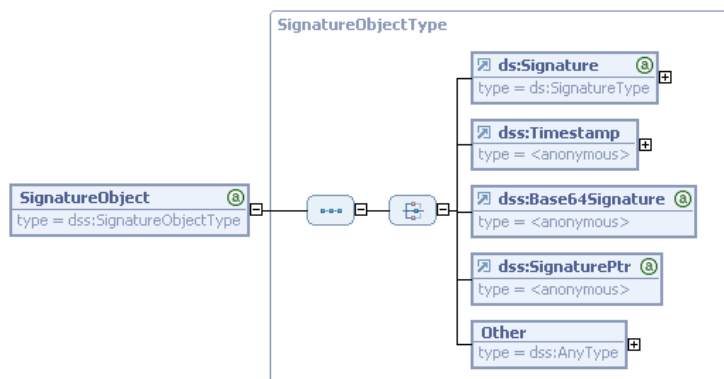
	<p>Alguns dels <i>OptionalInputs</i> visibles a l'esquema adjuntat, s'exposen en aquest mateix document a l'apartat 5, separats per funcionalitat.</p> <p><i>DSS Core Protocols, Elements, and Bindings</i>. Apartat 2.4</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### Estructura d'InputDocuments



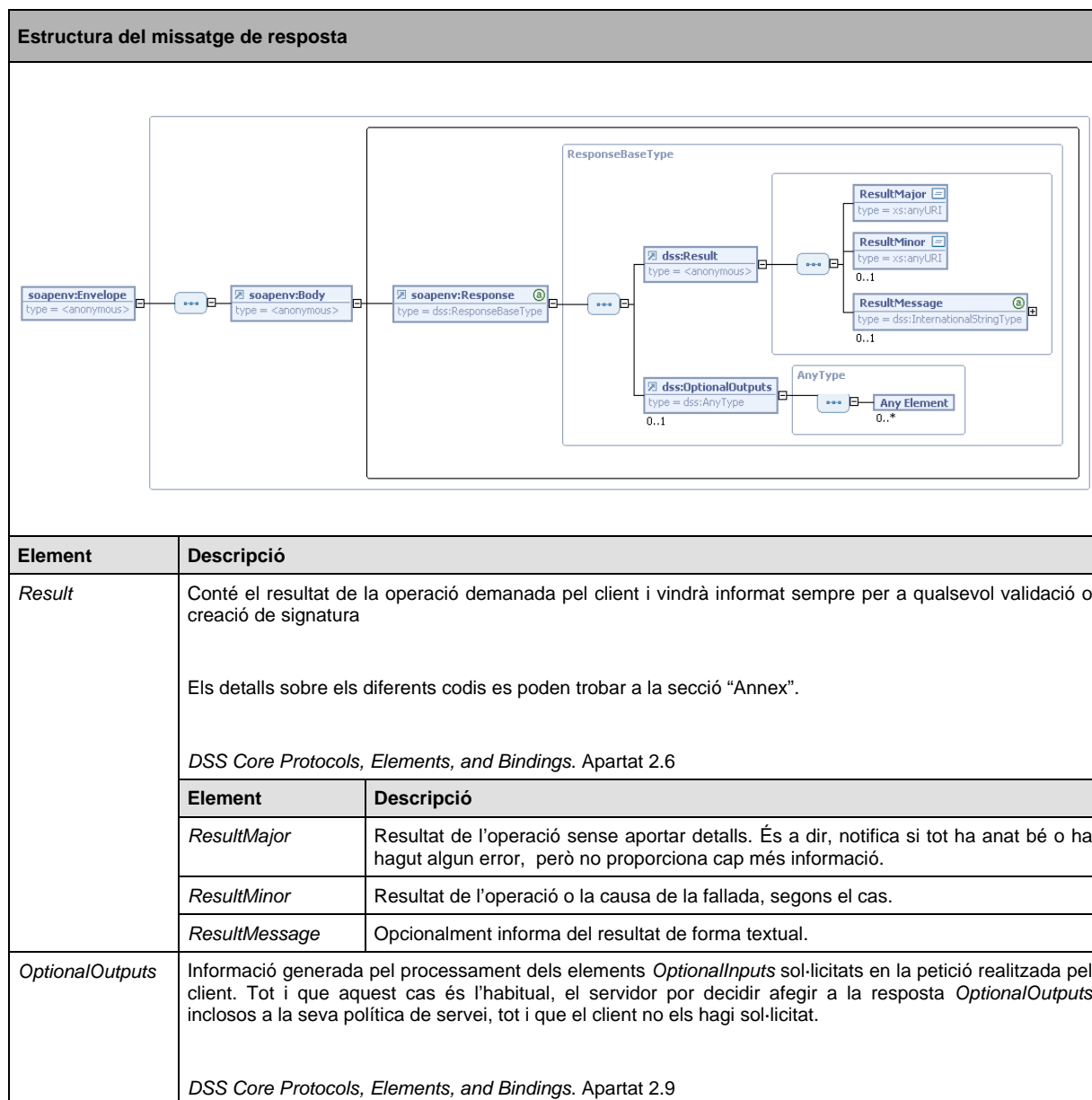
Element	Descripció
<i>Document</i>	<p>Aquest element contindrà un document a validar.</p> <p>Aquest document pot portar la signatura dins seu (signatura <i>attached enveloped</i>), o portar la signatura en el mateix missatge per separat (signatura <i>detached</i>) per a casos de validacions, o bé anar sol en cas d'estampació de segells de temps o creació de signatura.</p> <p><i>DSS Core Protocols, Elements, and Bindings</i>. Apartat 2.4</p>
<i>TransformedData</i>	<p>Aquest element conté un document sobre el qual s'ha efectuat algun tipus de transformació pel client abans d'enviar-lo al servidor. Aquestes transformacions poden ser qualsevol de les estàndards definides com ara <i>c14n</i>, <i>c14nexcl</i>, <i>base64</i>... o bé alguna definida <i>adhoc</i>, però suportada pel servidor.</p> <p><i>DSS Core Protocols, Elements, and Bindings</i>. Apartat 2.4</p>
<i>DocumentHash</i>	<p>En aquest tipus d'elements trobem dades a les quals el client ja ha aplicat un resum criptogràfic amb la finalitat que el document no viatgi al servidor, ja sigui per causes de mida o privacitat del mateix.</p> <p><i>DSS Core Protocols, Elements, and Bindings</i>. Apartat 2.4</p>
<i>Other</i>	<p>Permet ampliar els tipus de documents suportats sense haver d'alterar el protocol. En cas de que un altre profile necessiti informació diferent, es pot utilitzar aquest element per posar "Altres" coses de forma genèrica, així tenim la llibertat de posar noves informacions sense tenir que alterar la definició del protocol.</p>

### Estructura del *SignatureObject*



Element	Descripció
<i>Signature</i>	<p>Element que conté una signatura en format XMLDsig a validar.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 2.5</i></p>
<i>Timestamp</i>	<p>Element que conté un segell de temps a validar, ja sigui XML o CMS.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 5.1</i></p>
<i>Base64Signature</i>	<p>Element que conté una signatura CMS /PKCS#7 codificada en <i>base64</i>.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 2.5</i></p>
<i>SignaturePtr</i>	<p>Aquest element és un apuntador a una signatura XML que es troba inclosa dins d'un document dels indicats als <i>InputDocuments</i>. Es tracta d'una expressió XPath que apunta al document concret i a la signatura (o signatures) a validar.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 2.5</i></p>
<i>Other</i>	<p>Element on s'introduiran altres formes d'introduir una firma per futures aplicacions.</p> <p>Emprat per exemple per a enviar certificats a validar dins del protocol XSS.</p>

I les respostes de la plataforma PSIS, tenen la següent estructura:



## 5. Funcionalitats

De totes les funcionalitats que permet la plataforma PSIS, aquest document es centra en les següents:

- Validació de certificats
- Validació de signatures en format PKCS#7 / CMS i XML
- Validació signatures XAdES
- Validació de PDF
- Creació i validació de segells de temps

Es descriu la missatgeria que intervé a les comunicacions entre els clients i el servidor de la plataforma PSIS per tal d'executar les funcionalitats anomenades.

Per a cada funcionalitat es detalla el missatge a enviar des del client (missatge d'entrada) i el missatge que generarà la plataforma PSIS un cop processada la sol·licitud (missatge de sortida). Per a cada missatge s'inclouran breus explicacions dels paràmetres i elements que es fan servir en cada cas, a més d'incloure un detall amb els paràmetres opcionals que es poden afegir.

Adicionalment s'inclou el procés de desenvolupament amb els llenguatges de programació Java, .NET (C#) i Visual Basic 6, per a poder arribar a construir el missatge d'entrada que s'acabarà enviant a la plataforma PSIS.

**NOTA:** Els exemples que s'han documentat únicament fan una funció de suport de les explicacions donades per cada funcionalitat i les seves dades no són útils ja que s'han formatjat per tal de ser més aclaridores.

**NOTA:** Totes les descripcions d'estructures / elements que formen part de la missatgeria DSS contenen el nom del document a on es pot trobar el detall de la descripció juntament amb possibles comentaris particulars de la plataforma PSIS.

## 5.1 Validació de certificats

Aquesta funcionalitat permet fer la validació de certificats X509, així com l'extracció d'informació dels certificats fent servir funcionalitats definides al perfil XSS.

En el següent exemple, es descriu la missatgeria necessària per a poder sol·licitar al servidor la validació d'un certificat, així com d'altres funcionalitats addicionals a la mateixa:

### Missatge d'entrada

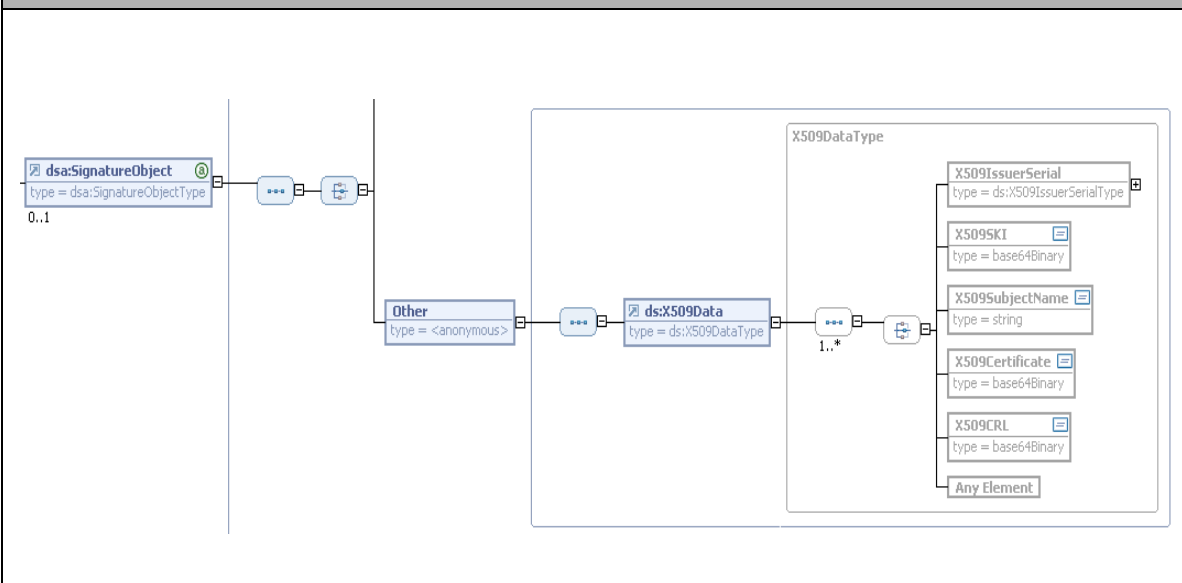
```
Validació de certificat X509

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <dss:VerifyRequest Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
      xmlns:xd="http://www.w3.org/2000/09/xmldsig#"
      xmlns:xss="urn:oasis:names:tc:dss:1.0:profiles:XSS">
      <dss:OptionalInputs>
        <dss:ReturnProcessingDetails/>
        <xss:ReturnX509CertificateInfo>
          <xss:AttributeDesignator
            Name="urn:oasis:names:tc:dss:1.0:profiles:XSS:certificateAttributes:Version"/>
          <xss:AttributeDesignator
            Name="urn:oasis:names:tc:dss:1.0:profiles:XSS:certificateAttributes:SerialNumber"/>
        </xss:ReturnX509CertificateInfo>
      </dss:OptionalInputs>
      <dss:SignatureObject>
        <dss:Other>
          <xd:X509Data>
            <xd:X509Certificate>
              MIIHxTCCBq2gAwIBAgIQOiZLlU8OHRFCG0+XhWb/ ...
              72e19BjFD6ELTFuO18J0qjwM/m8Z1vGnkNvgN2paGWE3WALgZdhQDh6dWb2IYvECbMw6qjJAigi3Ii7G1hsX66Ox0Y
              28TCBWGxkAxlhwsMYv01At2YHlSXlYpxv1cnVI+a3dLECaRRER1Q8C16YuPGaA0CdryPlCCZkBKwAhMOuPFdxhV/Hj
              9bLyUjgUQ=</xd:X509Certificate>
            </xd:X509Data>
          </dss:Other>
        </dss:SignatureObject>
      </dss:VerifyRequest>
    </soapenv:Body>
  </soapenv:Envelope>
```

Figura 4 Missatge de validació d'un certificat X509

El missatge d'entrada segueix l'estructura bàsica plantejada per l'estàndard DSS aportant una sèrie de *OptionalInputs* (que comentem amb detall posteriorment) dintre d'una *VerifyRequest*. El camp *SignatureObject* conté, en aquest cas, el certificat d'entitat final X509 a validar.

### Estructura de *SignatureObject*



Element	Descripció
<i>X509Certificate</i>	<p>Aquest tipus de petició conté el certificat a validar codificat en <i>base64</i> com a fill de l'element <i>SignatureObject</i> dintre d'un element <i>X509Data</i> definit per XMLDsig.</p> <p>Cal remarcar un cop més que es tracta d'una funcionalitat pròpia de XSS i no està present al Core de DSS.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 2.5</i></p>

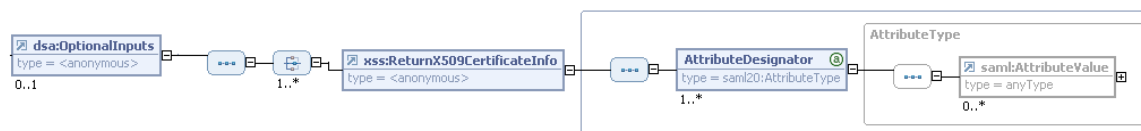
Dintre d'aquest tipus de petició, el servidor permet realitzar operacions addicionals relacionades amb la pròpia validació i que s'invoquen mitjançant la inclusió d'una sèrie d'*OptionalInputs* que ara es detallen.

### Estructura de *ReturnProcessingDetails*



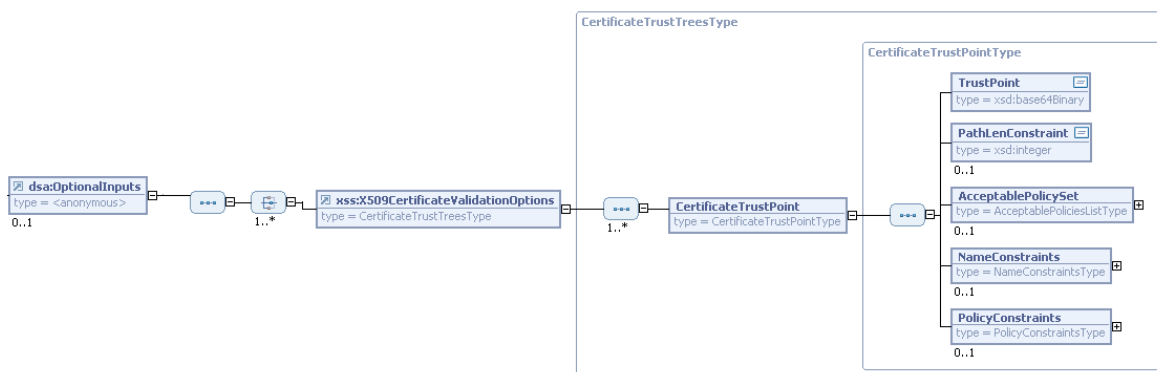
Element	Descripció
<i>ReturnProcessingDetails</i>	<p>Sol·licitud de consulta per part de client d'informació detallada sobre el procés de validació. Així, l'usuari demana al servidor que doni detalls dels diferents passos i causes que s'han dut a terme per tal de poder donar el resultat retornat.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.4</i></p>

## Estructura de *ReturnX509CertificateInfo*



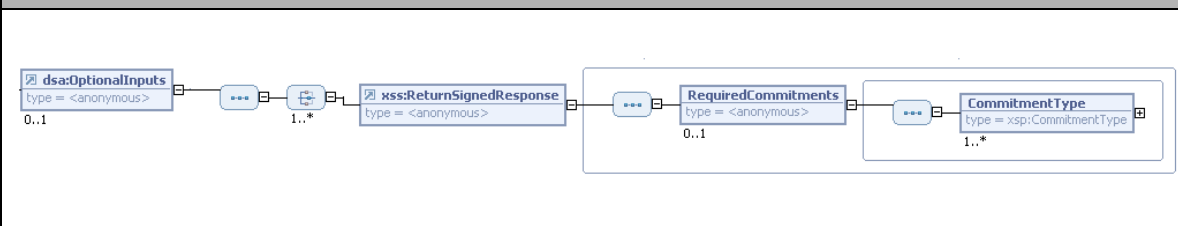
Element	Descripció
<i>ReturnX509CertificateInfo</i>	<p>Sol·licitud de consulta d'un atribut del certificat. Permet extreure informació del certificat proporcionat pel client.</p> <p>Es poden consultar 1 o N atributs en una mateixa consulta. En els annexes hi ha informació de tots els atributs de consulta disponibles.</p> <p><i>XSS Profile of the OASIS DSS. Apartat 3.1.5</i></p>

## Estructura de *X509CertificateValidationOptions*



Element	Descripció
<i>CertificateTrustPoint</i>	<p>Permet configurar amb molta precisió detalls de com el servidor realitzarà la validació del certificat. D'aquesta manera, es pot configurar les arrels de confiança per a la validació, la longitud del path de certificació i les restriccions de noms i polítiques de les CA's involucrades en el procés de validació.</p> <p><i>XSS Profile of the OASIS DSS. Apartat 5.1.4</i></p>

### Estructura de *ReturnSignedResponse*



Element	Descripció
<i>ReturnSignedResponse</i>	Demana al servidor que retorni la resposta signada.  XSS Profile of the OASIS DSS. Apartat 3.1.2

## Missatge de sortida

### Reposta de la validació de certificat X509

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <dss:VerifyResponse Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:Result>
        <dss:ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</dss:ResultMajor>
        <dss:ResultMinor>urn:oasis:names:tc:dss:1.0:profiles:XSS:resultminor:valid:certific
ate:Definitive</dss:ResultMinor>
      </dss:Result>
      <dss:OptionalOutputs>
        <dss:ProcessingDetails>
          <dss:ValidDetail
Type="urn:oasis:names:tc:dss:1.0:detail:ValidityInterval">
            <dss:Message xml:lang="en">The signing key is
inside its static validity
interval.</dss:Message>
          </dss:ValidDetail>
          <dss:ValidDetail
Type="urn:oasis:names:tc:dss:1.0:detail:IssuerTrust">
            <dss:Message xml:lang="en">The issuer of the
given key is
trusted.</dss:Message>
          </dss:ValidDetail>
          <dss:ValidDetail
Type="urn:oasis:names:tc:dss:1.0:detail:RevocationStatus">
            <dss:Message xml:lang="en">The signing key is
not revoked.</dss:Message>
          </dss:ValidDetail>
        </dss:ProcessingDetails>
        <urn:X509CertificateInfo
xmlns:urn="urn:oasis:names:tc:dss:1.0:profiles:XSS">
          <urn:Attribute
Name="urn:oasis:names:tc:dss:1.0:profiles:XSS:certificateAttributes:Version">
            <urn1:AttributeValue
xmlns:urn1="urn:oasis:names:tc:SAML:2.0:assertion"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:integer">3</urn1:AttributeValue>
          </urn:Attribute>
          <urn:Attribute
Name="urn:oasis:names:tc:dss:1.0:profiles:XSS:certificateAttributes:SerialNumber">
            <urn1:AttributeValue
xmlns:urn1="urn:oasis:names:tc:SAML:2.0:assertion"
xmlns:xs="http://www.w3.org/2001/XMLSchema"

```

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="xs:integer"
>77294064046332314987328417165725401088</urn1:AttributeValue>
</urn:Attribute>
</urn:X509CertificateInfo>
</dss:OptionalOutputs>
</dss:VerifyResponse>
</soapenv:Body>
</soapenv:Envelope>

```

Figura 5 Missatge resposta d'una validació d'un certificat X509

El missatge de sortida segueix l'estructura bàsica plantejada per l'estàndard DSS.

Els elements continguts a la resposta de la validació de certificat són:

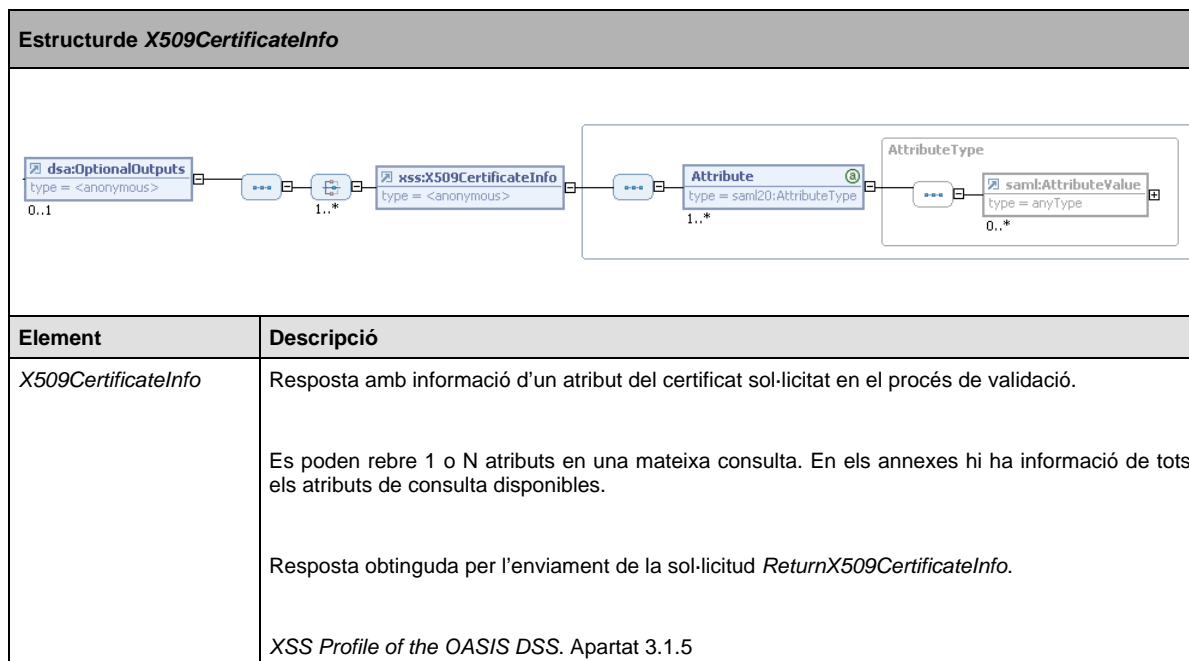
- *Result*

Estructura de dades amb el resultat del procés de validació. Com segueix el format DSS conté un major i un minor. Ambdós es troben detallats a l'annex corresponent i a la secció relativa als resultats de validació de certificats de XSS.

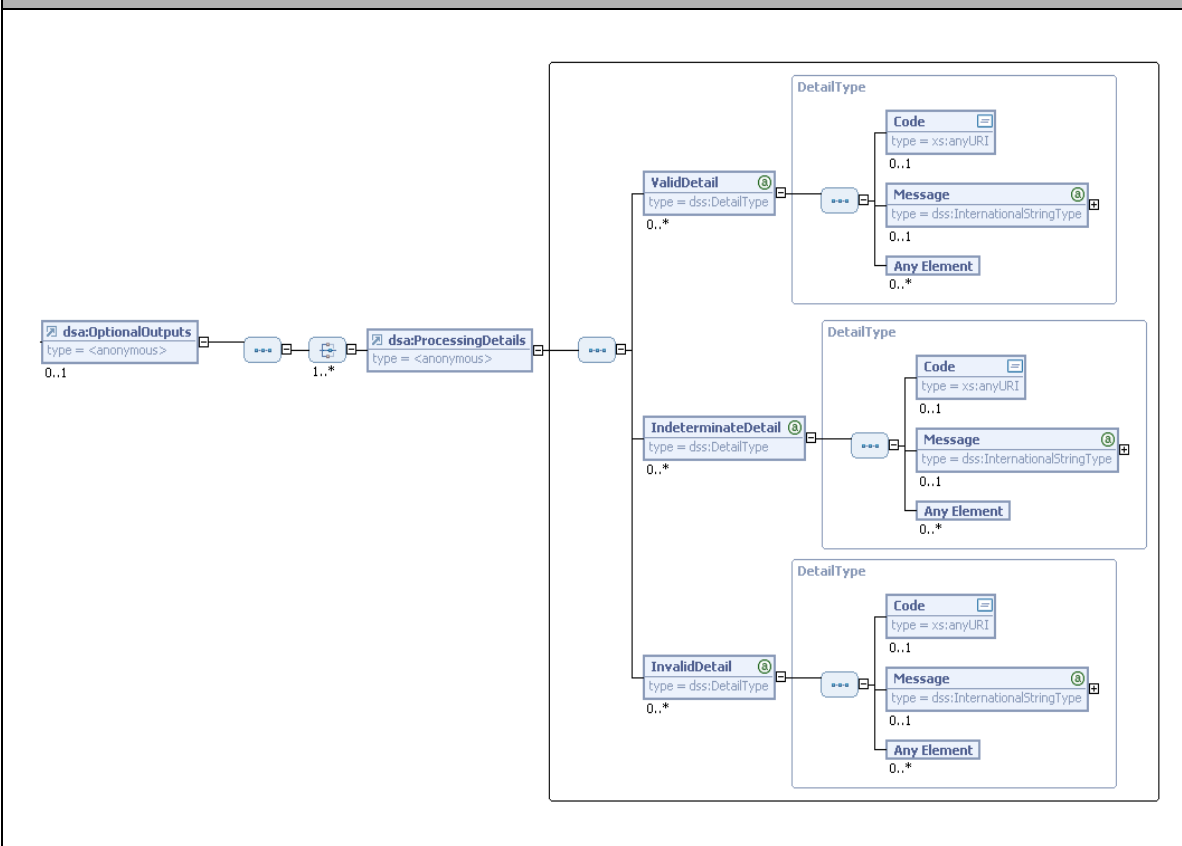
- *OptionalOutputs*

Estructura de dades que contindrà la informació sol·licitada pel client amb els elements introduïts dins de l'estructura <OptionalInputs>.

Es podran rebre els elements:

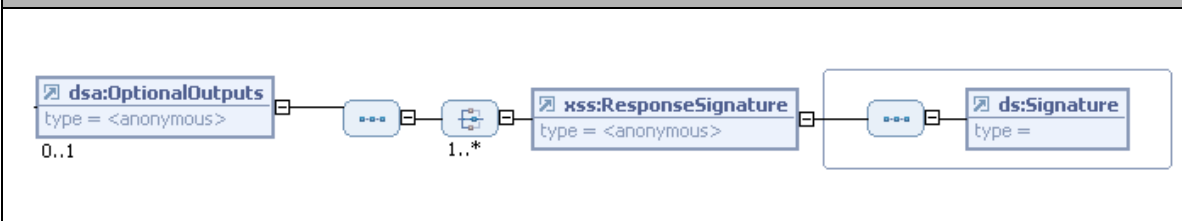


### Estructura de *ProcessingDetails*



Element	Descripció
<i>ProcessingDetails</i>	<p>Informació detallada del procés de validació.</p> <p>Resposta obtinguda per l'enviament de la sol·licitud <i>ReturnProcessingDetails</i>.</p> <p><i>DSS Core Protocols, Elements and Bindings. Apartat 4.6.4</i></p>

### Estructura de *ResponseSignature*



Element	Descripció
<i>ResponseSignature</i>	<p>Retorna una signatura sobre el missatge de resposta. En el cas que ens ocupa és una signatura XAdES-C sobre la <i>VerifyResponse</i>.</p> <p><i>XSS Profile of the OASIS DSS. Apartat 3.1.2</i></p>

## 5.2 Validació de signatures en format PKCS#7 / CMS i XML

Aquesta funcionalitat permet la validació de signatures simples, aquelles que no són enteses en terminologia PKI com a signatures complexes.

Dins el que considerem com a signatures simples, es pot fer una primera classificació en signatures PKCS#7 / CMS (*Cryptographic Message Syntax*) i signatures XML o XML DSig.

CMS és una ampliació del format PKCS#7, per la qual cosa quan es faci referència a PKCS#7 podem assimilar immediatament format CMS.

Ambdós sistemes defineixen dues modalitats per a signatures: *Detached* i *Attached*.

El format *Detached* significa que el contingut signat no és present dins el missatge de signatura i ha de ser transmès per altres vies. Així doncs, és el mètode més simple, donat que la signatura únicament conté el resum criptogràfic (*digest*) del contingut signat i no el contingut íntegre.

El format *Attached* implica que el contingut signat es troba dins el missatge de signatura. Ara bé, en el cas XMLDSig trobem dues modalitats: *Enveloping* i *Enveloped*. En el primer cas, les dades signades es troben com a referència dins la signatura, mentre que en el segon cas la signatura forma part del missatge (com a nus XML), la qual cosa implica que els algorismes de càlcul de signatura han d'ignorar el valor del camp de signatura en els càlculs.

Així doncs, els clients hauran de construir diferents missatges depenent de la topologia de la signatura que vulguin validar. Els detalls de les diferents topologies, així com els elements DSS que es fan servir en els mateixos, estan detallats a DSS, realitzant-se en aquest document una breu menció aclarativa.

En els següents exemples es descriu la missatgeria:

### Missatge d'entrada

Missatge per a la validació d'una signatura **CMS attached**. En aquest cas, no existeix cap document, donat que el document signat va dintre de la signatura i només proveïm d'un *SignatureObject* que conté una signatura binària codificada en *Base64*.

Validació de signatura CMS attached

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <!-- DSS message validating a correct CMS -->
    <!-- Result: Valid -->
    <dss:VerifyRequest xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:OptionalInputs>
        <dss:ReturnProcessingDetails/>
      </dss:OptionalInputs>
      <dss:SignatureObject>
        <dss:Base64Signature Type="urn:ietf:rfc:3369">
          .../66L1XkDrd8lNk7OFR6jdu5YL2g1oUQXBBRtCohbH5kTAS25CtBYFHYfN/Md6gzkdwXl+54gGHYH/mHq
          b8My+nAH/oOfbINBncnG0i5RfsvBuLymrh1YnUiEo01zd5VNSgC1QBfH6k3BOM5YC69znLmD0a8XsY3etywaD8ylUA
          AAAAAA</dss:Base64Signature>
        </dss:SignatureObject>
      </dss:VerifyRequest>
    </soapenv:Body>
  </soapenv:Envelope>
```

Figura 6 Missatge de validació de signatura CMS attached

Missatge per a la validació d'una signatura **CMS detached**. En aquest supòsit, existeix un document i proveïm un *SignatureObject* que conté una signatura binària codificada en *Base64*. El document adjuntat en aquest cas pot ser el document en sí codificat en *Base64* o bé el seu resum criptogràfic, tal i com disposa DSS.

Validació de signatura CMS detached

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <!-- DSS message validating a correct CMS -->
    <!-- Result: Valid -->
    <dss:VerifyRequest xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:OptionalInputs>
        <dss:ReturnProcessingDetails/>
      </dss:OptionalInputs>
      <dss:InputDocuments>
        <dss:Document>
          <dss:Base64Data>YWFhYWFhYWFh</dss:Base64Data>
        </dss:Document>
      </dss:InputDocuments>
      <dss:SignatureObject>
        <dss:Base64Signature Type="urn:ietf:rfc:3369">
          .../66L1XkDrd8lNk7OFR6jdu5YL2g1oUQXBBRtCohbH5kTAS25CtBYFHYfN/Md6gzkdwXl+54gGHYH/mHq
          b8My+nAH/oOfbINBncnG0i5RfsvBuLymrh1YnUiEo01zd5VNSgC1QBfH6k3BOM5YC69znLmD0a8XsY3etywaD8ylUA
          AAAAAA</dss:Base64Signature>
        </dss:SignatureObject>
      </dss:VerifyRequest>
    </soapenv:Body>
  </soapenv:Envelope>
```

Figura 7 Missatge de validació de signatura CMS detached

Missatge per a la validació d'una signatura **XML attached enveloping**, on com es pot veure el contingut a signar ha estat inclòs en el node amb *Id="Object"* de la mateixa. La signatura es proveeix dintre del *SignatureObject* com a una signatura XMLDsig.

Validació de signatura XML attached enveloping

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <dss:VerifyRequest Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:OptionalInputs>
        <dss:ReturnProcessingDetails/>
      </dss:OptionalInputs>
      <dss:SignatureObject>
        <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
          <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="#Object">
              <ds:Transforms>
                <ds:Transform
                  Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/></ds:Transform>
                <ds:DigestMethod
                  Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/></ds:DigestMethod>
                <ds:DigestValue>
                  8sEMF3ipDgNdjPoShP51Hbgly4k </ds:DigestValue>
              </ds:Reference>
            </ds:SignedInfo>
            <ds:SignatureValue> a/kmv5..... </ds:SignatureValue>
            <ds:KeyInfo>
              <ds:KeyValue>
                <ds:RSAKeyValue>
                  <ds:Modulus> uY17h.....
                  <ds:Exponent>AQAB</ds:Exponent>
                </ds:RSAKeyValue>
              </ds:KeyValue>
              <ds:X509Data>
                <ds:X509Certificate> MIIH...
              </ds:X509Data>
            </ds:KeyInfo>
            <ds:Object Id="Object">
              <a><b>.....</b></a>
            </ds:Object>
          </ds:Signature>
        </dss:SignatureObject>
      </dss:VerifyRequest>
    </soapenv:Body>
  </soapenv:Envelope>
```

Figura 8 Missatge de validació de signatura XML attached enveloping

Missatge per a la validació d'una signatura **XML attached enveloped**, on la signatura està continguda dintre del document adjunt.

Adicionalment, es pot proveir dintre del *SignatureObject* apuntant a la signatura dintre del document fent servir una expressió XPath com fem en aquest exemple. Per a més detalls sobre aquest tema, es pot consultar l'apartat sobre el **SignaturePrt** del document *DSS Core*.

## Validació de signatura XML attached enveloped

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"><soapenv:Body>
  <urn:VerifyRequest
    Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
    xmlns:xss="urn:oasis:names:tc:dss:1.0:profiles:XSS"
    xmlns:urn="urn:oasis:names:tc:dss:1.0:core:schema"
    xmlns:xsp="http://uri.etsi.org/2038/v1.1.1#"
    xmlns:xd="http://www.w3.org/2000/09/xmldsig#"
    xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
    <urn:OptionalInputs>
      <urn:ReturnProcessingDetails/>
    </urn:OptionalInputs>
    <urn:InputDocuments>
      <urn:Document ID="doc" RefURI="">
        <urn:InlineXML>
          <a><b><ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="id-1795ea06-d2ec-4dd0-8b67-05f9e47fa6de">
            <ds:SignedInfo>
              <ds:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"></ds:CanonicalizationMethod>
              <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"></ds:SignatureMethod>
              <ds:Reference URI="">
                <ds:Transforms>
                  <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"></ds:Transform>
                </ds:Transforms>
                <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"></ds:DigestMethod>
                <ds:DigestValue>630+0/BNGDxKudkxe5SUmEjFRmE=</ds:DigestValue>
              </ds:Reference>
              <ds:Reference Type="http://uri.etsi.org/01903/v1.2.2#SignedProperties" URI="#id-92759a28-aff4-4567-96f2-9d534d156592">
                <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"></ds:DigestMethod>
                <ds:DigestValue>QTpnCN3iXDOXa6pw6jyOALtgFYg=</ds:DigestValue>
              </ds:Reference>
            </ds:SignedInfo>
            <ds:SignatureValue>
              TnUs37pGeU43rONyqW9LgtEcmLvDH//ALUpCJpVlDyEy5uFRgoMV5UK6xg99dNCYEvK9Rf3Ho9R0
              017prbL6xwWsPnW7i01GPDkxWGFSS2J6+hdQLRVoDulIluX5Lw/09Fb0LtyUVhdAyBFjaWWzHTvV
              MQQfgvtQQavwqJbQ9cM=
            </ds:SignatureValue>
            <ds:KeyInfo><ds:X509Data><ds:X509Certificate>MIIFhjCCBmagAwIBAgIQJHlaK1NuFEBD0L+
              ...QfTY/vJJkcyKxNug6h0WRMA8A33sR3frMchUUPwF3pWB7YX8yLejdC6681rUkTHkPMvDi0rmaeK0BX/t7+nIi
              tdYZZ+Jz1NRCtE=</ds:X509Certificate><ds:X509Data><ds:KeyValue><ds:RSAKeyValue><ds:Modulus
              >AK4+XKbPxINVvVyYall60uLvrZHZS0mdjkbRRSSdkC8WOXbjSnx3BUSRv8H4I68GXHD2SNuc2HjfrGCnK2pInVi95
              VDEfACtkuTF8iYVpplAk3Gqan34wBCUCLNu93ALlmNd0VDUQ8ZUhb1K7MJ/LU47YNhs8sJRGHct4yk/m0YZ</ds:Mo
              dulus><ds:Exponent>AQAB</ds:Exponent></ds:RSAKeyValue></ds:KeyValue></ds:KeyInfo><ds:Objec
              t><xades:QualifyingProperties xmlns:xades="http://uri.etsi.org/01903/v1.2.2#" Target="#id-1795ea06-d2ec-4dd0-8b67-05f9e47fa6de"><xades:SignedProperties Id="id-92759a28-aff4-4567-96f2-9d534d156592"><xades:SignedSignatureProperties><xades:SigningCertificate><xades:Cert><xades:CertDigest><ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"></ds:DigestMethod><ds:DigestValue>/O+sUUrDp+pwvhlslJkCOC2zlnw=</ds:DigestValue></xades:CertDigest><xades:IssuerSerial><ds:X509IssuerName>CN=EC-SAFP,OU=Secretaria d'Administracio i Funcio Publica,OU=Vegeu https://www.catcert.net/verCIC-2 (c)03,OU=Serveis Publics de Certificacio ECV-2,L=Passatge de la Concepcio 11 08008 Barcelona,O=Agencia Catalana de Certificacio (NIF Q-0801176-I),C=ES</ds:X509IssuerName><ds:X509SerialNumber>48482304617635335619431060243083832167</ds:X509SerialNumber></xades:IssuerSerial></xades:Cert></xades:SigningCertificate></xades:SignedSignatureProperties></xades:SignedProperties></xades:QualifyingProperties></ds:Object></ds:Signature></b></a></urn:InlineXML>
        </urn:Document>
      </urn:InputDocuments>
    <urn:SignatureObject>
      <urn:SignaturePtr WhichDocument="doc"
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#" XPath="."/ds:Signature"/>
    </urn:SignatureObject>
  </urn:VerifyRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</soapenv:Body></soapenv:Envelope>
```

Figura 9 Missatge de validació de signatura XML attached enveloped

Missatge per a la validació d'una signatura **XML detached**, on s'està signant continguts presents a la petició de manera externa a la mateixa. La signatura es proveeix dintre d'un element SignatureObject, com en el cas enveloping.

#### Validació de signatura XML detached

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <dss:VerifyRequest Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:OptionalInputs>
        <dss:ReturnProcessingDetails/>
      </dss:OptionalInputs>
      <urn:InputDocuments>
        <urn:Document ID="doc" RefURI="mydoc.catcert.net">
          <urn:InlineXML>
            <a><b>.....</b></a>
          </urn:InlineXML>
        </urn:Document>
      </urn:InputDocuments>
      <dss:SignatureObject>
        <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
          <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="mydoc.catcert.net">
              <ds:Transforms>
                <ds:Transform
                  Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
                <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>
              8sEMF3ipDgNdjPoShP51Hbgly4k </ds:DigestValue>
          </ds:Reference>
          </ds:SignedInfo>
          <ds:SignatureValue> a/kmv5..... </ds:SignatureValue>
          <ds:KeyInfo>
            <ds:KeyValue>
              <ds:RSAKeyValue>
                <ds:Modulus> uY17h.....
              </ds:Modulus>
                <ds:Exponent>AQAB</ds:Exponent>
              </ds:RSAKeyValue>
            </ds:KeyValue>
            <ds:X509Data>
              <ds:X509Certificate> MIIH...
            </ds:X509Certificate>
          </ds:X509Data>
          </ds:KeyInfo></ds:Signature>
        </dss:SignatureObject>
      </dss:VerifyRequest>
    </soapenv:Body>
  </soapenv:Envelope>
```

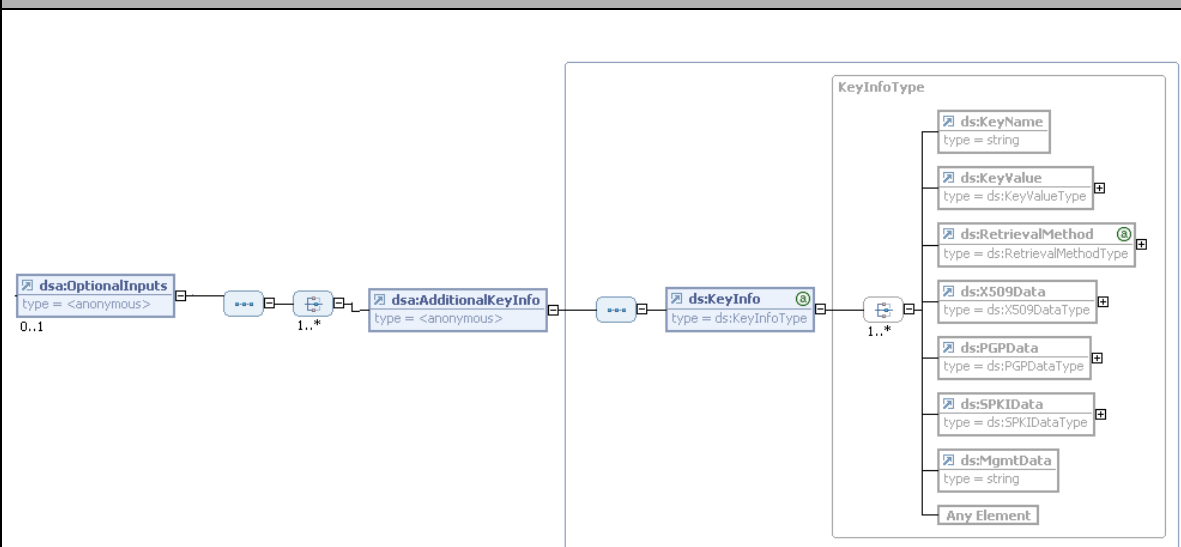
Figura 10 Missatge de validació de signatura XML detached

A més, de manera idèntica a la resta de peticions basades en DSS conté una sèrie de *OptionalInputs* que procedim a detallar:

Estructura de <i>ReturnProcessingDetails</i>	
Element	Descripció
<i>ReturnProcessingDetails</i>	<p>Molt similar al seu homònim ja tractat amb anterioritat a la validació de certificats, però en aquest cas proporciona detalls sobre la validació de la signatura en sí, a més dels referits a l'estat del certificat amb el qual es va realitzar la signatura.</p> <p><i>DSS Core Protocols, Elements and Bindings. Apartat 4.6.4</i></p>

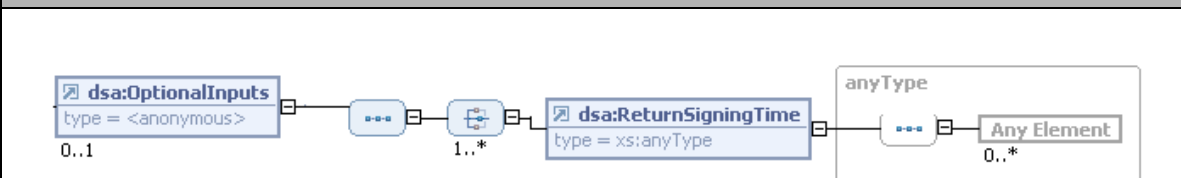
Estructura de <i>VerificationTime</i>	
Element	Descripció
<i>VerificationTime</i>	<p>Proporciona un instant de temps al·legat per a la validació de la signatura. Si no es proporciona cap, l'instant de validació per a aquest tipus de signatures serà l'actual.</p> <p><i>DSS Core Protocols, Elements and Bindings. Apartat 4.6.2</i></p>

### Estructura de *AdditionalKeyInfo*



Element	Descripció
<i>AdditionalKeyInfo</i>	<p>Proporciona al servidor informació addicional útil en el procés de validació com ara CA's intermitjies o CRL's necessàries per tal de poder validar la signatura.</p> <p><i>DSS Core Protocols, Elements and Bindings. Apartat 4.6.3</i></p>

### Estructura de *ReturnSigningTime*



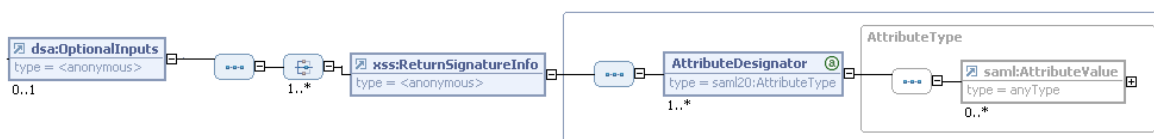
Element	Descripció
<i>ReturnSigningTime</i>	<p>Demana al servidor que retorni la data en la qual es va dur a terme la signatura, si és possible determinar-la a partir de la mateixa.</p> <p>La utilització d'aquest element generarà la creació de l'element <i>SigningTime</i> dintre de l'estructura <i>OptionalOutputs</i>.</p> <p><i>DSS Core Protocols, Elements and Bindings. Apartat 4.6.5</i></p>

#### Estructura de *ReturnSignerIdentity*



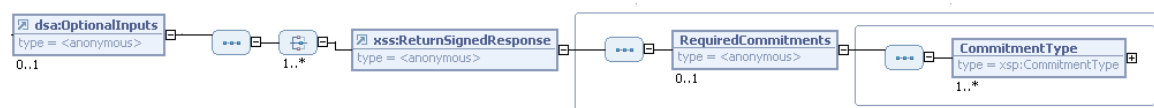
Element	Descripció
<i>ReturnSignerIdentity</i>	<p>Demana al servidor que retorni la identitat del creador de la signatura.</p> <p>La utilització d'aquest element generarà la creació de l'element <i>SignerIdentity</i> dintre de l'estructura <i>OptionalOutputs</i>.</p> <p><i>DSS Core Protocols, Elements and Bindings. Apartat 4.6.6</i></p>

#### Estructura de *ReturnSignatureInfo*



Element	Descripció
<i>ReturnSignatureInfo</i>	<p>Sol·licitud de consulta d'un atribut de la signatura o del seu certificat.</p> <p>La utilització d'aquest element passa per crear un nou element fill de tipus <i>AttributeDesignator</i> que contindrà un atribut amb el nom de "Name" i que agafarà el nom de l'atribut a consultar.</p> <p>Es poden consultar 1 o N atributs en una mateixa consulta. En els annexes hi ha informació de tots els atributs de consulta disponibles.</p> <p><i>XSS Profile of the OASIS DSS. Apartat 3.1.4</i></p>

#### Estructura de *ReturnSignedResponse*



Element	Descripció
<i>ReturnSignedResponse</i>	<p>Demana al servidor que retorni la resposta signada.</p>

XSS Profile of the OASIS DSS. Apartat 3.1.2

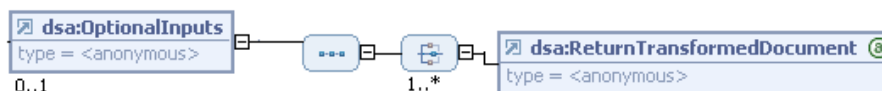
#### Estructura de *RequireQualifiedCertificate*



Element	Descripció
<i>RequireQualifiedCertificate</i>	Indica al servidor que verifiqui que el certificat que es farà servir per a verificar la signatura és un certificat correcte (d'acord amb la normativa de <i>EC Directive On Electronic Signatures</i> )  La utilització d'aquest element no genera cap sortida dintre de l'estructura <i>OptionalOutputs</i> .  <i>XSS Profile of the OASIS DSS. Apartat 5.1.6</i>

També podem trobar *OptionalInputs* exclusius per a signatures del tipus XML.

#### Estructura de *ReturnTransformedDocument*



Element	Descripció
<i>ReturnTransformedDocument</i>	Indica al servidor que ha de retornar en la resposta el document transformat sota una referència particular.  La utilització d'aquest element generarà la creació de l'element <i>TransformedDocument</i> dintre de l'estructura <i>OptionalOutputs</i> .  <i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.8</i>

#### Estructura de *VerifyManifests*



Element	Descripció
<i>VerifyManifests</i>	<p>Indica al servidor que ha de verificar els “Manifest” de la signatura.</p> <p>La utilització d’aquest element generarà la creació de l’element <i>VerifyManifestResults</i> dintre de l’estructura <i>OptionalOutputs</i>.</p> <p><b>La implementació actual de la plataforma PSIS no dona suport a aquest element.</b></p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.1</i></p>

## Missatge de sortida

Missatge de sortida per a una validació de signatura
<pre> &lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;soapenv:Body&gt;     &lt;dss:VerifyResponse Profile="urn:oasis:names:tc:dss:1.0:core:schema"       xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"&gt;       &lt;dss:Result&gt;         &lt;dss:ResultMajor&gt; urn:oasis:names:tc:dss:1.0:resultmajor:Success &lt;/dss:ResultMajor&gt;         &lt;dss:ResultMinor&gt; urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:onAllDocuments         &lt;/dss:ResultMinor&gt;       &lt;/dss:Result&gt;       &lt;dss:OptionalOutputs&gt;         &lt;dss:ProcessingDetails&gt;           &lt;dss:ValidDetail Type="urn:oasis:names:tc:dss:1.0:detail:Signature"&gt;             &lt;dss:Message xml:lang="en"&gt; The signature is valid. &lt;/dss:Message&gt;           &lt;/dss:ValidDetail&gt;         &lt;/dss:ProcessingDetails&gt;       &lt;/dss:OptionalOutputs&gt;     &lt;/dss:VerifyResponse&gt;   &lt;/soapenv:Body&gt; &lt;/soapenv:Envelope&gt; </pre>

Figura 11 Missatge de sortida per a una validació de signatura

El missatge de sortida segueix l’estructura bàsica plantejada per l’estàndard DSS.

Els elements més importants continguts a la resposta d’una validació de signatura són:

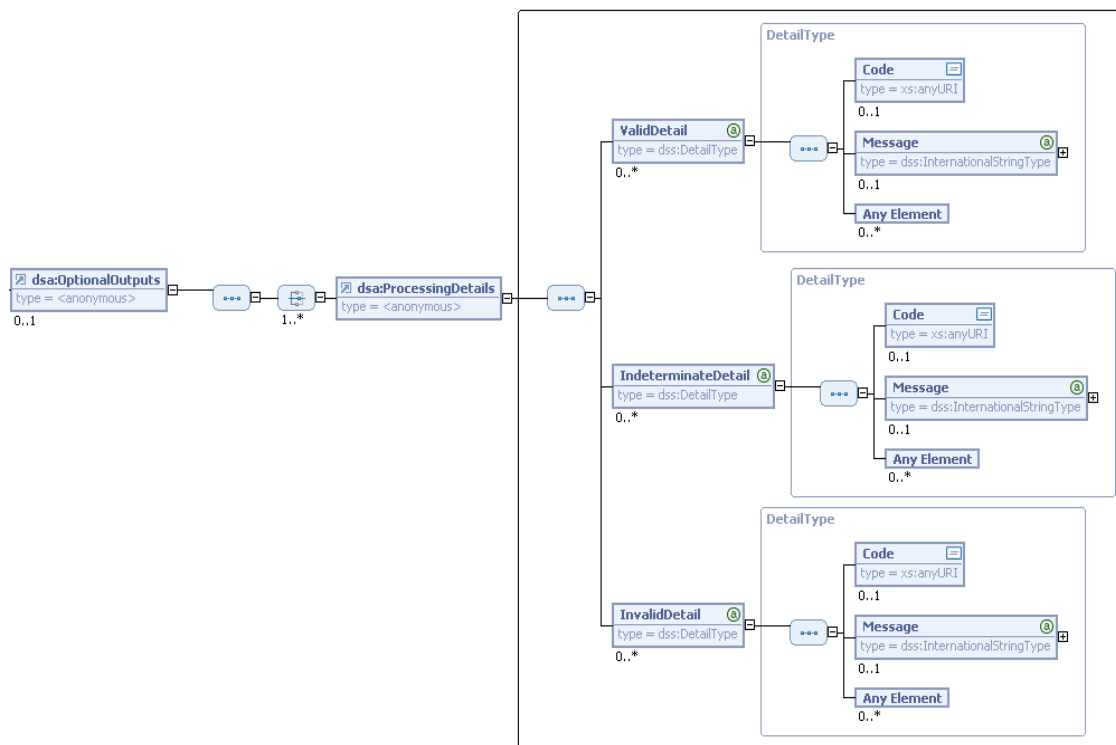
- *Result*

Estructura de dades amb el resultat del procés de validació seguint l’estructura definida a DSS.

- *OptionalOutputs*

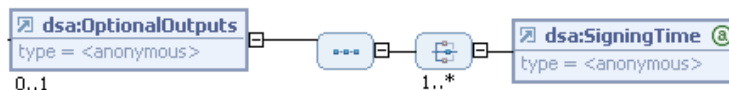
Estructura de dades que contindrà la informació sol·licitada pel client amb els elements introduïts dins de l’estructura *OptionalInputs*.

#### Estructura de *ProcessingDetails*



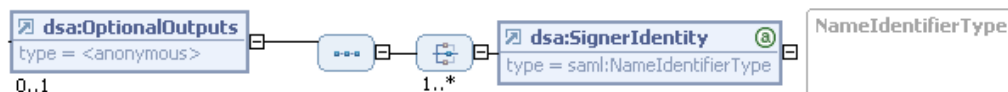
Element	Descripció
<i>ProcessingDetails</i>	<p>Informació detallada del procés de validació.</p> <p>Resposta obtinguda per l'enviament de la sol·licitud <i>ReturnProcessingDetails</i>.</p> <p><i>DSS Core Protocols, Elements, and Bindings</i>. Apartat 4.6.4</p>

#### Estructura de *SigningTime*



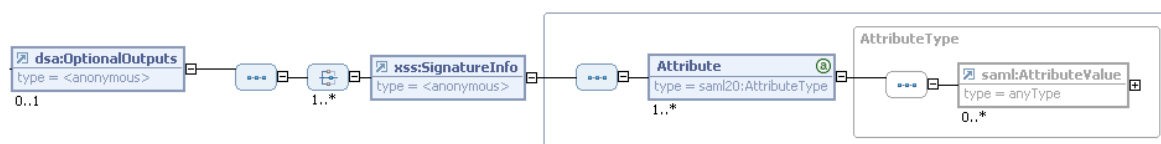
Element	Descripció
<i>SigningTime</i>	<p>Retorna el temps al·legat de signatura si aquest es troba present a la mateixa.</p> <p><i>DSS Core Protocols, Elements, and Bindings</i>. Apartat 4.6.5</p>

#### Estructura de *SignerIdentity*



Element	Descripció
<i>SignerIdentity</i>	Retorna la identitat del creador de la signatura.  <i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.6</i>

#### Estructura de *SignatureInfo*



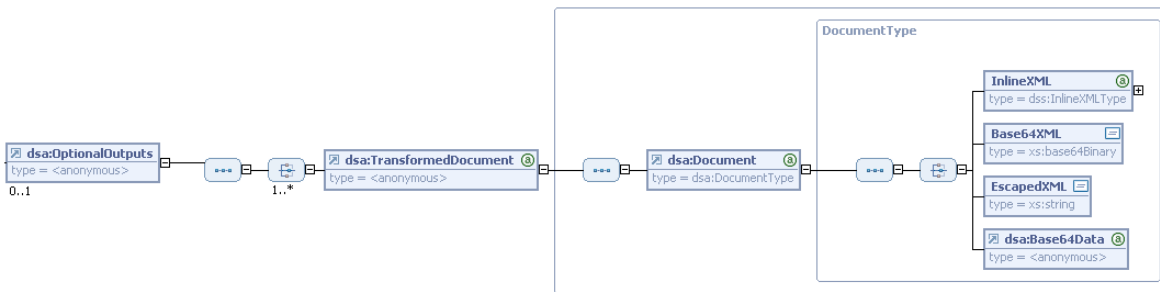
Element	Descripció
<i>SignatureInfo</i>	Retorna la informació extreta de la signatura o del seu certificat a partir del demanat a l' <i>OptionalInput ReturnSignatureInfo</i>  <i>XSS Profile of the OASIS DSS. Apartat 3.1.4</i>

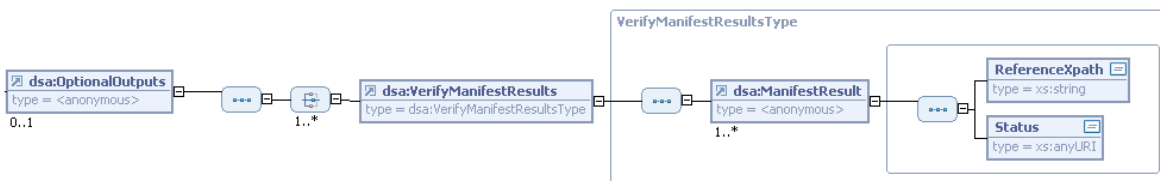
#### Estructura de *ResponseSignature*



Element	Descripció
<i>ResponseSignature</i>	Retorna una signatura sobre el missatge de resposta. En el cas que ens ocupa és una signatura XAdES-C sobre la <i>VerifyResponse</i>  <i>XSS Profile of the OASIS DSS. Apartat 3.1.2</i>

*OptionalOutputs* exclusius per a signatures XML.

Estructura de <i>TransformedDocument</i>	
	
Element	Descripció
<i>TransformedDocument</i>	<p>Retorna el document apuntat per la referència indicada per l'<i>OptionalInput</i>.</p> <p>El document transformat és el resultat d'aplicar totes les transformacions indicades a la signatura sobre el mateix.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.8</i></p>

Estructura de <i>VerifyManifestResults</i>	
	
Element	Descripció
<i>fManifestResult</i>	<p>Retorna el resultat de verificar els elements <i>Manifest</i> presents a la signatura XML.</p> <p><b>La implementació actual de la plataforma PSIS no pot generar aquesta sortida perquè no dóna suport a la seva petició.</b></p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.1</i></p>

## 5.3 Validació de signatures XAdES

Les signatures avançades estan dotades d'informació addicional i més robustesa enfront les signatures simples. XAdES i CAdES són els formats avançats de les signatures XMLDsig i CMS respectivament.

Aquestes signatures poden arribar a contenir informació de referències al certificat del signant, referències a la cadena de certificació i a la informació de revocació de la mateixa o fins i tot aquesta informació en sí. A part d'això, la signatura i la seva informació estan protegides per segells de temps per a dotar a la signatura d'instant de creació certificat, vigència durant el temps i protecció enfront de la majoria d'atacs criptogràfics fins i tot contra les CA's emissores dels certificats implicats.

Per a més informació sobre aquests tipus de signatura, es poden consultar els documents ETSI TS 101 903 v1.2.2 de *XML Advanced Electronic Signatures (XAdES)* i ETSI TS 101 733 de *CMS Advanced Electronic Signatures (CAdES)*.

En aquest apartat s'estudiarà la validació de signatures XAdES. El format dels missatges de petició és el mateix que l'utilitzat per a les signatures XML i tots els aspectes relatius a la topologia de la signatura respecte als elements signats és idèntica. Només trobem diferència en que les signatures XAdES disposen de *OptionalInputs/Outputs* no disponibles per a les signatures XML normals.

### Missatge d'entrada

En aquest missatge mostrem una petició de validació de signatura *XAdES attached enveloped*. Podem observar que el missatge es idèntic al cas XML i només varia la forma de la signatura en sí.

#### Validació d'una signatura XAdES

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <urn:VerifyRequest Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
      xmlns:xss="urn:oasis:names:tc:dss:1.0:profiles:XSS"
      xmlns:urn="urn:oasis:names:tc:dss:1.0:core:schema"
      xmlns:xsp="http://uri.etsi.org/2038/v1.1.1#"
      xmlns:xd="http://www.w3.org/2000/09/xmldsig#"
      xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
      <urn:OptionalInputs>
        <urn:ReturnProcessingDetails/>
      </urn:OptionalInputs>
      <urn:InputDocuments>
        <urn:Document ID="doc" RefURI="">
          <urn:InlineXML>
            <dss:VerifyResponse
              xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
              Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
              RequestID="I4e4e3dc3e3d38fa80887d84c52f97ce3"><dss:Result>
```

```

urn:oasis:names:tc:dss:1.0:resultmajor:Success </dss:ResultMajor>
    <dss:ResultMinor>
urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:onAllDocuments
    </dss:ResultMinor>
    </dss:Result>
    <dss:OptionalOutputs>
    <urn:ResponseSignature
    <ds:Signature
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="id-1795ea06-d2ec-4dd0-8b67-
05f9e47fa6de">
    <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="">
    <ds:Transforms>
    <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue> 630+0/BNGDxKudkxe5SUmEjFRmE=</ds:DigestValue>
    </ds:Reference>
    <ds:Reference Type="http://uri.etsi.org/01903/v1.2.2#SignedProperties" URI="#id-
92759a28-aff4-4567-96f2-9d534d156592">
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue> QTpnCN3iXDOXA6pw6jyOALtgFYg=</ds:DigestValue>
    </ds:Reference></ds:SignedInfo>
    <ds:SignatureValue>
TnUs37pGeU43rONYqW9LgtEcmLvDH//ALUpCJpVlDyEy5uFRgoMV5UK6xg99dNCYEvK9Rf3Ho9R0
017prbL6xwWsPnW7i01GPDkxWGFSS2J6+hdQLRVoDulIluX5Lw/09Fb0LtyUVhdAyBFjaWWzHTvV
MQQfgvtQQavwqJbQ9cM= </ds:SignatureValue><ds:KeyInfo>
    <ds:X509Data>
    <ds:X509Certificate>MIIHfjCCBmagAwIBAgIQJHlaK1NuFEBD0L+...Jz1NRCtE=</ds:X509Certifi
cate>
    </ds:X509Data>
    <ds:KeyValue>
    <ds:RSAKeyValue>
    <ds:Modulus>
AK4+XKbPxINVvVyYall60uLvrZHJS0mdjkbRRsSdkC8W0Xbjsnx3BUsRv8H4I68GXHD2SNuc2HjfrGCnK2p
InVi95VDEfACtkuTF8iyVpplAk3GqaN34wBCUC1Nu93ALlmNd0VDUQ8ZUhb1K7MJ/LU47YNhs8sJRGHct4yk/m0YZ
</ds:Modulus>
    <ds:Exponent> AQAB </ds:Exponent>
    </ds:RSAKeyValue>
    </ds:KeyValue></ds:KeyInfo>

    <ds:Object>
    <xades:QualifyingProperties xmlns:xades="http://uri.etsi.org/01903/v1.2.2#"
Target="#id-1795ea06-d2ec-4dd0-8b67-05f9e47fa6de">
    <xades:SignedProperties Id="id-92759a28-aff4-4567-96f2-9d534d156592">
    <xades:SignedSignatureProperties>
    <xades:SigningCertificate>
    <xades:Cert>
    <xades:CertDigest>
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
    <ds:DigestValue>/O+sUUrDp+pwvh1s1JkCOC2zlnw=</ds:DigestValue>
    </xades:CertDigest>
    <xades:IssuerSerial>
    <ds:X509IssuerName>
CN=EC-SAFP,OU=Secretaria
d'Administracio i Funcio
Publica,OU=Vegeu
https://www.catcert.net/verCIC-2
(c)03,OU=Serveis Publics de
Certificacio
ECV-2,L=Passatge de la
Concepcio 11 08008
Barcelona,O=Agencia Catalana
    </ds:X509IssuerName>
    <ds:X509SerialNumber>48482304617635335619431060243083832167</ds:X509SerialNumber>
    </xades:IssuerSerial>
    </xades:Cert>
    </xades:SigningCertificate>
    </xades:SignedSignatureProperties>
    </xades:SignedProperties>
    </xades:QualifyingProperties></ds:Object>

```

```

</ds:Signature>
</urn:ResponseSignature>
</dss:OptionalOutputs>
</dss:VerifyResponse>
</urn:InlineXML>
</urn:Document>
</urn:InputDocuments>
<urn:SignatureObject>
  <urn:SignaturePtr WhichDocument="doc"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" XPath="."/ds:Signature"/>
  </urn:SignatureObject>
</urn:VerifyRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Figura 12 Missatge de validació d'una signature XAdES

Tanmateix, alguns dels *OptionalInputs* ja mencionats per a les signatures simples canvien lleugerament el seu significat. Detallarem els *OptionalInputs/Outputs* nous, així com el seu nou significat:

Estructura de <i>ReturnSigningTime</i>	
Element	Descripció
<i>ReturnSigningTime</i>	<p>En aquest cas es idèntic al cas simple, però si no trobés un instant de temps al·legat dintre de la signatura, donaria com a instant més proper a la creació l'instant d'estampació del segell de temps més antic sobre la signatura que sigui vàlid.</p> <p>Tenim així una cota superior d'existència de la signatura certificada pel segell de temps.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.5</i></p>

Estructura de <i>ReturnUpdatedSignature</i>	
Element	Descripció
<i>ReturnUpdatedSignature</i>	<p>Demana al servidor que retorni la signatura actualitzada a la forma demanada. Hi poden haver multitud de formes (com ara les definides a XAdES) i aquestes defineixen una sèrie d'atributs que ha de contenir una signatura per a complir aquesta forma. El servidor intentarà afegir els atributs no presents a la signatura fins a complir la forma especificada.</p> <p>Atributs poden ser, per exemple, referències a les CA's del camí de certificació, el path en sí, referències a informació de revocació (CRL/OCSP) o la informació en sí, així com els diferents tipus de <i>timestamps</i> definits per XAdES.</p>

DSS Core Protocols, Elements, and Bindings. Apartat 4.6.7

## Missatge de sortida

### Resposta a una validació d'una signature XAdES

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <dss:VerifyResponse Profile="urn:oasis:names:tc:dss:1.0:profiles:XSS"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:Result>
        <dss:ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</dss:ResultMajor>
        <dss:ResultMinor>urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:onAllDocuments</dss:ResultMinor>
      </dss:Result>
      <dss:OptionalOutputs>
        <dss:ProcessingDetails>
          <dss:ValidDetail Type="urn:oasis:names:tc:dss:1.0:detail:Signature">
            <dss:Message xml:lang="en">The signature is valid.</dss:Message>
          </dss:ValidDetail>
        </dss:ProcessingDetails>
        <dss:UpdatedSignature
          Type="urn:oasis:names:tc:dss:1.0:profiles:XAdES:forms:ES-C">
          <dss:SignatureObject>
            <ds:Signature Id="id-1795ea06-d2ec-4dd0-8b67-05f9e47fa6de"
              xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <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="">
                    <ds:Transforms>
                      <ds:Transform
                        Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"
                        />
                    </ds:Transforms>
                  </ds:Reference>
                  <ds:DigestMethod
                    Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                  <ds:DigestValue>630+0/BNGDxKudkxe5SUmEjFRmE</ds:DigestValue>
                </ds:Reference>
                <ds:Reference
                  Type="http://uri.etsi.org/01903/v1.2.2#SignedProperties"
                  URI="#id-92759a28-aff4-4567-96f2-9d534d156592">
                    <ds:DigestMethod
                      Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                    <ds:DigestValue>QTpnCN3iXDOXa6pw6jyOALtgFYg</ds:DigestValue>
                  </ds:Reference>
                </ds:SignedInfo>
                <ds:SignatureValue Id="id-014f59ae-c379-4a63-9d76-bb41eb33eb29">TnUs37pGeU43rONYqW9LgtEcmLvDH//ALUpCJpV1DyEy5uFRgoMV5UK6xg99dNCYEvK9Rf3Ho9R0
                  017prbL6xwWsPnW7iO1GPDkxWGFSS2J6+hdQLRVoDulIluX5Lw/09Fb0LtyUVhdAyBFjaWWzHTvV
                  MQQfgvtQQavwqJbQ9cM=</ds:SignatureValue>
              </ds:KeyInfo>
                <ds:X509Data>
                  <ds:X509Certificate>MIIHfjCCBmagAwIBAgIQJHlaK1NuFEBD0L+...
                    NeQCEvpdHiBgwjts4pml76xs+8y0zor2MY6jBh0Iln/W0iS1lvoTeQfTY/vJJkcyKxDNug6h0WRMA8A33sR3frMChU
                    UPwF3pWB7XYX8yLejdC6681rUkTHkPMvDi0rmaeK0BX/t7+nIttdyZZ+Jz1NRctE=</ds:X509Certificate>
                  </ds:X509Data>
                  <ds:KeyValue>
                    <ds:RSAKeyValue>
                      <ds:Modulus>AK4+XKbPxINVvVyYall60uLvrZHS0mdjkbRRsSdKc8WOxbjSnx3BUSRv8H4I68GXHD2SNuc2HjfrG
                        CnK2pInVi95VDEfActkuTF8iyVpplAk3GqaN34wBCUC1Nu93ALlmNd0VDUQ8ZUhb1K7MJ/LU47YNhs8sJRGHct4yk/
                        mOYZ</ds:Modulus>
                      <ds:Exponent>AQAB</ds:Exponent>
                    </ds:RSAKeyValue>
                  </ds:KeyValue>
                </ds:KeyInfo>
              </ds:Object>
```

```

<xades:QualifyingProperties
  Target="#id-1795ea06-d2ec-4dd0-8b67-05f9e47fa6de"
  xmlns:xades="http://uri.etsi.org/01903/v1.2.2#">
  <xades:SignedProperties
    Id="id-92759a28-aff4-4567-96f2-9d534d156592">
    <xades:SignedSignatureProperties>
      <xades:SigningCertificate>
        <xades:Cert>
          <xades:CertDigest>
            <ds:DigestMethod
              Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue>/O+sUUrDp+pwvhlslJkCOC2zlnw=</ds:DigestValue>
          </xades:CertDigest>
          <xades:IssuerSerial>
            <ds:X509IssuerName>CN=EC-
              SAFF,OU=Secretaria
              d'Administracio i Funcio
              Publica,OU=Vegeu
              https://www.catcert.net/verCIC-2
              (c)03,OU=Serveis Publics de Certificacio
              ECV-2,L=Passatge de la Concepcio 11
              08008 Barcelona,O=Agencia Catalana de
              Certificacio (NIF Q-0801176-
              I),C=ES</ds:X509IssuerName>
            <ds:X509SerialNumber>48482304617635335619431060243083832167</ds:X509SerialNumber>
          </xades:IssuerSerial>
        </xades:Cert>
      </xades:SigningCertificate>
    </xades:SignedSignatureProperties>
  </xades:SignedProperties>
  <xades:UnsignedProperties>
    <xades:UnsignedSignatureProperties>
      <xades:SignatureTimeStamp
        Id="id-2c293f14-c0fa-48ac-8334-
        55b96b48a726">
        <xades:Include referencedData="false"
          URI="#id-014f59ae-c379-4a63-9d76-
          bb41eb33eb29"/>
        <ds:CanonicalizationMethod
          Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
        <xades:XMLTimeStamp>
          <ds:Signature
            Id="id-34e2090e-ac4d-4605-a7bb-
            2a4a1fe95f75">
            <ds:SignedInfo
              Id="id-f11abe77-f27e-4982-a841-
              19b531396625">
              <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="#TSTInfo-id-04ce4298-2b7c-4d7f-
                83c0-1fbc664aa607" Id="id-c811cbcd-eb2d-4d7c-94c5-353c1ae0e61d"
                Type="urn:oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken">
                <ds:Transforms>
                  <ds:Transform
                    Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"
                    />
                </ds:Transforms>
              <ds:DigestMethod
                Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
              <ds:DigestValue>HTLZhjNc8tRUWZewDHdEZKfi+QA=</ds:DigestValue>
            </ds:Reference>
          </ds:SignedInfo>
        </ds:Signature>
      </xades:SignatureTimeStamp>
    </xades:UnsignedSignatureProperties>
  </xades:UnsignedProperties>
  <xades:SignatureValue>
    <ds:Signature
      Id="id-8cfc0532-f664-4bb7-8f40-
      345120d0ba02">
      <ds:SignedInfo
        Id="id-34e2090e-ac4d-4605-a7bb-
        2a4a1fe95f75">
        <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="#TSTInfo-id-04ce4298-2b7c-4d7f-
          83c0-1fbc664aa607" Id="id-c811cbcd-eb2d-4d7c-94c5-353c1ae0e61d"
          Type="urn:oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken">
          <ds:Transforms>
            <ds:Transform
              Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"
              />
          </ds:Transforms>
        <ds:DigestMethod
          Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <ds:DigestValue>HTLZhjNc8tRUWZewDHdEZKfi+QA=</ds:DigestValue>
      </ds:SignedInfo>
    </ds:Signature>
  </xades:SignatureValue>
  <xades:KeyInfo>
    <ds:KeyInfo
      Id="id-8cfc0532-f664-4bb7-8f40-
      345120d0ba02">
      <ds:X509Data>
        <ds:X509Certificate>MIIHIDCCBgigAwIBAgIQbg5vDBUDLv1ELR24xDVsZzANBgkqhkiG9w0BAQUFADCB8zELMA
      </ds:X509Certificate>
    </ds:KeyInfo>
  </xades:KeyInfo>
</xades:QualifyingProperties>

```

```

kGA1UEBhMCRVMxOzA5BgNVBAoTMkFnZW5jaWEgQ2F0YWxhbmEgZGUGQ2VydGlmawNhY2l1vICChOSUYgUS0wODAxMTc2
LUkpMSgwJgYDVQQLEx9TZXJ2ZWlzlFB1YmXpY3MgZGUGQ2Vyd...+d09C8ZtLIwVdld9vbYAp9n0TvdxbWrHSHvFfk
FWjq8HRUD+ptXHDZaWBxmuUQ==</ds:X509Certificate>

</ds:X509Data>
<ds:KeyValue>
  <ds:RSAKeyValue>
    <ds:Modulus>ALMdRzdnoHg90PT4ukAz6VPNL+qDcfOjE7R+1N08SdlvqeFarXgkYze36dJ3J2ypXHf+vKWF5HsEYy
    jfsCkPRyil6CWYoqOfyhycVr1X3gAmsFSlQmiIZsrbuE3cXR+4I2ZIxGlvqbpSVRp0NmF090W5s4OofWbbemGSldpK
    pA8v</ds:Modulus>
    <ds:Exponent>AQAB</ds:Exponent>
  </ds:RSAKeyValue>
</ds:KeyValue>
</ds:KeyInfo>
<ds:Object>
  Id="TSTInfo-id-04ce4298-2b7c-4d7f-83c0-
1fbc664aa607" MimeType="application/xml"> <dss:TstInfo>
<dss:SerialNumber>435853668217230596844338280440350640549866592402</dss:SerialNumber>
  <dss:CreationTime>2006-08-
07T09:34:13.703Z</dss:CreationTime>
<dss:Policy>urn:oid:0.4.0.2023.1.1</dss:Policy>

  <dss:ErrorBound>PT1S</dss:ErrorBound>
  <dss:Ordered>true</dss:Ordered>
  <dss:TSA>

Format="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName"
  >CN=Servei de segellat de temps
  de PSIS,OU=Jerarquia Entitats de
  Certificacio Catalanes,OU=Vegeu
  https://www.catcert.net/verCIT-1
  (c)05,OU=Serveis Publics de
  Certificacio CIT-1,O=Agencia
  Catalana de Certificacio (NIF
  Q-0801176-I),C=ES</dss:TSA>
</dss:TstInfo>
</ds:Object>
</ds:Signature>
</xades:XMLTimeStamp>
</xades:SignatureTimeStamp>
<xades:CompleteCertificateRefs
  Id="id-e5c51748-e601-4837-af7b-
3c8430f3be0d">
  <xades:CertRefs>
    <xades:Cert>
      <xades:CertDigest>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>CVWrcqg7NzTDiQfhH9KVZkJxVuY=</ds:DigestValue>
        </xades:CertDigest>
      <xades:IssuerSerial>
        <ds:X509IssuerName>CN=EC-
de Catalunya,OU=Vegeu
https://www.catcert.net/verCIC-1
(c)03,OU=Serveis Publics de
Certificacio ECV-1,O=Agencia
Catalana de Certificacio (NIF
Q-0801176-I),C=ES</ds:X509IssuerName>
<ds:X509SerialNumber>148786221414740922626357724534625015094</ds:X509SerialNumber>
      </xades:IssuerSerial>
    </xades:Cert>
    <xades:Cert>
      <xades:CertDigest>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>8Ldbt5MRsOXQEPhtjcf1jxW+aP0=</ds:DigestValue>
        </xades:CertDigest>
      <xades:IssuerSerial>
        <ds:X509IssuerName>CN=EC-
ACC,OU=Jerarquia
Entitats de Certificacio
Catalanes,OU=Vegeu
https://www.catcert.net/verarrel
(c)03,OU=Serveis Publics de
Certificacio,O=Agencia Catalana de
Certificacio (NIF Q-0801176-
I),C=ES</ds:X509IssuerName>

```

```

<ds:X509SerialNumber>43517894977975666787701712876307936290</ds:X509SerialNumber>
    </xades:IssuerSerial>
  </xades:Cert>
</xades:CertRefs>
</xades:CompleteCertificateRefs>
<xades:CompleteRevocationRefs
  Id="id-46e2a30c-2011-4e75-af6f-
aa82cc2baf1c">
  <xades:CRLRefs>
    <xades:CRLRef>
      <xades:DigestAlgAndValue>
        <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>+FN/vlQis6YFguUHGKlKqrBfUfE=</ds:DigestValue>
        </xades:DigestAlgAndValue>
      <xades:CRLIdentifier>
        <xades:Issuer>CN=EC-SAFP,OU=Secretaria
d'Administracio i Funcio
Publica,OU=Vegeu
https://www.catcert.net/verCIC-2
(c)03,OU=Serveis Publics de
Certificacio ECV-2,L=Passatge de la
Concepcio 11 08008
Barcelona,O=Agencia Catalana de
Certificacio (NIF Q-0801176-
<xades:IssueTime>2006-08-
<xades:Number>796</xades:Number>
</xades:CRLIdentifier>
</xades:CRLRef>
      <xades:CRLRef>
        <xades:DigestAlgAndValue>
          <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>qUpulLKUwAbPLzW6zVUf7DUrfUQ=</ds:DigestValue>
          </xades:DigestAlgAndValue>
        <xades:CRLIdentifier>
          <xades:Issuer>CN=EC-
GENCAT,OU=Generalitat
de Catalunya,OU=Vegeu
https://www.catcert.net/verCIC-1
(c)03,OU=Serveis Publics de
Certificacio ECV-1,O=Agencia
Catalana de Certificacio (NIF
Q-0801176-I),C=ES</xades:Issuer>
<xades:IssueTime>2004-12-
<xades:Number>1</xades:Number>
</xades:CRLIdentifier>
</xades:CRLRef>
      <xades:CRLRef>
        <xades:DigestAlgAndValue>
          <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<ds:DigestValue>t5LTiMY8FEtMMtXn7WITY2sy9l8=</ds:DigestValue>
          </xades:DigestAlgAndValue>
        <xades:CRLIdentifier>
          <xades:Issuer>CN=EC-ACC,OU=Jerarquia
Entitats de Certificacio
Catalanes,OU=Vegeu
https://www.catcert.net/verarrel
(c)03,OU=Serveis Publics de
Certificacio,O=Agencia Catalana de
Certificacio (NIF Q-0801176-
<xades:IssueTime>2004-12-
<xades:Number>1</xades:Number>
</xades:CRLIdentifier>
</xades:CRLRef>
    </xades:CRLRefs>
  </xades:CompleteRevocationRefs>
</xades:UnsignedSignatureProperties>
</xades:UnsignedProperties>
</xades:QualifyingProperties>
</ds:Object>
</ds:Signature>
</dss:SignatureObject>
</dss:UpdatedSignature>
</dss:OptionalOutputs>

```

```
</dss:VerifyResponse>
</soapenv:Body>
</soapenv:Envelope>
```

Figura 13 Missatge de resposta a una validació d'una signature XAdES

El missatge de sortida segueix l'estructura bàsica plantejada per l'estàndard DSS.

Els elements més importants continguts a la resposta d'una validació de signatura XAdES són:

- *Result*

Estructura de dades amb el resultat del procés de validació. Indica el resultat de la validació com la resta de les respostes basades en DSS.

- *OptionalOutputs*

Estructura de dades que contindrà la informació sol·licitada pel client amb els elements introduïts dins de l'estructura *OptionalInputs*.

Estructura de <i>SigningTime</i>	
Element	Descripció
<i>SigningTime</i>	Retorna el temps de signatura. El procés d'obtenció està detallat a l' <i>OptionalInput</i> corresponent.  <i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.5</i>

Estructura de <i>UpdatedSignature</i>	
Element	Descripció
<i>UpdatedSignature</i>	Retorna la signatura actualitzada a la forma demanada (en cas que no la complís) així com la

	<p>URI corresponent a la forma més propera a la signatura retornada que pugui determinar el servidor.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.7</i></p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 5.4 Validació de signatures PDF

La validació de signatures a documents en format PDF és pràcticament igual que la validació d'una signatura qualsevol, a excepció de tres importants canvis i/o afegits:

- La URL del servidor correspon a la mateixa de les peticions normals DSS però afegint el tipus de perfil. Així, si el servidor normal és <http://PSIS...../catcert/dss>, per a validacions de signatures PDF les peticions han de ser dirigides contra <http://PSIS.../catcert/dsspdf>
- Les peticions han de tenir el perfil (*profile*) següent: *urn:OASIS:names:tc:dss:1.0:profiles:DSS\_PDF*
- Als camps opcionals d'entrada/sortida es pot afegir el camp de Motiu de la signatura (*SignatureReason*). Així si es vol obtenir aquesta informació s'ha d'indicar com a opcional d'entrada (*OptionalInput*) el valor **ReturnSignatureReason**.

### Missatge d'entrada

Un possible missatge d'entrada podria ser el següent (on el contingut complert del document PDF, la signatura del qual es vol validar, s'ha eliminat per a una major claredat):

Validació d'un document PDF
<pre>&lt;SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"&gt;   &lt;SOAP-ENV:Body&gt;     &lt;dss:VerifyRequest Profile="urn:oasis:names:tc:dss:1.0:profiles:DSS_PDF"       xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"       xmlns:pdf="urn:oasis:names:tc:dss:1.0:profiles:DSS_PDF"&gt;       &lt;dss:OptionalInputs&gt;         &lt;dss:ReturnSigningTime/&gt;         &lt;dss:ReturnSignerIdentity/&gt;         &lt;pdf:ReturnSignatureReason/&gt;       &lt;/dss:OptionalInputs&gt;       &lt;dss:InputDocuments&gt;         &lt;dss:Document&gt;           &lt;dss:Base64Data&gt;JVBER...&lt;/dss:Base64Data&gt;         &lt;/dss:Document&gt;       &lt;/dss:InputDocuments&gt;     &lt;/dss:VerifyRequest&gt;   &lt;/SOAP-ENV:Body&gt; &lt;/SOAP-ENV:Envelope&gt;</pre>

Figura 14 Missatge de validació d'un document PDF

#### Estructura de *ReturnSigningTime*



Element	Descripció
<i>ReturnSigningTime</i>	Demana al servidor que retorni l'instant de creació de la signatura.

#### Estructura de *ReturnSignerIdentity*



Element	Descripció
<i>ReturnSignerIdentity</i>	Demana al servidor que retorni la identitat de qui va crear aquesta signatura

#### Estructura de *ReturnSignatureReason*

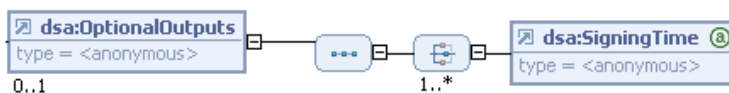


Element	Descripció
<i>ReturnSignatureReason</i>	Demana al servidor que retorni la raó que va definir per a la creació de la signatura el seu creador.

## Missatge de sortida

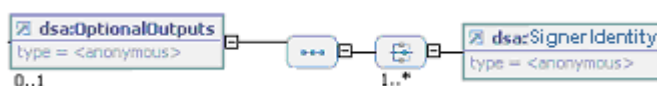
El format del missatge de sortida segueix els mateixos criteris que la resta de validació de signatura.

#### Estructura de *SigningTime*



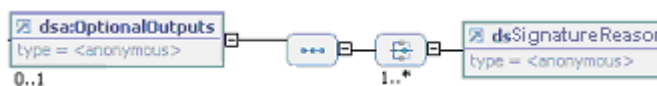
Element	Descripció
<i>SigningTime</i>	Retorna el temps de creació de la signatura sobre el document. El procés d'obtenció està detallat a l' <i>OptionalInput</i> corresponent.

#### Estructura de *SignerIdentity*



Element	Descripció
<i>SignerIdentity</i>	Retorna la identitat del creador de la signatura.

#### Estructura de *SignatureReason*



Element	Descripció
<i>SignatureReason</i>	Retorna la raó de la signatura que el seu creador va especificar quan la va crear.

## 5.5 Creació de segells de temps

Aquesta funcionalitat permet crear un segell de temps basat en documents signats amb el certificat que s'envia al propi missatge.

Un segell de temps és una evidència electrònica mitjançant la qual podem assegurar sense possibilitat de repudiació que una dada existia en un moment de temps determinat. Els segells de temps són creats per unes autoritats especials anomenades TSA (*Timestamp Authorities*).

Les TSA són entitats en els rellotges de les quals, o sistemes de medició del temps, es diposita confiança.

A grans trets, el procés per a obtenir un segell de temps consisteix en generar el resum digital (*digest* o *hash*) del document a segellar per a transmetre'l a la TSA desitjada. La TSA, quan rep el missatge a segellar, genera una marca de temps (*timestamp*) la qual afegeix al hash rebut i el signa digitalment amb la seva clau privada.

En el següent exemple es descriu la missatgeria:

### Missatge d'entrada

En aquest tipus de missatge estem demanant al servidor que ens retorni un segell de temps sobre el document proporcionat. Podem demanar diferents tipus de segells de temps (CMS/XML o XAdES) sobre diferents tipus de components. Això vol dir que els continguts a segellar poden ser un *digest*, dades en *Base64* o XML, però no tots són compatibles amb tots.

A continuació podem veure una taula on es detalla les compatibilitats amb tots els tipus.

Taula comparativa	
Tipus de segell de temps	Tipus de contingut a segellar compatible
CMS	Digest, Base64
XML	Digest, Base64, XML
XAdES	Digest, Base64, XML

Figura 15 Taula de compatibilitat entre formats de segells de temps i els continguts a estampar

Normalment els segells de temps CMS es fan servir per a dades binàries en *Base64* o digerides mentre que els XML i XAdES poden segellar qualsevol tipus de contingut.

Aquí donarem exemples de missatges per a la creació de segells de temps XML i CMS. Cal parar especial atenció al *OptionalInput KeySelector* on es demana al servidor que generi un segell de temps amb el certificat proporcionat. En el cas que ens pertoca haurien de proveir el certificat de TSA de CATCert.

Aquest primer missatge demana la creació d'un segell de temps XML sobre el contingut proporcionat com a document.

Creació d'un segell de temps en format XML

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <dss:SignRequest RequestID="I9e54e5e59e9724683b2379f846ec0f98"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
      Profile="urn:oasis:names:tc:dss:1.0:profiles:timestamping">
      <dss:OptionalInputs>
        <dss:IncludeObject WhichDocument="Doc1"
hasObjectTagsAndAttributesSet="false"
          ObjId="objDoc1" createReference="true"/>
        <dss:KeySelector>
          <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <ds:X509Data>
              <ds:X509Certificate>
/DCCATEGA1UdiwSCASgwgEkgBSgw4tEqjelRb+XgFrR8Xiim+ldjaGB+aSB9jCB8zELMAkGALUEBhMCRVMxOzA5Bg
NVBAoTMkFnZW5jaWEgQ2F0YWxhbmEgZGUgQ2VydGlmawNnY2lvcChOSUYgUS0wODAxMTc2LUkpMSgwJgYDVQQLEx9T
ZXJ2ZWlziFB1YmtpY3MgZGUgQ2VydGlmawNnY2lvcH0wMTUwMwYDVQQLEx9WZWRldSBodHRwc2ovL3d3dy5jYXRzZXJ0Lm
5ldC92ZXJhcnJlbCAoYykwMzE1MDMGA1UECXM5SmVYXJxdWlhIEVudG10YXRzIGRlIENlcnRpZmljYWNpbyBDYXRh
bGFnZXMxZDZANBgNVBAMTBkVdLUFDQ4IQ7is969Qh3hSoYqwe893EATCBxgYDVR0gBIG+
...+jnEIYe/BTisMTXqx8+o0eYoI9uFX/imQmms569KsPXGnVdbuyys5LE6iCfeeOOVwz9ruKeDwX6f+MQw9mkTguh
7vFebCNpyfxIzjbDHXKy1NOeVdd1UYs2tgPhOqHBXZLepU8aRZx3ixKF7TQaipnQc4PLrKUqPrPkQCMtN73b1RrOt
fc+d09C8ZtLIwVEld9vbYAp9n0TvdxbWvHSHvFfkFWjq8HRUD+ptXHDZaWBxmuUQ==</ds:X509Certificate>
            </ds:X509Data>
          </ds:KeyInfo>
        </dss:KeySelector>
      </dss:OptionalInputs>
      <dss:SignatureType>oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken</dss:SignatureType>
    </dss:SignRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Figura 16 Missatge de creació d'un segell de temps en format XML

En el següent exemple es pot comprovar com es pot demanar la creació d'un segell de temps CMS sobre un contingut ja digerit pel client.

Creació d'un segell de temps en format CMS

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <dss:SignRequest
      RequestID="I9e54e5e59e9724683b2379f846ec0f98"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
      Profile="urn:oasis:names:tc:dss:1.0:profiles:timestamping">
      <dss:OptionalInputs>
        <dss:KeySelector>
          <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <ds:X509Data>
              <ds:X509Certificate>MIIHIDCCBgigAwIBAgIQbg5vDBUDLv1ELR24xDVsZzANBgkqhkiG9w0BAQUFADCB8zELMA
kGALUEBhMCRVMxOzA5BgNVBAoTMkFnZW5jaWEgQ2F0YWxhbmEgZGUgQ2VydGlmawNnY2lvcChOSUYgUS0wODAxMTc2
LUkpMSgwJgYDVQQLEx9TZXJ2ZWlziFB1YmtpY3MgZGUgQ2VydGlmawNnY2lvcH0wMTUwMwYDVQQLEx9WZWRldSBodHRwc2ovL3d3dy5jYXRzZXJ0Lm
5ldC92ZXJhcnJlbCAoYykwMzE1MDMGA1UECXM5SmVYXJxdWlhIEVudG10YXRzIGRlIENlcnRpZmljYWNpbyBDYXRh
bGFnZXMxZDZANBgNVBAMTBkVdLUFDQ4IQ7is969Qh3hSoYqwe893EATCBxgYDVR0gBIG+
...+jnEIYe/BTisMTXqx8+o0eYoI9uFX/imQmms569KsPXGnVdbuyys5LE6iCfeeOOVwz9ruKeDwX6f+MQw9mkTguh7vFebCNpyfxIzjbDHXKy1NOeVdd1UY
s2tgPhOqHBXZLepU8aRZx3ixKF7TQaipnQc4PLrKUqPrPkQCMtN73b1RrOtfc+d09C8ZtLIwVEld9vbYAp9n0TvdxbWvHSHvFfkFWjq8HRUD+ptXHDZaWBxmuUQ==</ds:X509Certificate>
            </ds:KeyInfo>
        </dss:KeySelector>
      </dss:OptionalInputs>
    </dss:SignRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

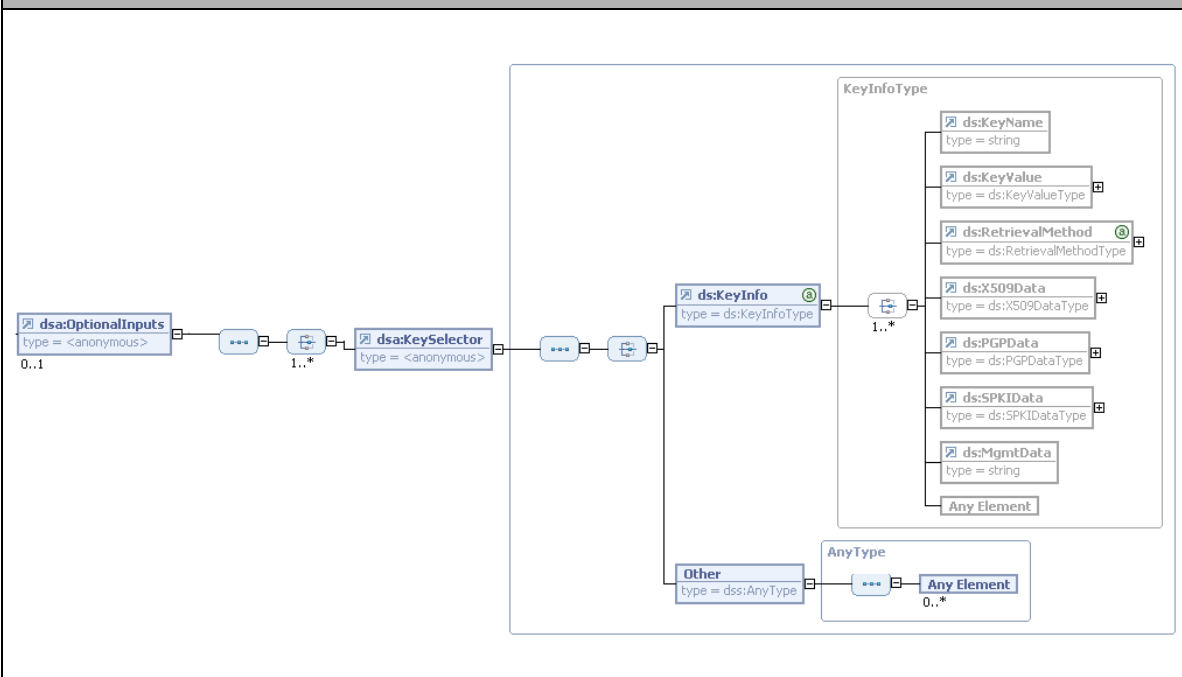
```
bWrHSHvFfkFWjq8HRUD+ptXHDZaWBxmuUQ==</ds:X509Certificate>
  </ds:X509Data>
  </ds:KeyInfo>
</dss:KeySelector>
  <dss:SignatureType>urn:ietf:rfc:3161</dss:SignatureType>
</dss:OptionalInputs>
<dss:InputDocuments>
  <dss:DocumentHash ID="Doc1">
    <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
    <ds:DigestValue
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">2jmj715rSw0yVb/vlWAYkK/YBwk=</ds:DigestValue
>
  </dss:DocumentHash>
</dss:InputDocuments>
</dss:SignRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Figura 17 Missatge de creació d'un segell de temps en format CMS

En el cas de creació de segells de temps, els *OptionalInputs* prenen un paper clau i són determinants per a la correcta creació dels segells.

Estructura de <i>SignatureType</i>									
Element	Descripció								
<i>SignatureType</i>	<p>Determina el tipus de segell de temps a generar pel servidor.</p> <p><i>DSS Core Protocols, Elements, and Bindings. Apartat 3.5.1</i></p> <table> <tr> <th>Tipus de signatura disponibles</th><th>Valor</th></tr> <tr> <td><i>Timestamp</i> (CMS)</td><td>urn:ietf:rfc:3161</td></tr> <tr> <td><i>Timestamp</i> (XMLDsig)</td><td>OASIS:names:tc:dss:1.0:core:schema:XMLTimeStampToken</td></tr> <tr> <td><i>Timestamp</i> (XAdES)</td><td>OASIS:names:tc:dss:1.0:core:schema:XAdESTimeStampToken</td></tr> </table>	Tipus de signatura disponibles	Valor	<i>Timestamp</i> (CMS)	urn:ietf:rfc:3161	<i>Timestamp</i> (XMLDsig)	OASIS:names:tc:dss:1.0:core:schema:XMLTimeStampToken	<i>Timestamp</i> (XAdES)	OASIS:names:tc:dss:1.0:core:schema:XAdESTimeStampToken
Tipus de signatura disponibles	Valor								
<i>Timestamp</i> (CMS)	urn:ietf:rfc:3161								
<i>Timestamp</i> (XMLDsig)	OASIS:names:tc:dss:1.0:core:schema:XMLTimeStampToken								
<i>Timestamp</i> (XAdES)	OASIS:names:tc:dss:1.0:core:schema:XAdESTimeStampToken								

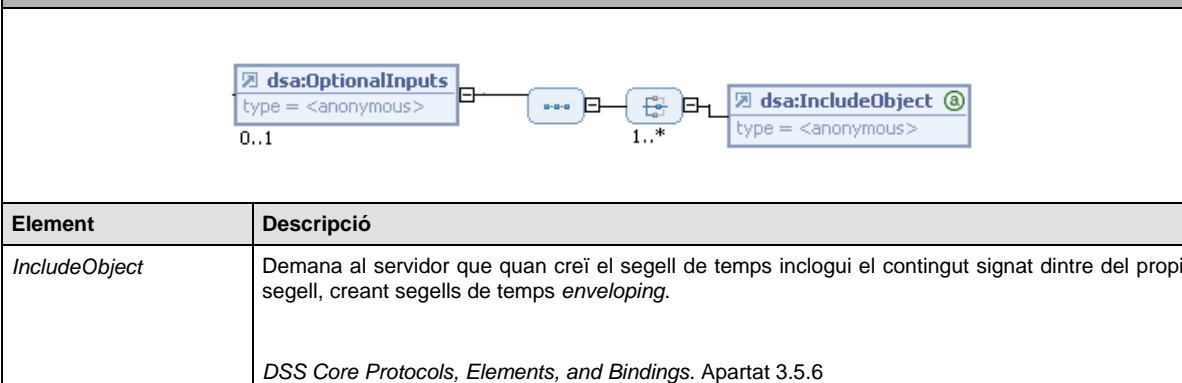
### Estructura de *KeySelector*



Element	Descripció
<i>KeyInfo</i>	Determina la clau que ha de fer servir el servidor per a crear el segell de temps ja que el certificat proporcionat indica al servidor quina és la identitat que ha d'assumir a l'hora de signar el segell.  <i>XSS Profile of the OASIS DSS. Apartat 4.13</i>

Aquest *OptionalInput* només aplica al cas XML i XAdES.

### Estructura de *IncludeObject*



Element	Descripció
<i>IncludeObject</i>	Demana al servidor que quan creï el segell de temps inclogui el contingut signat dintre del propi segell, creant segells de temps <i>enveloping</i> .  <i>DSS Core Protocols, Elements, and Bindings. Apartat 3.5.6</i>

### Missatge de sortida

El missatge de sortida ens retorna a part d'un codi que ens informa de si el procés ha anat de manera correcta o no (veure codis de retorn DSS per a més informació) un

*SignatureObject* idèntic al present a les peticions de validació però que conté el segell de temps generat.

Aquí trobarem un element *Signature* si el segell es XML o XAdES i un *timestamp* codificat en Base64 en cas de que sigui CMS.

**Resposta de creació d'un segell de temps**

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <dss:SignResponse Profile="urn:oasis:names:tc:dss:1.0:profiles:timestamping"
      RequestID="I9e54e5e59e9724683b2379f846ec0f98"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:Result>
        <dss:ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</dss:ResultMajor>
        <dss:ResultMessage xml:lang="en">Signature created</dss:ResultMessage>
      </dss:Result>
      <dss:OptionalOutputs/>
      <dss:SignatureObject>
        <dss:Timestamp>
          <ds:Signature Id="id-35628298-7ff4-4b3d-99f2-a90b66852f5e"
            xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <ds:SignedInfo Id="id-edf7e577-0201-4d47-aced-5345661e474a">
              <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 Id="id-6b24e8c2-8b5a-4378-85f4-aelb7c66e723">
                <ds:DigestMethod
                  Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                <ds:DigestValue>2jmmj7l5rSw0yVb/vlWAYkK/YBwk=</ds:DigestValue>
              </ds:Reference>
              <ds:Reference URI="#TSTInfo-id-a58ac98d-106e-41fb-af7a-
4be4779d5a02" Id="id-172cal37-8939-4af6-8636-13a9961832af"
                Type="urn:oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken">
                <ds:Transforms>
                  <ds:Transform
                    Algorithm="http://www.w3.org/2001/10/xml-exc-
c14n#"/>
                </ds:Transforms>
                <ds:DigestMethod
                  Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                <ds:DigestValue>Bfg4btfaYTPHH+WF8476LIq90/o=</ds:DigestValue>
              </ds:Reference>
            </ds:SignedInfo>
            <ds:SignatureValue>
Rju4P8zGqsQw8Yn0HjAnFYT+uVkhUTXe/yzpL4z6YTS3A3mYv4SKmsBLmvt0HGwVRHR3STMiQC2o
qdf5e/KrEDIkF4CG/+PUTJp4M7766mRMR5yrBg7x157F8SNEbwbQsqnblcrF3poW5JcW0ayS2uf
zXS+vU/zZD+kseX3ag0=</ds:SignatureValue>
          <ds:KeyInfo>
            <ds:X509Certificate>
MIIDCCBgigAwIBAgIQbg5vDBUDLv1ELR24xDVsZzANBgk
.../BAQDAgeAMBYGA1UdJQEB/wQMMAoGCCsGAQUFBwMIMB0GA1UdDgQWBBQp0Ii85Ebkb/+WT2f5cqRUu8oo3F4REK
59hnPbxLaVZ/8zDp2afqKqGOAd9e8TCKY3Nx7tOi jnd+ jnEIYe/BTisMTXqx8+o0eYoI9uFX/iMQmms569KsPXGnVd
byuys5LE6iCfee0OVwz9ruKeDwX6f+MQw9mkTguh7vFebCNpyfxIzjbDHXKy1NOeVdd1UYs2tgPhOqHBHXXZLepU8aR
Zx3ixKF7TQaipnQc4PLrKUQPrPkQCMtN73b1RtOfc+dO9C8ZtLIwVeld9vbYAp9n0TvdxbWrHSHvFfkFWjq8HRUD+
ptXHDZaWBxmuUQ==</ds:X509Certificate>
            </ds:X509Certificate>
            <ds:KeyInfo>
              <ds:RSAKeyValue>
                <ds:Modulus>ALMdRzdnoHg90PT4ukAz6VPNL+qDcfOjE7R+1NO8SdlvqeFarXqkYze36dJ3J2ypXHf+vKWF5HsEYy
jfsCkPRyil6CWYoqOfyhycVr1X3gAmsFSlQmiIZsrbuE3cXR+4I2ZIxGlvqbpSVRp0NmF090W5s4OofWbbemGSldpK
pA8v</ds:Modulus>
                <ds:Exponent>AQAB</ds:Exponent>
              </ds:RSAKeyValue>
            </ds:KeyInfo>
          </ds:SignatureObject>
        </dss:Timestamp>
      </dss:Result>
    </dss:SignResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

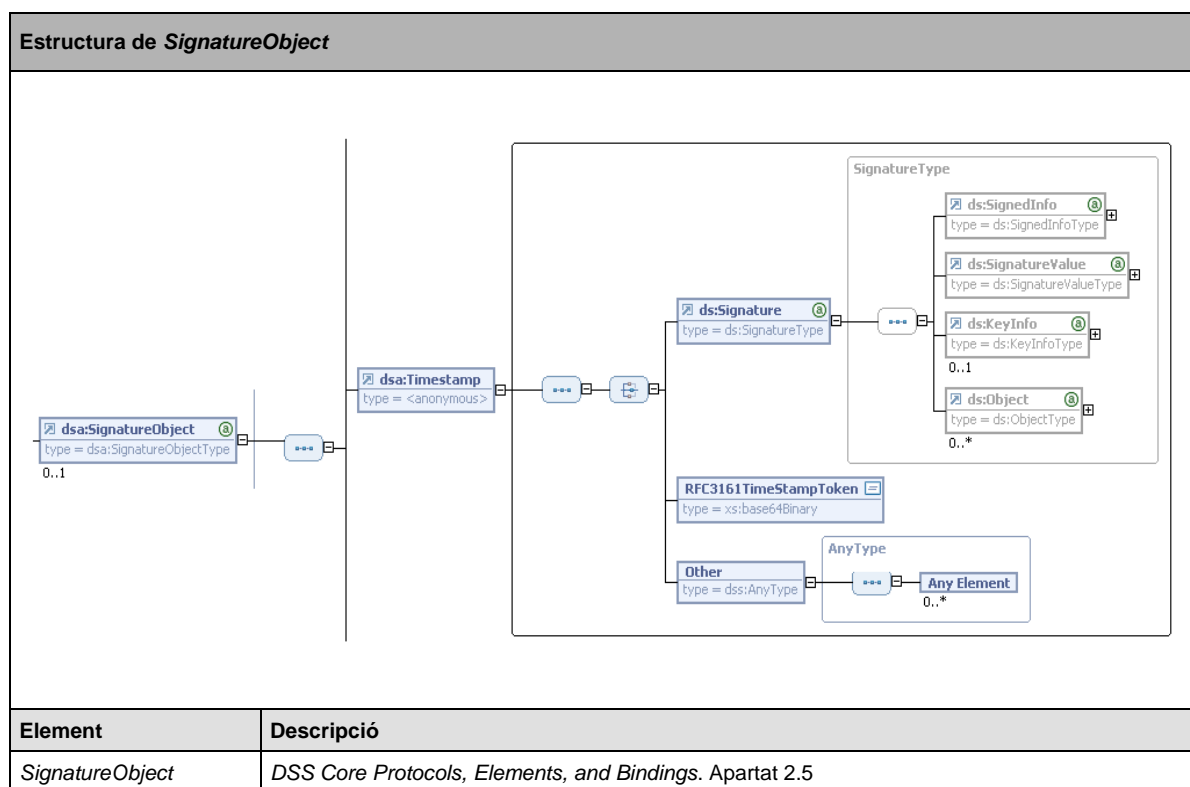
```

MimeType="application/xml">
  <dss:TstInfo>
    <dss:SerialNumber>487517772120598484027575696654783309170220764526</dss:SerialNumber>
    <dss:CreationTime>2006-08-
08T09:29:11.515Z</dss:CreationTime>
    <dss:Policy>urn:oid:0.4.0.2023.1.1</dss:Policy>
    <dss:ErrorBound>PT1S</dss:ErrorBound>
    <dss:Ordered>true</dss:Ordered>
    <dss:TSA
      Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:X509SubjectName"
      >CN=Servei de segellat de temps de PSIS,OU=Jerarquia
Entitats de
Certificacio Catalanes,OU=Vegeu
https://www.catcert.net/verCIT-1
      (c)05,OU=Serveis Publics de Certificacio CIT-
1,O=Agencia
      Catalana de Certificacio (NIF Q-0801176-
I),C=ES</dss:TSA>
    </dss:TstInfo>
  </ds:Object>
</ds:Signature>
</dss:Timestamp>
</dss:SignatureObject>
</dss:SignResponse>
</soapenv:Body>
</soapenv:Envelope>

```

Figura 18 Missatge de resposta de creació d'un segell de temps

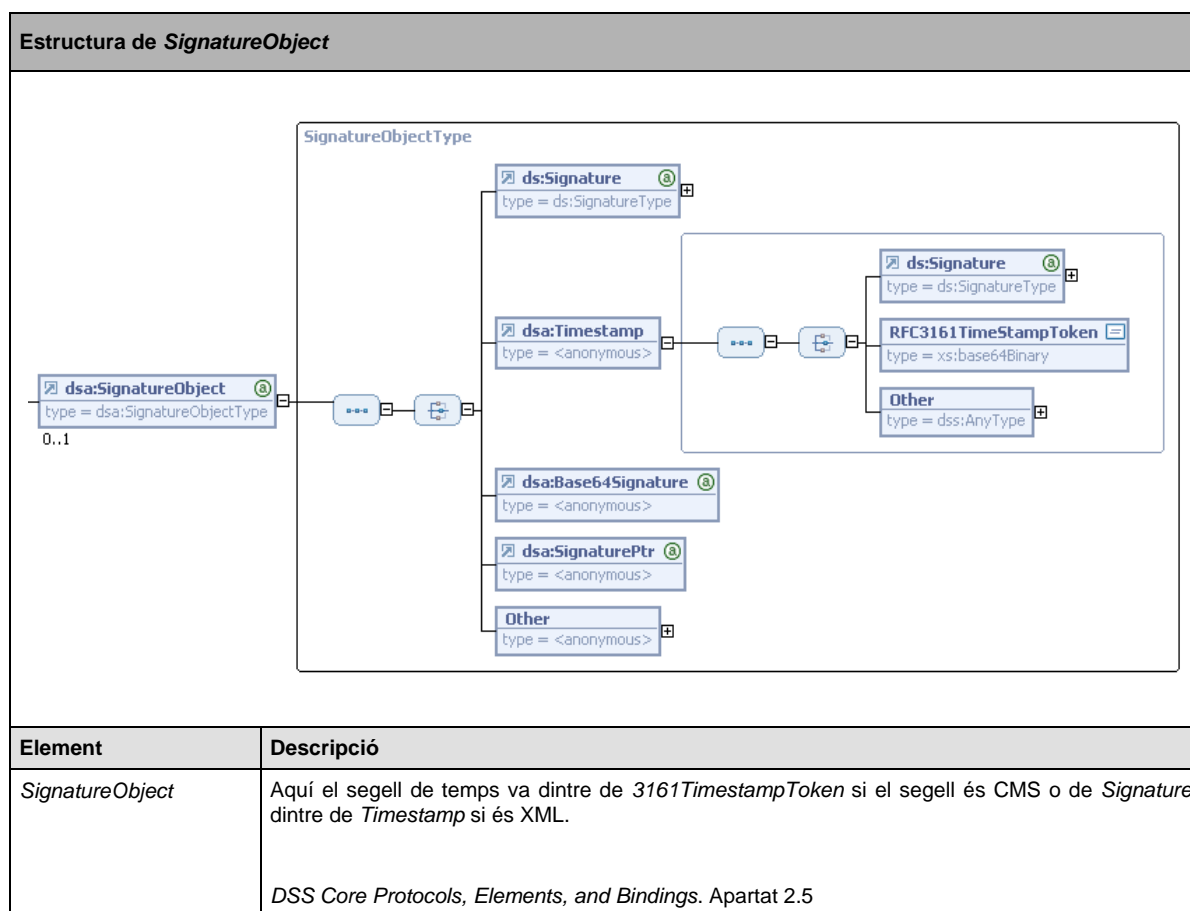
Així els segells de temps resultants aniran dintre de l'estructura *SignatureObject* següent.



## 5.6 Validació de segells de temps

Aquesta funcionalitat permet fer la validació d'un segell de temps. És molt similar a la validació de signatures (donat que un segell de temps no és més que una signatura amb constància de l'instant temporal) i la seva missatgeria és gairebé idèntica.

L'únic aspecte en el qual mostren discrepàncies importants és que el segell de temps va dintre de l'element *Timestamp* del *SignatureObject* i no dintre del *SignatureObject* en sí.



En el següent exemple es descriu la missatgeria:

### Missatge d'entrada

Aquí mostrem un exemple de validació de segell de temps XML.

## Validació d'un segell de temps en format XML

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <dss:VerifyRequest Profile="urn:oasis:names:tc:dss:1.0:profiles:timestamping"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:InputDocuments>
        <dss:DocumentHash ID="Doc1">
          <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
            xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
          <ds:DigestValue xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
            >2jmj715rSw0yVb/vlWAYkK/YBwk=</ds:DigestValue>
        </dss:DocumentHash>
      </dss:InputDocuments>
      <dss:SignatureObject>
        <dss:Timestamp>
          <ds:Signature Id="id-43d445db-c7b8-474b-8671-59cfd1f1d4b6"
            xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <ds:SignedInfo Id="id-1481a051-8df1-43ce-b5f3-2651a485b60d">
              <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="#TSTInfo-id-53785d35-b3d4-4053-b8e4-
015420b7a652"
                Id="id-9be94625-1874-4e12-8a4e-66644e54bc31"
                Type="urn:oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken">
                <ds:Transforms>
                  <ds:Transform
                    Algorithm="http://www.w3.org/2001/10/xml-exc-
c14n#" />
                  </ds:Transforms>
                  <ds:DigestMethod
                    Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                  <ds:DigestValue>Ra+HCOC2g2ET1aPTEqSt9fa2FrY=</ds:DigestValue>
                </ds:Reference>
                <ds:Reference Id="id-edac93f9-1a1d-48e5-a464-8c81b2abb090">
                  <ds:DigestMethod
                    Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                  <ds:DigestValue>2jmj715rSw0yVb/vlWAYkK/YBwk=</ds:DigestValue>
                </ds:Reference>
              </ds:SignedInfo>
              <ds:SignatureValue>
ccb595lHswznoFYkXwwjc4RZ2ozXuVngOgg3ABSixat3yEsxG4oGzbRAeZnE+xeurYnD9RRv1rTE
edAawGn8YQskeUY74ONeyGmdOXMv1fc8n7ssJdThLWEINQtXg/nAksXtALSinQW/BW9OGbmRyXN+
+RA115eflX6sJHqFtPE=</ds:SignatureValue>
              <ds:KeyInfo>
                <ds:X509Data>
                  <ds:X509Certificate>MIIHIDCCBgigAwIBAgIQbg5vDBUDLv1ELR24xDVsZzANBgkqhkiG9w0BAQUFADCB8zELMA
kGA1UEBhMCRVMxOzA5BGNVBAoTMkFnZW5jaWEgQ2F0YWxhbmEgZGUgQ2VydGlmawNhY21vIChOSUYgUS0wODAxMTc2
LUkpMSgwJgYDVQQLEx9TZXJ2ZWlzfIFB1Ymxp...+dO9C8ZtLiWVlEd9vbYAp9n0TvdxbWvRSHSHvFfkFWjq8HRUD+ptX
HDZaWBxmuUQ==</ds:X509Certificate>
                </ds:X509Data>
                <ds:KeyValue>
                  <ds:RSAKeyValue>
                    <ds:Modulus>ALMdrZdnoHg90PT4ukAz6VPNL+qDcfOjE7R+1N08SdlvqeFarXqkYze36dJ3J2ypXHF+vKWF5HsEYy
jfsCkPRyil6CWYqOfyhyCvRlX3gAmsFS1QmiIZsrbuE3cXR+4I2IZxGlVqbpSVRp0NmF090W5s4OofWbbemGSldpK
pA8v</ds:Modulus>
                    <ds:Exponent>AQAB</ds:Exponent>
                  </ds:RSAKeyValue>
                </ds:KeyValue>
              </ds:KeyInfo>
            <ds:Object Id="TSTInfo-id-53785d35-b3d4-4053-b8e4-015420b7a652"
              MimeType="application/xml">
              <dss:TstInfo
xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
                <dss:SerialNumber>70640059021933358515457823778492235485949471232</dss:SerialNumber>
                <dss:CreationTime>2006-04-
12T11:51:36.977Z</dss:CreationTime>
                <dss:Policy>urn:oid:9.9.9.9</dss:Policy>
                <dss:ErrorBound>PT1S</dss:ErrorBound>
              </dss:TstInfo>
            </ds:Object>
          </dss:Timestamp>
        </dss:SignatureObject>
      </dss:VerifyRequest>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

```

                                <dss:Ordered>true</dss:Ordered>
                                <dss:TSA
                                Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:X509SubjectName"
                                >CN=Servei de segellat de temps de PSIS,OU=Jerarquia
Entitats de
                                Certificacio Catalanes,OU=Vegeu
https://www.catcert.net/verCIT-1
                                (c)05,OU=Serveis Publics de Certificacio CIT-
1,O=Agencia
                                Catalana de Certificacio (NIF Q-0801176-
I),C=ES</dss:TSA>
                                </dss:TstInfo>
                                </ds:Object>
                                </ds:Signature>
                                </dss:Timestamp>
                                </dss:SignatureObject>
                                </dss:VerifyRequest>
                                </SOAP-ENV:Body>
                                </SOAP-ENV:Envelope>

```

Figura 19 Missatge de validació d'un segell de temps en format XML

El següent és un exemple de validació de segell de temps CMS:

#### Validació d'un segell de temps en format CMS

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <dss:VerifyRequest Profile="urn:oasis:names:tc:dss:1.0:profiles:timestamping"
xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:OptionalInputs>
        <dss:ReturnProcessingDetails/>
      </dss:OptionalInputs>
      <dss:InputDocuments>
        <dss:DocumentHash ID="Doc1">
          <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" />
          <ds:DigestValue
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">2jmj715rSw0yVb/vlWAYkK/YBwk=</ds:DigestValue
>
        </dss:DocumentHash>
      </dss:InputDocuments>
      <dss:SignatureObject><dss:Timestamp><dss:RFC3161TimeStampToken>MIAGCSqGSib3DQEHAqCAMIACAQM
xCzAJBgUrDgMCGGUAMIAGCyqGSib3DQEJEAEEoIAEggGHMIIbGwIBAQYEgnEJCTAhMAkGBSsOAwIaBQAQEFNo5o+5ea
0sNMLW/75VgGJCv2AcJAhQ2aw5KFUwGIemIkQPis+xpPNQlhRgPMjAwNjA0MTIxMTUzMjZamAKCAQGAQGBAQEBaf+
...
+Q9b10Be20H1p46QxiQ3s+5R+eWNZqCjtRWdwdqa9ITmbOIe0sNalL1E9hRQtTXr4kgmHG55w+ULDTqoK0mW07ABLU
MUEwLtSrVpzRmAAAAAAA</dss:RFC3161TimeStampToken></dss:Timestamp></dss:SignatureObject>
    </dss:VerifyRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Figura 20 Missatge de validació d'un segell de temps en format CMS

Els *OptionalInputs/Outputs* són els mateixos que a les validacions de signatura. Només s'inclouen els que canvien el seu significat.

#### Estructura de *ReturnSigningTime*



Element	Descripció
<i>ReturnSigningTime</i>	Retorna el temps en el qual el segell de temps va ésser estampat.  <i>DSS Core Protocols, Elements, and Bindings. Apartat 4.6.5</i>

### Missatge de sortida

El missatge de sortida es idèntic al retornat pel servidor en el cas de validacions de signatures, així que les mateixes consideracions sobre *Result* i *OptionalOutputs* son vàlides.

#### Resposta d'una validació d'un segell de temps

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <dss:VerifyResponse Profile="urn:oasis:names:tc:dss:1.0:profiles:timestamping"
      xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema">
      <dss:Result>
        <dss:ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</dss:ResultMajor>
        <dss:ResultMinor>urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:onAllDocuments</ds
s:ResultMinor>
      </dss:Result>
      <dss:OptionalOutputs/>
    </dss:VerifyResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Figura 21 Missatge de resposta d'una validació d'un segell de temps

## 5.7 Perfil de Timestamp

Existeix un perfil de DSS anomenat *Timestamp Profile* que dona més detall i és més estricte que no pas ho és el DSS en el seu *core*.

PSIS suporta ambdues maneres de procedir amb el tractament de segells de temps (creació i validació) i per tal de poder accedir a la plataforma PSIS en aquest perfil el client només ha d'afegir l'atribut **Profile="urn:OASIS:names:tc:dss:1.0:profiles:timestamping"** a la **VerifyRequest** en cas de validacions o a la **SignRequest** en cas de peticions.

---

## 6. Requisits previs

---

Tot seguit es descriuen els requisits necessaris per a desenvolupar un client de la plataforma PSIS.

### 6.1 Comunicacions

Cal una connexió a Internet configurada a l'ordinador on es faci el desenvolupament, per a poder:

- Obtenir arxius de compilació WSDL i XSD.
- Obtenir llibreries per al desenvolupament.
- Compilar el client a partir del WSDL.
- Executar el servei de la plataforma PSIS des del client.

Cal verificar que la connexió amb la plataforma PSIS es troba disponible (garantia que el servei està funcionant i no hi ha incidències). Es disposa d'un entorn que es pot accedir de diverses formes. Per a verificar l'accés, només s'han d'introduir les adreces que s'indiquen tot seguit al navegador i verificar la resposta del servidor:

- Entorn d'integració (ó preproducció):
  - <https://psisbeta.catcert.net/psis/catcert-test/dss-secure> (sota connexió segura SSL o TLS)
  - <https://psisbeta.catcert.net/psis/catcert-test/dsspdf> (sota connexió segura SSL o TLS)
- Entorn d'explotació (ó producció):
  - <https://psis.catcert.net/psis/catcert/dss> (sota connexió segura SSL o TLS)
  - <https://psis.catcert.net/psis/catcert/dsspdf> (sota connexió segura SSL o TLS)

Si la connexió és satisfactòria, apareixerà un missatge semblant a la figura següent.

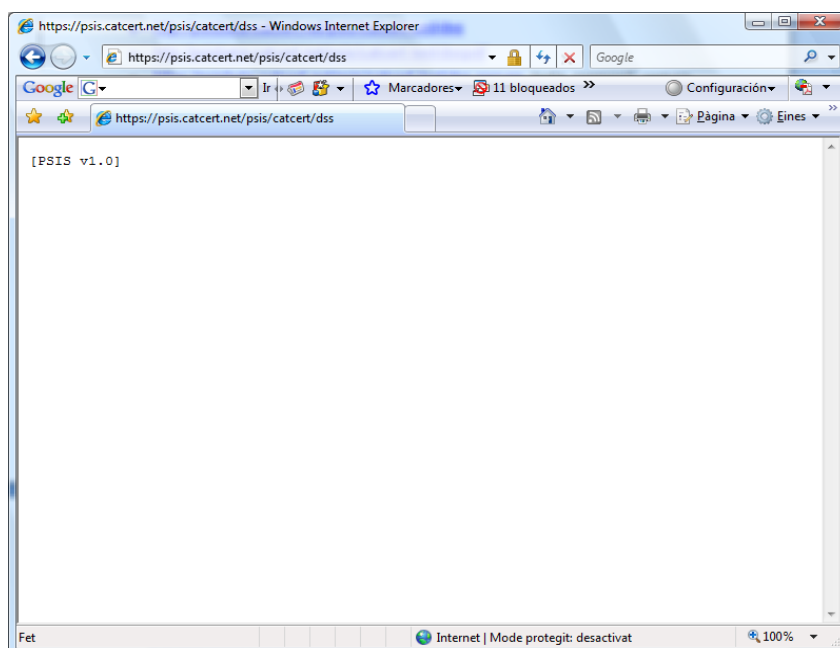


Figura 22 Captura de pantalla amb una connexió correcta a la plataforma PSIS

La plataforma PSIS es troba disponible sota connexió segura (SSL). Això fa que s'hagin de configurar els certificats SSL a l'ordinador client per tal de poder tenir connexió un cop haguem creat el client de la plataforma. En el proper punt, es detalla el procés per configurar el client de PSIS per treballar amb autenticació.

## 6.2 Autenticació

Els serveis que ofereix la plataforma PSIS solament poden ser accedits de forma autenticada, segons s'indica a continuació:

- AVS
  - o Entorn d'integració:
    - Autenticació: <https://psisbeta.catcert.net/psis/catcert-test/dss-secure>
  - o Entorn d'exploació:
    - Autenticació. <https://psis.catcert.net/psis/catcert/dss>
- ASC
  - o Entorn d'integració:
    - Autenticació: <https://psisbeta.catcert.net/psis/catcert-test/dss-secure>
  - o Entorn d'exploació:
    - Autenticació. <https://psis.catcert.net/psis/catcert/dss>
- TSA
  - o Entorn d'integració:
    - Obert: <http://psisbeta.catcert.net/psis/catcert-test/tsp>

- Entorn d'exploració:
  - Obert. <http://psis.catcert.net/psis/catcert/tsp>
- PDF Lite
  - Entorn d'integració:
    - Autenticació: <https://psisbeta.catcert.net/psis/catcert-test/dsspdf>
  - Entorn d'exploració:
    - Autenticació. <https://psis.catcert.net/psis/catcert/dsspdf>

### **Obtenció dels certificats**

En primer lloc, s'han d'obtenir els certificats que participen en l'autenticació. Les pautes a seguir són les següents:

- Descarregar les claus de les entitats de certificació de la jerarquia CATCert. Es poden obtenir de la següent adreça

- [http://www.catcert.net/web/cat/5\\_4\\_descarrega\\_claus.jsp](http://www.catcert.net/web/cat/5_4_descarrega_claus.jsp)

S'han de descarregar els certificats corresponents a les entitats EC-ACC, EC-GENCAT i EC-SAFP.

El certificat que utilitza PSIS per autenticar-se és un certificat d'aplicació (CDA), i que es pot consultar accedint a

- <https://psis.catcert.net/psis/catcert/dss>

- Disposar del certificat de client que s'utilitzarà per autenticar-se davant PSIS. En el pack d'integrador es proporciona un contenidor PKCS12 amb un certificat CDA (d'aplicació) de proves (*psisauth.p12*), només vàlid per autenticar-se davant l'entorn d'integració de PSIS. Quan es vulgui accedir a l'entorn d'exploració, s'haurà d'obtenir un CDA real, sol·licitant-lo a través de la pàgina web de Catcert: [http://www.catcert.net/web/cat/1\\_0\\_cataleg.jsp#g](http://www.catcert.net/web/cat/1_0_cataleg.jsp#g)

#### **6.2.1. Autenticació mitjançant canal SSL**

Consisteix en establir una comunicació xifrada mitjançant el protocol SSL, fent servir autenticació tant de client com de servidor. Quan el client presenti el seu certificat a PSIS,

com a part del procés d'establiment del canal segur, PSIS decideix si accepta o denega la temptativa d'accés.

L'entorn de PSIS que admet aquest tipus de peticions està ubicat a:

- Entorn integració: <https://psisbeta.catcert.net/psis/catcert-test/dss-secure>
- Entorn explotació: <https://psis.catcert.net/psis/catcert/dss>

Es detalla a continuació la configuració de client per realitzar autenticació SSL contra PSIS.

### Client Java

- Enregistrament del certificat de servidor

El magatzem de certificats públics en els quals es confia es denomina *truststore*, i per defecte s'utilitzarà el situat al directori `JAVA_HOME/jre/lib/security/cacerts`

En aquest contenidor de certificats de confiança s'hi han d'incloure els certificats de les arrels de CATCert descarregats anteriorment, que composen la jerarquia de certificació del CDA de PSIS. També existeix l'opció d'enregistrar-hi directament el CDA de PSIS. El procés per instal·lar un certificat en el magatzem es detalla a continuació:

#### NOTA

*El certificat de client és un CDA (certificat d'aplicació) emès per l'entitat SAPF de la jerarquia CATCert. Serveix per atacar l'entorn de Pre-Producció (en el moment que es vulgui realitzar la integració amb la plataforma7 PSIS entorn de Producció, s'haurà d'emetre un certificat vàlid i no de proves).*

## Passos per instal·lar un certificat

Tecnologia	Descripció
Java	<p>Per defecte, la màquina virtual estarà instal·lada a la carpeta:</p> <p><b>%Archivos de programa%\Java\jre_&lt;versió de jvm&gt;\</b></p> <p><b>Des d'una línia de comandes podrà localitzar a la carpeta d'instal·lació de la màquina virtual de Java, la carpeta <code>/jre/lib/security/</code>.</b></p> <p><b>Un cop a la carpeta, cal executar:</b></p> <pre>..\..\bin\keytool -import -file &lt;ruta certificat&gt; -alias &lt;alias&gt; -keystore cacerts</pre> <ul style="list-style-type: none"> <li><i>Ruta certificat = localització al disc dur d'on es troba el certificat a instal·lar</i></li> <li><i>Alias = alias del certificat</i></li> </ul> <p>Per a completar el procés de registre del certificat, caldrà introduir la clau d'accés del magatzem de certificats. Per defecte, aquesta clau serà <b>"changeit"</b>.</p> <p>Finalment, confirmarem tot el procés de registre amb una <b>"y"</b> i el certificat quedarà registrat a la màquina virtual.</p> <p>Es pot comprovar la correcta instal·lació del certificat executant:</p> <pre>..\..\bin\keytool -list -v -keystore cacerts</pre>

Per comprovar que s'han instal·lat correctament tots els certificats, es pot fer ús de la MMC:

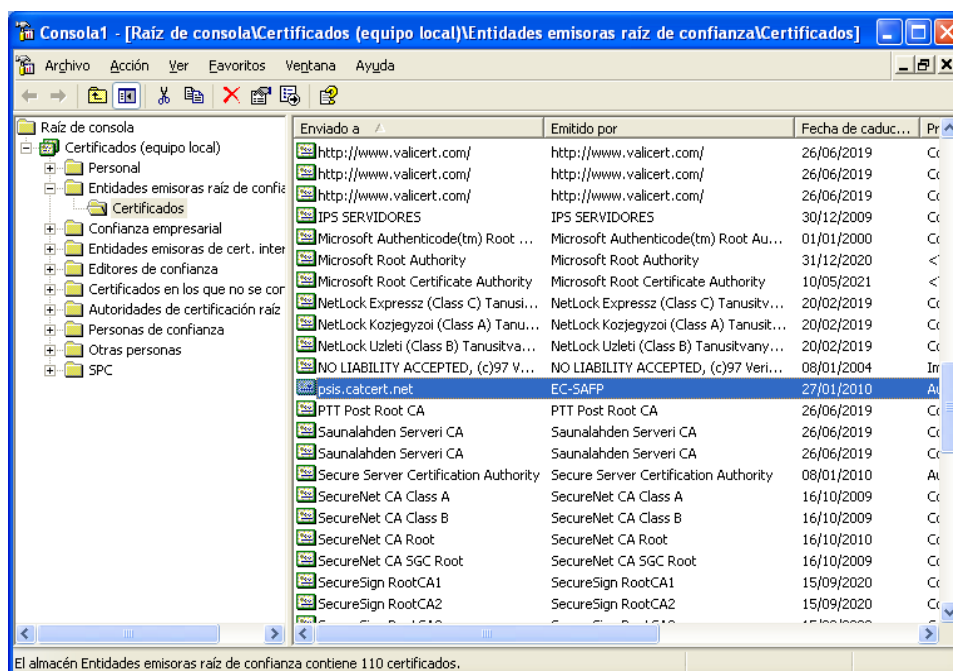


Figura 23 Imatge de la MMC

Per indicar al client Java que s'ha d'utilitzar el magatzem *truststore* com a repositori de certificats de confiança, cal afegir el següent codi:

```
System.setProperty("javax.net.ssl.trustStore","path_to_cacerts");  
System.setProperty("javax.net.ssl.trustStoreType","JKS");  
System.setProperty("javax.net.ssl.trustStorePassord","changeit");
```

- Selecció del certificat de client

Per seleccionar el certificat de client amb el que es farà l'autenticació, només cal afegir el següent codi al client:

```
System.setProperty("javax.net.ssl.keyStore","path_to_pl2");  
System.setProperty("javax.net.ssl.keyStoreType","pkcs12");  
System.setProperty("javax.net.ssl.keyStorePassword","password_pl2");
```

En el cas de fer servir el certificat de proves, el password és "slikZSmz"

Existeix també la possibilitat d'afegir el certificat d'usuari al repositori .keystore, que es troba al directori home de l'usuari. En ell s'insereixen els certificats del propi usuari.

És necessari també afegir el proveïdor de seguretat:

```
System.setProperty("java.protocol.handler.pkgs","com.sun.net.ssl.inte  
rnal.www.protocol");  
Security.addProvider(new com.sun.net.ssl.internal.ssl.Provider());
```

### Client .NET i Visual Basic

És important conèixer que hi ha tres tipus de repositori de certificats:

1. Sistema
2. Usuari
3. Comptes de servei

La decisió d'instalar-ho en un o altre repositori vindrà condicionada per la visibilitat desitjada.

- Certificats instalats en el repositori de sistema: tenen visibilitat per a tots els processos que s'executin en el sistema.
- Certificats instalats en el repositori d'usuari: només està disponible pels processos el propietari dels quals és l'usuari autènticat a la sessió.
- Certificats instalats en el repositori de comptes de servei: és idèntic al repositori d'usuari però s'ha d'indicar quin compte de servei tindrà accés.

La darrera opció cal tenir-la en compte quan es tracti d'aplicacions ASP.NET, les qual s'executen amb el compte de servei d'IIS.

- Enregistrament del certificat de servidor

Tal i com s'ha explicat per al client Java, s'han d'enregistrar al repositori escollit els certificats que conformen la jerarquia del CDA de PSIS, o bé directament el CDA.

Per a instal·lar un certificat en el repositori es pot fer mitjançant l'opció del menú de context de Windows o des de la Consola d'Administració. El menú de context instala per defecte en el repositori de l'usuari autenticat mentre que la Consola permet instal·lar en qualsevol repositori.

Per defecte la Consola d'Administració no ve preconfigurada amb la vista de certificats. Per tal de tenir accés a aquesta vista s'han de seguir les següents passes:

1. Executar la Consola amb el comandament **mmc**
2. Agregar o Quitar complementos (**CTRL-M**)
3. Agregar el complement **Certificats**

El procés d'instal·lació d'un certificat és el següent:

.NET(C#) o Visual Basic 6	<p>Per a accedir a la MMC, cal executar la comanda <b>mmc</b> i fer les següents passes:</p> <ol style="list-style-type: none"> <li>1. Al menú <b>Consola</b>, faci clic en <b>Agregar o quitar complemento</b>.</li> <li>2. Faci clic en <b>Agregar</b>.</li> <li>3. Triï <b>Certificados</b> i, després, fer clic en <b>Agregar</b>.</li> <li>4. Esculli <b>Cuenta de equipo</b> i, després, faci clic en <b>Siguiente</b>.</li> <li>5. Seleccionei <b>Equipo local</b> i després, faci clic en <b>Finalizar</b>.</li> <li>6. Faci clic en <b>Cerrar</b> i a continuació en <b>Aceptar</b>.</li> <li>7. Al panell esquerre, obri <b>Certificados (equipo local)</b>, després, <b>Entidades emisoras de certificados raíz de confianza</b> i finalment <b>certificados</b>.</li> <li>8. Amb el botó secundari, triï <b>Todas las tareas</b> i a continuació <b>Importar</b>.</li> <li>9. Faci clic a <b>Siguiente</b>, introdueixi la ruta i el nom de l'arxiu .cer de la CA.</li> <li>10. Faci clic a <b>Siguiente</b>.</li> <li>11. Seleccionar <b>Colocar todos los certificados en el siguiente almacén</b> i faci clic a <b>Examinar</b>.</li> <li>12. Triï <b>Mostrar almacenes físicos</b>.</li> <li>13. Obri <b>Entidades emisoras de certificados raíz de confianza</b> en la llista i després, triï <b>Equipo local</b>.</li> <li>14. Faci clic a <b>Aceptar</b>, faci clic en <b>Siguiente</b> i a continuació, en <b>Finalizar</b>.</li> </ol>
---------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	<p>15. Facci clic a <b>Aceptar</b> per tancar el missatge de confirmació.</p> <p>16. Refresqui la carpeta <b>Certificados</b> en el MMC i verifiqui que el certificat apareix a la llista.</p> <p>17. Tanqui el MMC.</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### - Selecció del certificat de client

Per seleccionar el certificat de client amb el que es farà l'autenticació, només cal afegir el següent codi al client:

```
using System.Security.Cryptography.X509Certificates;

...

X509Certificate2 p12client = new X509Certificate2(path_to_p12,
password_p12);

...

proxy.Url = "https://psisbeta.catcert.net/psis/catcert-test/dss-secure";
proxy.ClientCertificates.Add(p12client);
```

Com es pot veure, només cal instanciar el contenidor PKCS12 del certificat de client, i indicar al proxy que l'utilitzi per fer l'autenticació. Una altra opció consisteix en enregistrar prèviament el certificat de client al repositori, juntament amb la seva clau privada, i després al codi instanciar només la part pública del certificat:

```
using System.Security.Cryptography.X509Certificates;

...

X509Certificate clientCertificate =
X509Certificate.CreateFromCertFile(path_to_cert);

...

proxy.Url = "https://psisbeta.catcert.net/psis/catcert-test/dss-secure";
proxy.ClientCertificates.Add(clientCertificate);
```

#### NOTA

*X509Certificate2 només està disponible a partir del Framework .NET 2.0 i per tant no existeix en versions anteriors com la 1.1. El Framework .NET 2.0 només es pot usar amb el Visual Studio 2005 però no amb el 2003.*

### 6.2.2. Autenticació mitjançant petició signada

Consisteix en aplicar una signatura *XAdES enveloped* sobre el missatge de petició, realitzada amb el certificat amb el qual el client es vol autenticar. D'aquesta forma, al rebre la petició signada, PSIS validarà la signatura i comprovarà la identitat del signatari validant el certificat.

Per implementar una petició signada, tan sols s'ha de construir un nou `OptionalInput`, anomenat `ClaimedIdentity`, que serà el contenidor de la signatura, i afegir-lo a la petició sense signar. Seguidament es mostra la semàntica del `ClaimedIdentity` (per a més detalls, consultar el [profile DSS](#)):

## OptionalInput &lt;ClaimedIdentity&gt;

```
<dss:ClaimedIdentity>
    <dss:Name Format="urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName">cn=#142543504953522d31205066ed73696361204465206c61205365f1612050727565626173646974,2.5.4.5=#1309383938393030303245,2.5.4.42=#14075066ed73696361,2.5.4.4=#14154465206c61205365f1612050727565626173646974,ou=#140b496e666f726de174696361,ou=serveis public de certificacio cpisr-1,ou=vegeu https://www.catcert.net/vercpisir-1 c(03),o=#141b442e472e20496e666f726de1746963612054726962757461726961,c=es
    </dss:Name>

    <dss:SupportingInfo>
        <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <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="">
                    <ds:Transforms>
                        <ds:Transform
Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
                        <ds:Transform
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#WithComments"/>
                    </ds:Transforms>

                    <ds:DigestMethod
Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>

                    <ds:DigestValue>KSeIPW6DZshRClds18FSmwMr6rI=</ds:Digest Value>

                </ds:Reference>
            </ds:SignedInfo>

            <ds:SignatureValue>SW7NJJaMcDmQijz2L5R//9PwqkTwnbnklL5lqkeqOFjQIWwoof0lKvWX3soERayk675clAzluzE6/mbdvIvQpsCAldlrIpFSCDWpfbgclPPWNggG+Zt/JiG4LqC518zxgtEje9bESGPsfUfeI7MMMyLu9ht7nLvY9T5AhYktbWc=</ds:SignatureValue>
            <ds:KeyInfo>
                <ds:X509Data>

                <ds:X509Certificate>MIIHxTCBcq2gAwIBAgIQOiZLLuOHRFCGO+XhWb/RJANBgkqhkiG9w0BAQUFADCCASYxCzAJBgNVBAYTAkVTMTswQQYDVQQKEzJBZ2Vuy2lhIENhdGFsYW5hIGRlIElnlc nRpZmljYWNpbAoTk1GIFFETMDgwMTE3Ni1JKTEOMDIGAUEBxMrUGFc2F0Z2UgZGUgbGEgQ29uY2VWy2lvIDExIDA4MDA4IEJhcmlNlbG9uYTEuMCwGA1UECxMU2VydmVpcyBQdWJsawNZ IGRl IElnlc nRpZmljYWNpbAoTk1GIFFETMDgwMTE3Ni1JKTEOMDIGAUEBxMtVmVnZXUgaHR0cHM6Ly93d3cuY2F0Y2VydcS5uZXQvd mVvyQ0LDLTigIchjKTAAzMswKgYDVQQLEYNBZGlpbmllzdHJhY2lvbnMGtgT9jYXxzIGRl IElnhdG FsdW55YTEOMA wGA1UEAxMFRTUMtQUwwHHcNMDDUwMjIyMTUyODIzWHcNMDDkwMjIyMTUyNzQyYjCCA SoxCzAJBgNVBAYTAkVTMSQwIgYDVQQKFBEtELkcuiEluZm9yeFE0aWNhIFRYaWJldGFyaWEEXnzAl BgNVBAS TLlZl2VlIgh0dhNBziOIsvd3d3LmNhndGNlc QubmV0L3Zlc kNQSVNSLTegYYgwMYkxMDA uBgNVBAS TJlNlc nZlaXMgUH VibGl jcyBKZSBSDZXJOaWZpy2FjaW8gQlBJUlItMT EUMBIGA1UE CxQLSW5mb3Jt4XRpy2EXhjAcBgNVBAQUFUURlIGxhIFNl18WEgUHVJlZWJhc2RpdDEQM A4GA1UEKhQH U Gbtcl2lj YTESMBAGA1UEBRMJODk4OTAwMDJFM S4wLAYDVQQDFCVDUelTUio xIFBm7XNP y2EGRGU g bGequ2XxYSBOc nVlymFzzGL0MIGFMAOGCSsgSi b3DOEBQAUA A4GNADC BiQBkQC5 jXuFd X3ta qQQ
```

```
mW15VLzJiOI63izbOdKxSFPvexkAlwdffgOvp4YwllVpZ9GsMMyjOsGAUW7dR/riexlJCigR78L
qy5rUYjNhk7bqYapCHCf2ysLB/2ns8JrwIbdIDXUAGTOJEa/SI3kQAHHJluzsFxn9L803Y3IpCf
4whVz4MxwIDAQABo4IDaTCCA2UwOAYDVRORBDEwL4EVZm9saXZlcmFzQGhNdGNlcnQubmV0pBYw
FDESMBAGAlUEBRMJUzI4MjYwMjRIMA4GA1UdDwEB/wQEAWIHgDADBgNVHSEUfjAUBggrBgEFBQc
DAgYIKwYBBQUHAWQwEQYJYIZIAAyb4QGEBBAQDAgWgMB0GA1UdGgQWBbT4uGQK3i6hgBSq7zt3iZ
kWzL8mlTCCATEGA1UdIwSCASgwgEkgBRM7I1JlCsCA5rQSDAKS2u9MXqmNKGb+aSB9jCB8zELM
AkGA1UEBhMCRVMxOzA5BgNVBAoTMkFnZW5jaWEgQ2F0YWxhbmEgZGUgQ2VydG1maWNhY2lviChO
SUYgUS0wODAxMTc2LUkpmSGwJgYDVQQLEx9TXZJ2ZWlzfIFB1YmxyY3MgZGUgQ2VydG1maWNhY2l
vMTUwMwYyYDVQLEYxWZwldSBodHRwc2ovL3d3dy5jYXRjZXJ0Lm5ldC92ZXJhcnJlbCAoYykwMz
ElMDMGA1UECXM5SmVYXjYxdWlhIEVudG10YXRzIGRlIENlcnRpZmljYWNpbyBDYXRhbGfuZXMx
dZANBgNVBAMTBkVudlUFDQ4IQPZfTkWQ5Yio+HE2mvtFzDjCB4AYDVROgBIHYMIHVMIHSGsrBgEE
AfV4AQMAUTCBwJArBggrBgEFBQcCARYfaHR0cHM6Ly93d3cuY2F0Y2VydC5uZXRvdmVyVEVTVDC
BkgYIKwYBBQUHAgIwYUagYjBcXVl3c3Qg6XMgdW4gY2VydG1maWNhDbKzSBUrVNUIHBlcnNvb
FIHJlY29uZWdhdkBkXCdYdGVudG1maWNhY2nzTgkvc2lnbmF0dXJhIGRlIGNSYXNzSZAuL1BwZ
WldSBodHRwc2ovL3d3dy5jYXRjZXJ0Lm5ldC92ZXJURVNUMDQGCCsGAQUFBwEBBCCgwJjAkBggr
BgEFBQcwAYYYaHR0cHM6Ly9vY3NwLmNhdGNlcnQubmV0MBGCCCsGAQUFBwEDBAwwCjAIBgYEA15
GAQEWAYDVROfBFkwVzBVFOgUYyMAHR0cDovL2Vwc2NkLmNhdGNlcnQubmV0L2NybC9lYy1hbC
5jcmyGj2h0dHA6Ly9lchNjZDIuY2F0Y2VydC5uZXRvY3JsL2VlWFsLmNybDANBgkqhkiG9w0BA
QUFAAOCAQEALaJaN0zwICOR8nb7amKsuK02b8rRP8/AX0fSaQRD/VdSlddy26C0s784Ab5MmOT
mR7eJ/BznC9/LWwm2yKKkHMqbaQKFDNpPEg4Z4VQGg09gNFRycWONUcmc6mWGETk+6ZeYfg+5t
PrFwa2pijnV26872el9bJFD6ELTFu0l8J0qjwM/m8ZlVgNkNvgN2pagaWE3WALgZdhQdhd6dWb2IY
vECBmW6qjJAiGi3iI7GhlXK60xOy28TCBWGxkAzhxvY3v0l2jVlWfSLmNybDANBgkqhkiG9w0BA
RRERlQ8C16YpGaAOCdryPlCCZkBKwAhMOuPFdxhV/Hj9bLyUjgUQ=
```

Aquí es mostra un exemple de com signar una petició. Aquesta versió tan sols serveix per signar peticions de creació de signatures. Pròximament s'inclouran les versions per signar peticions d'arxivat o Compound.

## MessageSigner

```

public static void main(String[] args) throws Exception{
    ...

    //Missatge SOAP sense signar
    InputStream requestToBeSigned =
        ClassLoader.getResourceAsStream(requestToBeSignedPath);

    //Signar la petició
    InputStream signedRequest = signRequest(requestToBeSigned);

    //Creació del missatge
    SignRequestDocument requestDocument =
        SignRequestDocument.Factory.parse(extractBody(new
        String(getBytesFromStream(signedRequest))));

    // Execució del servei
    SignResponseDocument responseDocument = port.sign(requestDocument);

    ...
}

public static InputStream signRequest(InputStream toBeSigned)
    throws KeyStoreException, IOException, NoSuchAlgorithmException,
        CertificateException, UnrecoverableKeyException, XmlException,
        ParserConfigurationException {

    KeyStore keyStore = KeyStore.getInstance("PKCS12");
    FileInputStream fileInputStream = new FileInputStream(p12Path);
    keyStore.load(fileInputStream, p12Pass.toCharArray());

```

```
String alias = keyStore.aliases().nextElement();
X509Certificate certificate = (X509Certificate) keyStore.getCertificate(alias);
PrivateKey key = (PrivateKey) keyStore.getKey(alias,
params.getProperty("keyStorePassword").toCharArray());

InputStream resourceSupport = MessageSigner.signRequest(toBeSigned, certificate, key);

return resourceSupport;
}

public static byte[] getBytesFromStream(InputStream is) throws IOException {
    ...
}

//Extreu les dades del missatge d'usuari
public static String extractBody(String message) {
    ...
}
```

### Nota

El codi en Java per implementar la signatura de peticions s'adjunta en el paquet d'integració.

## 6.3 Software

Abans de posar-se a desenvolupar el seu client, ha de conèixer quins són els requisits de *software* necessaris per al llenguatge de programació que utilitzarà.

A la taula següent es detallen aquests requisits i més endavant podrà veure algun exemple d'ús.

Tecnologia	Software necessari	Descripció / documentació / descàrrega
Java	JDK v1.5 o superior	Paquet bàsic de desenvolupament Java. <ul style="list-style-type: none"> <li><a href="http://java.sun.com/javase/downloads/index.jsp">http://java.sun.com/javase/downloads/index.jsp</a></li> </ul>
	Ant 1.6.2 (Si s'utilitza Eclipse, no és necessari descarregar-ho).	Llibreria per a la creació de <i>scripts</i> fent ús d'arxius de tipus XML. <ul style="list-style-type: none"> <li><a href="http://ant.apache.org/manual/install.html">http://ant.apache.org/manual/install.html</a></li> <li><a href="http://ant.apache.org/bindownload.cgi">http://ant.apache.org/bindownload.cgi</a></li> </ul>
	XFire 1.2.6	Framework per a la connexió i configuració de serveis web. <ul style="list-style-type: none"> <li><a href="http://repository.codehaus.org/org/codehaus/xfire/xfire-distribution/1.2.6/xfire-distribution-1.2.6.zip">http://repository.codehaus.org/org/codehaus/xfire/xfire-distribution/1.2.6/xfire-distribution-1.2.6.zip</a></li> </ul>
C# (.NET)	Microsoft (R) .NET Framework 1.1.4322.573 o superior (Per a poder compilar és necessari l'SDK)	Paquet bàsic per a desenvolupar i utilitzar aplicacions .NET. <ul style="list-style-type: none"> <li><a href="http://msdn.microsoft.com/netframework/downloads/updates/default.aspx">http://msdn.microsoft.com/netframework/downloads/updates/default.aspx</a></li> </ul>

Visual Basic 6	<i>Microsoft Visual Basic 6 + Service Pack 6</i>	Paquet bàsic per a desenvolupar i utilitzar aplicacions Visual Basic 6.  Els desenvolupaments es poden distribuir mitjançant instal·lables que copien a l'ordinador de l'usuari totes les llibreries i arxius necessaris per a la seva execució.
----------------	--------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 6.4 WSDL

Els serveis que ofereix la plataforma PSIS s'han desenvolupat facilitant l'ús d'una definició coneguda tècnicament amb el nom WSDL (*Web Service Definition Language*).

WSDL defineix tots els mètodes, interfícies i elements necessaris per què el programador desenvolupi el codi del seu client d'una forma autònoma i gairebé automàtica si s'utilitzen eines de compilació de WSDL.

Es pot descarregar l'arxiu WSDL de la plataforma PSIS des de les següents URLs:

- Integració:
  - o <http://psisbeta.catcert.net/wsdl/dss-pre.wsdl>
  - o <https://psisbeta.catcert.net/wsdl/dss-pre.wsdl> (sota connexió segura SSL o TLS)
- Producció:
  - o <http://psis.catcert.net/wsdl/dss.wsdl>
  - o <https://psis.catcert.net/wsdl/dss.wsdl> (sota connexió segura SSL o TLS)

## 7. Creació del client

A continuació, es detallarà el procés de creació del client de la plataforma PSIS en els següents llenguatges de programació:

- Java
- .NET (C#)
- Visual Basic 6

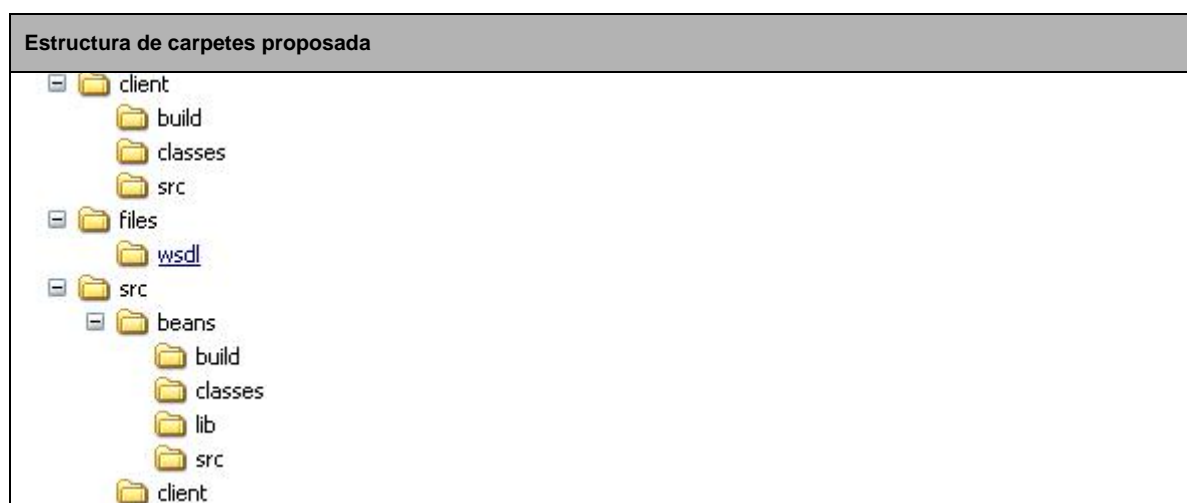
### 7.1 Java

Pautes a seguir per a la creació del client Java.

#### 1. Preparació

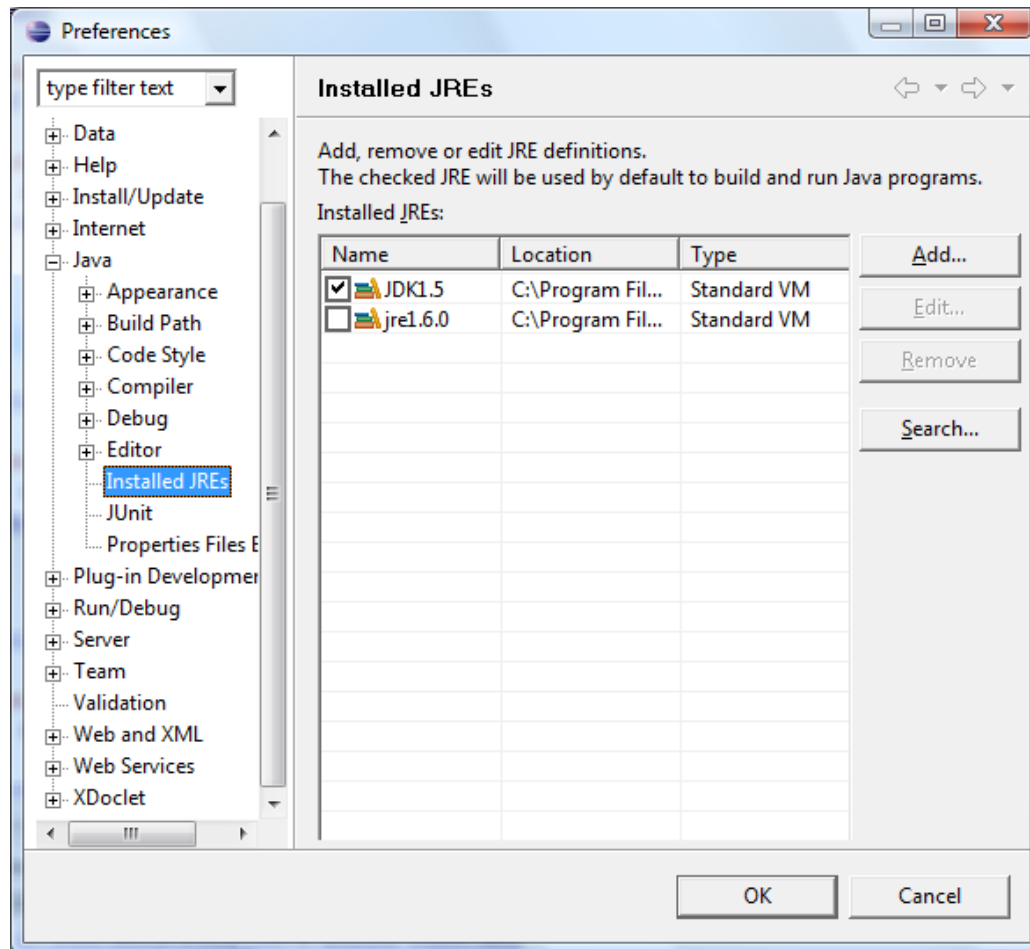
Per a facilitar el procés de creació del client ens ajudarem de les eines que proporciona *Ant* (eina per realitzar tasques automàtiques i repetitives normalment durant la fase de compilació i generació de codi font).

Per a configurar *Ant* proposem una estructura de carpetes, que caldrà adaptar si per necessitats del client es vol modificar.



Realitzar les següents accions:

- Copiar les llibreries contingudes dins del directori /lib del paquet Xfire (XFire 1.2.6 ) al directori src/beans/lib.
- Ubicar el fitxer xfire-all-1.2.6.jar, també del paquet Xfire, al directori src/beans/lib.
- Ubicar el fitxer dss.wsdl (extret de la URL indicada al punt 6.4) al director files/wsd
- Seleccionar el compilador de JDK (*javac.exe*) enlloc del JRE.



- Si es treballa amb Eclipse, cal que verifiquem que tenim la versió correcta de JDK afegida a les preferències. S'hi pot accedir a través del menú superior: Window > Preferences.
- Dins la finestra que s'obre, s'accedeix a Java > Installed JREs, i tenint en compte que s'hagi instal·lat prèviament el JDK 1.5, només cal que l'afegim si no hi és present. Add...

## 2. Generació

Es procedeix a la generació de les llibreries que utilitzarà el client per construir missatges de petició a PSIS i processar les respostes.

En primer lloc, s'ha de crear el següent arxiu d'*Ant*, que anomenarem **build-libs.xml**, i que s'ubicarà sota el directori arrel del projecte:

### build-libs.xml

```
<project name="PSIS_LIBS" default="default" basedir=". ">

    <!-- Definició de les carpetes per a poder generar els beans amb XmlBeans -->
    <property name="beans.dir" value="${basedir}/src/beans" />
    <property name="beans.lib" value="${beans.dir}/lib" />
    <property name="beans.src" value="${beans.dir}/src" />
    <property name="beans.classes" value="${beans.dir}/classes" />
    <property name="beans.build" value="${beans.dir}/build" />

    <!-- Definició de les carpetes per a poder generar el client amb Xfire -->
    <property name="client.dir" value="${basedir}/src/client" />

    <!-- Definició de l'arxiu WSDL -->
    <property name="wsdl.location" value="${basedir}/files/wsdl/dss.wsdl" />

    <!-- Definició de llibreries necessaries per a compilar els beans -->
    <path id="xmlbeans.classpath">
        <fileset dir="${beans.lib}">
            <include name="xbean-2.2.0.jar" />
            <include name="jsr173_api_1.0.jar" />
        </fileset>
    </path>

    <!-- Definició de llibreries necessaries per a compilar el wsdl -->
    <path id="wsdlgenerator.classpath">
        <fileset dir="${beans.lib}">
            <include name="*.jar" />
        </fileset>
        <fileset dir="${beans.build}">
            <include name="*.jar" />
        </fileset>
    </path>

    <!-- Definició del compilador de XMLBeans -->
    <taskdef name="scomp" classname="org.apache.xmlbeans.impl.tool.XMLBean"
classpathref="xmlbeans.classpath">
    </taskdef>

    <!-- Definició del compilador de XFire -->
    <taskdef name="wsген" classname="org.codehaus.xfire.gen.WsGenTask"
classpathref="wsdlgenerator.classpath" />

    <!-- Tasca de compilació dels beans -->
    <target name="build-beans">
        <scomp schema="${wsdl.location}" destfile="${beans.build}/psis-beans.jar"
download="true" classpathref="xmlbeans.classpath" classgendir="${beans.classes}"
srcgendir="${beans.src}" />
    </target>

    <!-- Tasca de compilació del wsdl -->
    <target name="compile-wsdl">
        <wsген outputDirectory="${client.dir}/src" wsdl="${wsdl.location}" overwrite="true"
binding="xmlbeans" />
    </target>

    <!-- Tasca global -->
    <target name="default" depends="build-beans,compile-wsdl" />
</project>
```

```
</project>
```

Figura 24 Contingut del fitxer ant per generar el client Java de PSIS fent servir el fitxer WSDL

#### Nota

Aquest fitxer està subjecte a les versions de XMLBeans i Xfire a utilitzar. En tot cas, reviseu que coincideixin amb les vostres en el punt...

```
<path id="xmlbeans.classpath">
  <fileset dir="${beans.lib}">
    <include name="xbean-2.2.0.jar" />
    <include name="jsr173_api_1.0.jar" />
  </fileset>
</path>
```

o adapteu el fitxer.

Com es pot veure, en aquest arxiu està descrit l'arbre de directoris proposat anteriorment. En el cas de que es vulgui utilitzar una estructura diferent, també s'haurà de modificar aquest document amb les noves carpetes. Cal comprovar també que les versions de les llibreries utilitzades coincideixen amb les descrites al build-libs.xml.

Un cop creat l'arxiu anterior, es procedirà a la seva execució mitjançant *Ant*. Aquesta execució té de dues tasques principals que s'han de realitzar en un ordre concret (en el cas d'utilitzar Eclipse, la compilació del build-libs.xml mitjançant Ant inclou les dues tasques, realitzades automàticament de forma consecutiva: target name="default").

La primera tasca (**build-beans**) que cal fer ens permetrà generar les classes Java que posteriorment ens ajudaran a construir els missatges d'enviament i recepció de la plataforma PSIS. Per a executar-la, escriurem la següent comanda:

#### Sentència

```
ant -buildfile build-libs.xml build-beans
```

I obtindrem aquest resultat:

#### Log de sortida per pantalla

```
Buildfile: build-libs.xml

build-beans:
[scomp] Time to build schema type system: 1.632 seconds
[scomp] Time to generate code: 2.033 seconds
[scomp] Compiling 496 source files to D:\PSIS\src\beans\classes
[scomp] Note: * uses or overrides a deprecated API.
[scomp] Note: Recompile with -Xlint:deprecation for details.
[scomp] Time to compile code: 10.856 seconds
[scomp] Building jar: D:\PSIS\src\beans\build\psis-beans.jar

BUILD SUCCESSFUL
Total time: 49 seconds
```

Observarem que, com a resultat, obtenim el fitxer **psis-beans.jar** sota el directori `src/beans/build/`. Aquest paquet conforma tot el gruix de classes que permet al client interactuar amb PSIS, i s'ha d'afegir al *classpath* del projecte..

La segona tasca (**compile-wsdl**) genera les classes Java que implementen el client de la plataforma PSIS. Per a fer-ho escriurem la següent comanda:

### Sentència

```
ant -buildfile build-libs.xml compile-wsdl
```

I obtindrem aquest resultat:

### Log de sortida per pantalla

```
Buildfile: build-libs.xml

compile-wsdl:
[wsge] log4j:WARN No appenders could be found for logger
(org.codehaus.xfire.gen.Wsdl11Generator).
[wsge] log4j:WARN Please initialize the log4j system properly.
[wsge] Retrieving schema at 'http://www.w3.org/TR/2002/REC-xmlsig-core-
20020212/xmlsig-core-schema.xsd', relative to 'file:/D:/PSIS/files/wsdl/dss.wsdl'.
[wsge] Retrieving schema at 'http://www.w3.org/TR/xmlsig-core/xmlsig-core-
schema.xsd', relative to 'file:/D:/PSIS/files/wsdl/dss.wsdl'.
[wsge] Retrieving schema at 'http://www.w3.org/2001/xml.xsd', relative to
'file:/D:/PSIS/files/wsdl/dss.wsdl'.
[wsge] Retrieving schema at 'http://docs.oasis-open.org/security/saml/v2.0/saml-schema-
assertion-2.0.xsd', relative to 'file:/D:/PSIS/files/wsdl/dss.wsdl'.
[wsge] Retrieving schema at 'http://www.w3.org/TR/2002/REC-xmlsig-core-
20020212/xmlsig-core-schema.xsd', relative to 'http://docs.oasis-
open.org/security/saml/v2.0/saml-schema-assertion-2.0.xsd'.
[wsge] Retrieving schema at 'http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/xenc-
schema.xsd', relative to 'http://docs.oasis-open.org/security/saml/v2.0/saml-schema-
assertion-2.0.xsd'.
[wsge] Retrieving schema at 'http://www.w3.org/TR/2002/REC-xmlsig-core-
20020212/xmlsig-core-schema.xsd', relative to 'http://www.w3.org/TR/2002/REC-xmlenc-core-
20021210/xenc-schema.xsd'.
[wsge] Retrieving schema at 'http://www.w3.org/2001/xml.xsd', relative to
'file:/D:/PSIS/files/wsdl/dss.wsdl'.
[wsge] Retrieving schema at 'http://www.oasis-
open.org/committees/download.php/3408/oasis-sstc-saml-schema-protocol-1.1.xsd', relative to
'file:/D:/PSIS/files/wsdl/dss.wsdl'.
[wsge] Retrieving schema at 'http://www.w3.org/TR/xmlsig-core/xmlsig-core-
schema.xsd', relative to 'http://www.oasis-open.org/committees/download.php/3408/oasis-sstc-
saml-schema-protocol-1.1.xsd'.
[wsge] Retrieving schema at 'http://www.w3.org/TR/xmlsig-core/xmlsig-core-
schema.xsd', relative to 'file:/D:/PSIS/files/wsdl/dss.wsdl'.

[wsge] oasis\names\tc\dss\_1_0\core\wsdl\SOAPport.java

[wsge] oasis\names\tc\dss\_1_0\core\wsdl\digitalSignatureServiceClient.java
[wsge] oasis\names\tc\dss\_1_0\core\wsdl\digitalSignatureServiceImpl.java
BUILD SUCCESSFUL
Total time: 22 seconds
```

El resultat és un nou paquet que conforma el client de la plataforma PSIS amb les següents classes:

Packet	Classe	Descripció
<b><i>oasis.names.tc.dss_1_0.core.wsd.</i></b>	<code>digitalSignatureServiceClient</code>	Factoria de clients de la plataforma PSIS
	<code>digitalSignatureServiceImpl</code>	Implementació de la interfície SOAPPort
	<code>SOAPport</code>	Interfície de definició de mètodes de la plataforma PSIS

## 4. Compilació

Per a facilitar l'ús del client, s'empaquetarà el codi font del client dintre d'un arxiu JAR que posteriorment s'inclourà en el del *classpath* del nostre projecte Java. El resultat serà un fitxer ***psis-client.jar***, ubicat sota *src/client/build*.

Per a fer-ho, configurarem l'arxiu *Ant* següent, anomenat ***build-client.xml***, ubicat sota el directori arrel.

### build-client.xml

```
<project name="PSIS_CLIENT" default="default" basedir=". ">

    <!-- Definició de les carpetes per poder generar els beans amb XmlBeans -->
    <property name="beans.dir" value="${basedir}/src/beans" />
    <property name="beans.lib" value="${beans.dir}/lib" />
    <property name="beans.build" value="${beans.dir}/build" />

    <!-- Definició de directoris per poder generar el client amb Xfire -->
    <property name="client.dir" value="${basedir}/src/client" />
    <property name="client.lib" value="${client.dir}/lib" />
    <property name="client.src" value="${client.dir}/src" />
    <property name="client.classes" value="${client.dir}/classes" />
    <property name="client.build" value="${client.dir}/build" />

    <!-- Definició de llibreries necessàries per compilar el client -->
    <path id="client.classpath">
        <fileset dir="${beans.lib}">
            <include name="*.jar" />
        </fileset>
        <fileset dir="${beans.build}">
            <include name="*.jar" />
        </fileset>
    </path>

    <!-- Tasca de compilació del client -->
    <target name="build-client">
        <mkdir dir="${client.classes}" />
        <mkdir dir="${client.build}" />
        <javac srcdir="${client.dir}/src" destdir="${client.dir}/classes"
includes="**/*.java" classpathref="client.classpath" failonerror="true" debug="true"
source="1.5" />
        <jar destfile="${client.build}/psis-client.jar" basedir="${client.dir}/classes" />
    </target>

    <!-- Tasca global -->
    <target name="default" depends="build-client" />

</project>
```

Figura 25 Contingut del fitxer ant per compilar el client Java generat a partir del fitxer WSDL

I executarem la següent comanda:

Sentència
ant -buildfile build-client.xml build-client

Obtenint aquest resultat:

Log de sortida per pantalla
Buildfile: build-client.xml  build-client: [javac] Compiling 3 source files to D:\PSIS\src\client\classes [javac] Note: D:\PSIS\src\client\src\wsdl\core\_0\_1\dss\tc\names\oasis\digitalSignatureServiceClient.java uses unchecked or unsafe operations. [javac] Note: Recompile with -Xlint:unchecked for details. [jar] Building jar: D:\PSIS\src\client\build\psis-client.jar  BUILD SUCCESSFUL Total time: 2 seconds

Recordar que s'ha d'afegir aquest paquet al *classpath* del projecte.

Si no tenim seleccionada o instal·lada correctament la versió del JRE (JDK 1.5 o superior), l'Eclipse ens llençarà un error d'aquest tipus:

Problems Javadoc Declaration Console
<terminated> PSIS_Java build-libs.xml [Ant Build] C:\Program Files\Java\jre1.6.0\bin\javaw.exe (21/01/2008 17:08:46) Buildfile: C:\Projects\PSIS_Java\build-libs.xml build-beans: [scomp] Time to build schema type system: 7.51 seconds [scomp] Time to generate code: 22.999 seconds [scomp] Compiling 844 source files to C:\Projects\PSIS_Java\src\beans\classes  BUILD FAILED C:\Projects\PSIS_Java\build-libs.xml:51: Error running javac.exe compiler  Total time: 32 seconds

## 5. Utilització

El següent exemple de codi fa una connexió a la plataforma PSIS i envia un missatge de validació buit, per comprovar la correcta integració.

**Exemple de creació de la connexió amb la plataforma PSIS**

```
public static void main(String args[]) throws Exception {  
  
    // Inicialització del client  
    digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();  
  
    // Indicar el servidor  
    SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-  
test/dss");  
  
    // Composició del missatge  
    VerifyRequestDocument requestDocument1 = VerifyRequestDocument.Factory.newInstance();  
  
    // Codi de composició del missatge  
    ...  
  
    // Execució del servei per verificar  
    VerifyResponseDocument responseDocument1 = port.verify(requestDocument1);  
  
    // Codi de processat de la resposta  
    ...  
}
```

**Figura 26 Exemple de creació de la connexió amb la plataforma PSIS en Java**

Al paquet d'integrador es proporciona un joc de proves més extens, per testejar les funcionalitats bàsiques de PSIS.

Caldrà canviar el package de les classes java d'acord a l'estructura montada segons cada implementació.

## 7.2 .NET (C#)

Pautes a seguir per a la creació del client .NET (C#).

### 1. Generació

Per a poder generar totes les classes necessàries per a fer la connexió a la plataforma PSIS s'ha d'executar la següent instrucció que generarà el codi font del client en llenguatge C#: La utilitat wsdl.exe només es compatible amb el fitxer WSDL de PSIS en la seva versió inclosa al framework .net 1.1. Per a *clients integrant fent servir .net 2.0 el fitxer WSDL s'ha de compilar fent servir .net 1.1* i el fitxer generat si que es pot compilar fent servir .net 2.0 .

**Sentència**

```
wsdl /language:cs  
/namespace:net.catcert.psis
```

/out:PSISClient.cs <http://psisbeta.catcert.net/wsdl/dss-pre.wsdl>

**NOTA**

El procés de generació dels stubs fent servir l'eina wsdl.exe genera una sèrie de WARNINGS durant el procés. Aquests són totalment normals i derivats de la natura de certs dels missatges involucrats en el procés de prestació de servei de PSIS. Si no apareix cap ERROR i només apareixen WARNINGS el procés es pot considerar totalment correcte.

**NOTA**

S'ha indicat en el procés de generació dels stubs es fa servir la referencia al servidor psisbeta. Es pot generar el client directament des de l'entorn de producció indicant la URL <http://psis.catcert.net/wsdl/dss.wsdl> i mantenint la resta de paràmetres.

**NOTA**

La generació del client a partir del WSDL proporcionat obliga a realitzar alguns canvis posteriors. Existeix la possibilitat d'obtenir directament el fitxer PSISClient.cs que s'inclou dins del paquet d'integració.

### 3. Compilació

Un cop es té el codi font generat, cal fer-ne la compilació. Aquesta compilació s'utilitzarà per a crear la llibreria (DLL) que posteriorment farem servir des d'un nou projecte de C# (.NET) o VB6.

El procés de compilació varia en funció de la màquina virtual de .NET que es tingui instal·lada a l'ordinador.

#### .NET SDK 1.1

```
sn      -k PSISClient.snk
csc     /out:PSISClient.netmodule /target:module *.cs
al      /out:PSISClient.dll PSISClient.netmodule /keyFile:PSISClient.snk
```

#### .NET SDK 2.0

```
sn      -k PSISClient.snk
csc     /out:PSISClient.dll /t:library *.cs /keyfile:PSISClient.snk
```

#### NOTA

Al directori on es realitzin aquestes operacions han d'ubicar-se prèviament PSISClient.cs obtingut prèviament, i el Utils.cs proporcionat amb el paquet d'integració.

### 4. Utilització

El següent exemple de codi fa la connexió a la plataforma PSIS i envia un missatge buit. Posteriorment es veuran exemples per a poder compondre els missatges que s'enviaran.

#### Exemple de creació de la connexió amb la plataforma PSIS

```
private void main()
{
    // Inicialització del client
    digitalSignatureService proxy = new digitalSignatureService();

    // Indicar el servidor
    proxy.Url = http://psisbeta.catcert.net/catcert-test/dss;

    VerifyRequest requestDocument1 = new VerifyRequest();

    // Codi de composició del missatge
    ...

    // Execució del servei per a verificar
    VerifyResponse responseDocument1 = proxy.verify (requestDocument1);
```

```
// Codi de processament de la resposta  
...  
}
```

**Figura 27 Exemple de creació de la connexió amb la plataforma PSIS en .net**

## 7.3 Visual Basic 6

El desenvolupament d'un client en Visual Basic 6 pot ser costós degut al poc suport que es disposa per a aquest llenguatge d'operacions amb *web services*. Per a facilitar el desenvolupament, farem servir un objecte de .NET dins de Visual Basic 6 a mode de referència.

Per tant, els usuaris de Visual Basic han de seguir primer els passos de generació del client fent ús del .NET explicats en l'apartat previ.

Aquí detallarem com s'ha de registrar i desregistrar (en cas d'error) la llibreria generada en .NET i poder-ne fer ús des de Visual Basic 6.

### NOTA

En el procés d'integració fent servir Visual Basic 6 s'han descrit cassos de mal funcionament dels clients generats. Aquests problemes venen derivats d'un estat inestable en el registre de sistema Windows on es desenvolupa la integració. Es recomana fer ús d'eines de registre de Windows en cas de que aquests problemes es presentin.

### 1. Registre

En funció de la versió de .NET que es tingui instal·lada a l'ordinador on es faci el desenvolupament, cal fer el registre de la llibreria fent servir un procés diferent de registre per a cada cas.

Els passos recollits a continuació són per la compilació i registre de la DLL per fer servir en el VBasic en funció del .NET que s'hagi fet servir.

Aquest procés descrit registra la DLL que ha d'haver estat generada fent servir .net (amb casuístiques diferents per a llibreries compilades i generades amb .net 1.1, 2.0 o bé una combinació de les mateixes) i que es registra a Windows per tal de que els clients escrits en Visual Basic 6 en pugin fer ús.

La idea del procés és crear un embolcall en forma d'objecte COM sobre els objectes .net per tal de que Visual Basic 6 en pugin fer ús.

### .NET SDK 1.1

#### Requisits:

La llibreria PSISClient.dll s'ha hagut de generar fent servir els passos descrits amb anterioritat en aquesta documentació.

#### Comanda:

```
regasm /tlb:PSISClient.tlb PSISClient.dll
gacutil /i PSISClient.dll
tlbexp PSISClient.dll
```

### .NET SDK 2.0 (registre d'una llibreria generada amb .NET SDK 1.1)

Requisits:	La llibreria PSISClient.dll i l'arxiu PSISClient.netmodule s'han hagut de generar fent servir els passos descrits amb anterioritat en aquesta documentació.
Comanda:	<pre>regasm /tlb:PSISClient.tlb PSISClient.dll gacutil /i PSISClient.dll</pre>

## 2. Desregistrar

### .NET SDK 1.1 o 2.0

Requisits:	La llibreria PSISClient.dll s'ha hagut d'instal·lar prèviament fent servir els passos indicats en els punts anteriors.
Comanda:	<pre>regasm /u PSISClient.dll</pre>

### NOTA

S'adjunta el fitxer PSISClient.dll en el paquet d'integració, per si es volen obviar tots aquests passos.

### 3. Utilització

El següent exemple de codi fa la connexió a la plataforma PSIS i envia un missatge buit. Posteriorment es veuran exemples per a poder compondre els missatges que s'enviaran.

#### Exemple de creació de la connexió amb la plataforma PSIS

```
Private Sub main()  
    ' Inicialització del client  
    Dim proxy As New PSISClient.digitalSignatureService  
  
    proxy.url="http://psisbeta.catcert.net/psis/catcert-test/dss"  
  
    ' Composició del missatge  
    Dim requestDocument1 As PSISClient.VerifyRequest  
    Dim responseDocument1 As PSISClient.VerifyResponse  
  
    ' Codi de composició del missatge  
    ...  
  
    ' Execució del servei per verificar  
    Set responseDocument1 = proxy.verify(requestDocument1)  
  
    ' Codi de processat de la resposta  
    ...  
  
End Sub
```

Figura 28 Exemple de creació de la connexió amb la plataforma PSIS en .net

El següent exemple de codi realitza també la connexió a la plataforma PSIS i envia un missatge buit amb optional inputs, utilitzant la versió VB 9.0.

#### Exemple de creació de la connexió amb la plataforma PSIS amb VB 9.0

```
Private Sub main()  
    'Utilitats  
    Dim Utils As New net.catcert.psis.Utils  
    Dim proxy As net.catcert.psis.digitalSignatureService  
  
    proxy = New Net.catcert.psis.digitalSignatureService  
    'Necessitem autenticació?  
    ' proxy.Url = "https://psisbeta.catcert.net/psis/catcert-test/dss-secure" o bé  
    ' proxy.Url = "https://psis.catcert.net/psis/catcert/dss"  
    proxy.Url = "https://psisbeta.catcert.net/psis/catcert-test/dss-secure"  
  
    'Indiquem un timeout de 20 segons'  
    proxy.Timeout = "20000"  
  
    Dim p12client As New X509Certificate2("psisauth.p12", "sIikZSmz")  
    proxy.ClientCertificates.Add(p12client)  
  
    'Composició del missatge  
    'Preparant el SignatureOption
```

```
'En funció de la codificació triem base64 o binari
'certsObj(0) = Utils.Base64File(rutaFitxer)
'certsObj(0) = Local.BinariFile(rutaFitxer)
Dim certsObj(1)

certsObj(0) = Local.BinariFile(rutaFitxer)

Dim types(1) As Integer
types(0) = Net.catcert.psis.ItemsChoiceType.X509Certificate

Dim certificate As New Net.catcert.psis.X509DataType
certificate.Items = certsObj
certificate.ItemsElementName = types

Dim other As New Net.catcert.psis.SignatureObjectTypeOther
other.X509Data = certificate

Dim signatureType As New Net.catcert.psis.SignatureObjectType
signatureType.Item = other

'Anem a preparar els Optional Inputs
Dim OptInputsNew As New Net.catcert.psis.OptionalInputs

'Creació del missatge DSS
Dim requestDocument As New Net.catcert.psis.VerifyRequest
requestDocument.SignatureObject = signatureType
requestDocument.OptionalInputs = PreparaOptionals(OptInputsNew)
requestDocument.Profile = "urn:oasis:names:tc:dss:1.0:profiles:XSS"

'Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument))

'Execució del servei
Dim responseDocument As Net.catcert.psis.VerifyResponse
responseDocument = proxy.verify(requestDocument)

'Visualització de la resposta del primer optional output demanat
Presenta_Solucio(responseDocument)

End Sub
```

**Figura 298b Exemple de creació de la connexió amb la plataforma PSIS en .net (VB 9.0)**

## 8. Creació de la missatgeria

Exemples d'ús dels clients amb els tres llenguatges de programació que s'han utilitzat prèviament per a crear-los.

En aquests exemples s'utilitza un paquet anomenat *Utils* que no forma part del protocol dss ni hauria d'utilitzar-se com a punt de partida en un desenvolupament, però ens servirà d'ajuda per no repetir codis senzills en aquest document com poden ser llegir de disc.

Per tant quan es decideixi provar aquests exemples d'integració es pot optar per implementar aquestes senzilles funcions, o directament es poden demanar aquest codis per utilitzar-los en aquests exemples.

Els exemples desenvolupen les següents funcionalitats:

- Validació de certificats
- Validació de signatures en format PKCS#7 / CMS
- Validació signatures XMLDsig
- Validació signatures XAdES
- Validació de signatures PDF
- Creació de segells de temps
- Validació de segells de temps
- Validació de certificats amb autenticació de client SSL

### 8.1 Java

#### Validació de certificats

```
import java.util.Hashtable;

import org.apache.xmlbeans.XmlBase64Binary;
import org.apache.xmlbeans.XmlOptions;

import org.w3.x2000.x09.xmlldsig.X509DataDocument;
import org.w3.x2000.x09.xmlldsig.X509DataType;

import oasis.names.tc.dss._1_0.core.wsdl.SOAPport;
import oasis.names.tc.dss._1_0.core.wsdl.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.OptionalInputsDocument;
import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType.Other;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument.VerifyRequest;
```

```
public class ValidacioCertificat {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        // Inicialització del client
        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-test/dss");

        // Composició del missatge

        // Certificat que es verificarà
        byte[] certificate = Utils.readBase64File("certificate.dat");

        // Definició dels namespaces correctes
        Hashtable prefixes = new Hashtable();
        prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
        prefixes.put("http://www.w3.org/2000/09/xmldsig#", "ds");

        XmlOptions options = new XmlOptions();
        options.setSaveSuggestedPrefixes(prefixes);

        // Creació del missatge DSS
        VerifyRequestDocument requestDocument =
        VerifyRequestDocument.Factory.newInstance(options);

        VerifyRequest request = requestDocument.addNewVerifyRequest();
        request.setProfile("urn:oasis:names:tc:dss:1.0:profiles:XSS");

        // Creació de l'element amb el certificat a verificar
        SignatureObjectType signature = request.addNewSignatureObject();

        X509DataDocument x509doc = X509DataDocument.Factory.newInstance();
        X509DataType x509data = x509doc.addNewX509Data();

        XmlBase64Binary b64certificate = x509data.addNewX509Certificate();
        b64certificate.setByteArrayValue(certificate);

        Other any = signature.addNewOther();
        any.set(x509doc);

        signature.setOther(any);

        // Creació de l'element amb els paràmetres opcionals a consultar
        OptionalInputs optional = request.addNewOptionalInputs();
        optional.addNewReturnProcessingDetails();

        // Consulta de l'atribut X509 SubjectDN
        ReturnX509CertificateInfo info = optional.addNewReturnX509CertificateInfo();

        AttributeType SubjectDNcommonName = info.addNewAttributeDesignator();
        SubjectDNcommonName.setName("urn:oasis:names:tc:dss:1.0:profiles:XSS:certificateAttributes:SubjectDistinguishedName:commonName");

        // Execució del servei per verificar
        VerifyResponseDocument responseDocument = port.verify(requestDocument);

        // Visualització de la petició
        System.out.println(requestDocument.xmlText(options));

        // visualització de la resposta
        System.out.println(responseDocument.xmlText(options));

    }

}
```

Figura 30 Exemple en Java de validació de certificats

### Validació de signatures PKCS#7 / CMS

```
import java.util.Hashtable;

import org.apache.xmlbeans.XmlOptions;

import oasis.names.tc.dss._1_0.core.wsdl.SOAPport;
import oasis.names.tc.dss._1_0.core.wsdl.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.DocumentType;
import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.Base64DataDocument.Base64Data;
import x0CoreSchema.oasisNamesTcDss1.Base64SignatureDocument.Base64Signature;
import x0CoreSchema.oasisNamesTcDss1.InputDocumentsDocument.InputDocuments;
import x0CoreSchema.oasisNamesTcDss1.OptionalInputsDocument.OptionalInputs;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument.VerifyRequest;

public class ValidacioSignaturaCMS {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        // Inicialització del client
        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-test/dss");

        // Composició dle missatge de petició de validació de signatura PKCS#7/CMS
        // de tipus detached document amb el document complet

        // Signatura a verificar
        byte[] signature = Utils.readBase64File("cms-signature.dat");

        // Document en B64 que s'ha signat
        byte[] doc = Utils.readBase64File("cms-doc.dat");

        // Tipus de signatura
        String type = "urn:ietf:rfc:3852";

        // Definició dels namespaces correctes
        Hashtable prefixes = new Hashtable();
        prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
        prefixes.put("http://www.w3.org/2000/09/xmlsig#", "ds");

        XmlOptions options = new XmlOptions();
        options.setSaveSuggestedPrefixes(prefixes);

        // Creació del missatge DSS
        VerifyRequestDocument requestDocument =
        VerifyRequestDocument.Factory.newInstance(options);

        VerifyRequest request = requestDocument.addNewVerifyRequest();

        // Creació de l'element amb la signatura CMS a verificar
        SignatureObjectType signatureType = request.addNewSignatureObject();

        Base64Signature b64signature = Base64Signature.Factory.newInstance();
        b64signature.setByteArrayValue(signature);
        b64signature.setType(type);

        signatureType.setBase64Signature(b64signature);

        // Creació de l'element amb els documents a enviar
        InputDocuments inpDocuments = request.addNewInputDocuments();

        Base64Data b64data = Base64Data.Factory.newInstance();
        b64data.setByteArrayValue(doc);
```

```

DocumentType document = inpDocuments.addNewDocument();
document.setBase64Data(b64data);

// Creació de l'element amb els paràmetres opcionals a consultar
OptionalInputs optional = request.addNewOptionalInputs();
optional.addNewReturnProcessingDetails();

// Execució del servei per verificar
VerifyResponseDocument responseDocument = port.verify(requestDocument);

// Visualització de la petició
System.out.println(requestDocument.xmlText(options));

// visualització de la resposta
System.out.println(responseDocument.xmlText(options));

}
}

```

**Figura 31 Exemple en Java de validació de signatura CMS**

#### Validació de signatures XMLDsig

```

import java.util.Hashtable;

import org.apache.xmlbeans.XmlOptions;

import oasis.names.tc.dss._1_0.core.wsdl.SOAPport;
import oasis.names.tc.dss._1_0.core.wsdl.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.OptionalInputsDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument.VerifyRequest;

public class ValidacioSignaturaXMLDsig {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        // Inicialització del client
        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-test/dss");

        // Composició del missatge

        // Validació d'una signatura xml enveloping
        String signature = Utils.readXmlFile("xmldsig-signature.dat");

        // Definició dels namespaces correctes
        Hashtable prefixes = new Hashtable();
        prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
        prefixes.put("http://www.w3.org/2000/09/xmldsig#", "ds");

        XmlOptions options = new XmlOptions();
        options.setSaveSuggestedPrefixes(prefixes);

        // Creació del missatge DSS
        VerifyRequestDocument requestDocument =
        VerifyRequestDocument.Factory.newInstance(options);

        VerifyRequest request = requestDocument.addNewVerifyRequest();

        // Creació de l'element amb la signatura XMLDsig
        SignatureObjectType signaturetype = SignatureObjectType.Factory.parse(signature,
        options);
    }
}

```

```
request.setSignatureObject(signaturetype);

// Creació de l'element amb els paràmetres opcionals a consultar
OptionalInputs optInputs = OptionalInputs.Factory.newInstance(options);
optInputs.addNewReturnProcessingDetails();

request.setOptionalInputs(optInputs);

// Execució del servei per verificar
VerifyResponseDocument responseDocument = port.verify(requestDocument);

// Visualització de la petició
System.out.println(requestDocument.xmlText(options));

// visualització de la resposta
System.out.println(responseDocument.xmlText(options));
}
}
```

**Figura 32 Exemple en Java de validació de signatura XML**

#### Validació de signatures XAdES

```
import java.util.Hashtable;
import org.apache.xmlbeans.XmlOptions;

import oasis.names.tc.dss._1_0.core.wsdl.SOAPport;
import oasis.names.tc.dss._1_0.core.wsdl.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument.VerifyRequest;

public class ValidacioSignaturaXAdES {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        // Inicialització del client
        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-test/dss");

        // Composició del missatge

        // Validació d'una signatura xades enveloping
        String signature = Utils.readXmlFile("xades-signature.dat");

        // Definició dels namespaces correctes
        Hashtable prefixes = new Hashtable();
        prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
        prefixes.put("http://www.w3.org/2000/09/xmldsig#", "ds");

        XmlOptions options = new XmlOptions();
        options.setSaveSuggestedPrefixes(prefixes);

        // Creació del missatge DSS
        VerifyRequestDocument requestDocument =
        VerifyRequestDocument.Factory.newInstance(options);

        VerifyRequest request = requestDocument.addNewVerifyRequest();

        // Creació de l'element amb la signatura XAdES
        SignatureObjectType signaturetype = SignatureObjectType.Factory.parse(signature,
```

```
options);

request.setSignatureObject(signaturetype);

// Execució del servei per verificar
VerifyResponseDocument responseDocument = port.verify(requestDocument);

// Visualització de la petició
System.out.println(requestDocument.xmlText(options));

// visualització de la resposta
System.out.println(responseDocument.xmlText(options));
}
}
```

Figura 33 Exemple en Java de validació de signatura XAdES

#### Validació de documents PDF signats

```
import java.util.Hashtable;
import org.apache.xmlbeans.XmlOptions;

import oasis.names.tc.dss._1_0.core.wsdl.SOAPport;
import oasis.names.tc.dss._1_0.core.wsdl.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.DocumentType;
import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.Base64DataDocument.Base64Data;
import x0CoreSchema.oasisNamesTcDss1.InputDocumentsDocument.InputDocuments;
import x0CoreSchema.oasisNamesTcDss1.OptionalInputsDocument.OptionalInputs;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument.VerifyRequest;

public class ValidacioPDF {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        //
        // !!! IMPORTANT !!!
        //
        // Inicialització del client (ATENCIO A LA URL: http://.../dsspdf)
        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-
test/dsspdf");

        // Composició del missatge de petició de validació de signatura PDF

        // Document en B64 que s'ha signat
        byte[] doc = Utils.readBase64File("doc.pdf");

        // Definició dels namespaces correctes
        Hashtable prefixes = new Hashtable();
        prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
        prefixes.put("http://www.w3.org/2000/09/xmldsig#", "ds");

        XmlOptions options = new XmlOptions();
        options.setSaveSuggestedPrefixes(prefixes);

        // Creació del missatge DSS
        VerifyRequestDocument requestDocument =
        VerifyRequestDocument.Factory.newInstance(options);

        VerifyRequest request = requestDocument.addNewVerifyRequest();

        //
```

```
// !!! IMPORTANT !!!
//
// Activar el profile PDF
request.setProfile("urn:oasis:names:tc:dss:1.0:profiles:DSS_PDF");

// Creació de l'element amb els documents a enviar
InputDocuments inpDocuments = request.addNewInputDocuments();

Base64Data b64data = Base64Data.Factory.newInstance();
b64data.setByteArrayValue(doc);

DocumentType document = inpDocuments.addNewDocument();
document.setBase64Data(b64data);

// Creació de l'element amb els paràmetres opcionals a consultar
OptionalInputs optional = request.addNewOptionalInputs();

// optional.addSignatureReason();

// Execució del servei per verificar
VerifyResponseDocument responseDocument = port.verify(requestDocument);

// Visualització de la petició
System.out.println(requestDocument.xmlText(options));

// visualització de la resposta
System.out.println(responseDocument.xmlText(options));
}
}
```

**Figura 34 Exemple en Java de validació de documents PDF signats**

#### Creació de segells de temps

```
import java.util.Hashtable;

import org.apache.xmlbeans.XmlAnyURI;
import org.apache.xmlbeans.XmlBase64Binary;
import org.apache.xmlbeans.XmlOptions;

import org.w3.x2000.x09.xmlldsig.DigestMethodType;
import org.w3.x2000.x09.xmlldsig.KeyInfoType;
import org.w3.x2000.x09.xmlldsig.X509DataType;

import oasis.names.tc.dss._1_0.core.wsd1.SOAport;
import oasis.names.tc.dss._1_0.core.wsd1.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.SignRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.SignResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.DocumentHashDocument.DocumentHash;
import x0CoreSchema.oasisNamesTcDss1.IncludeObjectDocument.IncludeObject;
import x0CoreSchema.oasisNamesTcDss1.InputDocumentsDocument.InputDocuments;
import x0CoreSchema.oasisNamesTcDss1.KeySelectorDocument.KeySelector;
import x0CoreSchema.oasisNamesTcDss1.OptionalInputsDocument.OptionalInputs;
import x0CoreSchema.oasisNamesTcDss1.SignRequestDocument.SignRequest;

public class CreacioSegellDeTemps {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        // Inicialització del client
        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-test/dss");

        // Composició del missatge
```

```
// Certificat amb què es generarà la signatura o timestamp
byte[] certificate = Utils.readBase64File("timestamp-certificate.dat");

// Digest
byte[] digest = Utils.readBase64File("timestamp-digest1.dat");

// Tipus de signatura disponibles
// TimeStamp amb signatura CMS/CADES:
// urn:ietf:rfc:3161
// TimeStamp amb signatura XMLDsig:
// oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken
// TimeStamp amb signatura XAdES:
// oasis:names:tc:dss:1.0:core:schema:XAdESTimeStampToken
String type = "oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken";

// Definició dels namespaces correctes
Hashtable prefixes = new Hashtable();
prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
prefixes.put("http://www.w3.org/2000/09/xmldsig#", "ds");

XmlOptions options = new XmlOptions();
options.setSaveSuggestedPrefixes(prefixes);

// Creació del missatge DSS
SignRequestDocument requestDocument = SignRequestDocument.Factory.newInstance(options);

SignRequest request = requestDocument.addNewSignRequest();
request.setProfile("urn:oasis:names:tc:dss:1.0:profiles:timestamping");

// Creació de l'element amb el document per a generar el timestamp
InputDocuments inpDocuments = InputDocuments.Factory.newInstance();

DocumentHash document = inpDocuments.addNewDocumentHash();
document.setID("Doc1");
document.setDigestValue(digest);

DigestMethodType digestMethod = document.addNewDigestMethod();
digestMethod.setAlgorithm("http://www.w3.org/2000/09/xmldsig#sha1");

request.setInputDocuments(inpDocuments);

// Creació de l'element amb els paràmetres opcionals necessaris per a
// generar el timestamp
OptionalInputs optional = request.addNewOptionalInputs();

KeySelector selector = optional.addNewKeySelector();

KeyInfoType key = selector.addNewKeyInfo();

X509DataType x509data = key.addNewX509Data();

// S'afegeix el certificat de TSA de CATCert en Base64
XmlBase64Binary b64certificate = x509data.addNewX509Certificate();
b64certificate.setByteArrayValue(certificate);

XmlAnyURI any = optional.addNewSignatureType();
any.setStringValue(type);

IncludeObject object = optional.addNewIncludeObject();
object.setObjId("Doc1");
object.setWhichDocument("Doc1");
object.setHasObjectTagsAndAttributesSet(false);
object.setCreateReference(true);

// Execució del servei per verificar
SignResponseDocument responseDocument = port.sign(requestDocument);

// Visualització de la petició
System.out.println(requestDocument.xmlText(options));

// visualització de la resposta
System.out.println(responseDocument.xmlText(options));
}
```

Figura 35 Exemple en Java de creació de segell de temps

**Validació de segells de temps**

```
import java.util.Hashtable;

import org.apache.xmlbeans.XmlOptions;

import org.w3.x2000.x09.xmlldsig.DigestMethodType;

import oasis.names.tc.dss._1_0.core.wsd1.SOAPport;
import oasis.names.tc.dss._1_0.core.wsd1.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.DocumentHashDocument.DocumentHash;
import x0CoreSchema.oasisNamesTcDss1.InputDocumentsDocument.InputDocuments;
import x0CoreSchema.oasisNamesTcDss1.OptionalInputsDocument.OptionalInputs;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument.VerifyRequest;

public class ValidacioSegellDeTemps {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        // Inicialització del client
        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-test/dss");

        // Composició del missatge

        // Timestamp a verificar
        String timestamp = Utils.readXmlFile("timestamp-xml.dat");

        // Dades del digest
        byte[] digest = Utils.readBase64File("timestamp-digest1.dat");

        // Definició dels namespaces correctes
        Hashtable prefixes = new Hashtable();
        prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
        prefixes.put("http://www.w3.org/2000/09/xmlldsig#", "ds");

        XmlOptions options = new XmlOptions();
        options.setSaveSuggestedPrefixes(prefixes);

        // Creació del missatge DSS
        VerifyRequestDocument requestDocument =
        VerifyRequestDocument.Factory.newInstance(options);

        VerifyRequest request = requestDocument.addNewVerifyRequest();
        request.setProfile("urn:oasis:names:tc:dss:1.0:profiles:timestamping");

        // Creació de l'element amb els segell de temps
        SignatureObjectType signaturetype = SignatureObjectType.Factory.parse(timestamp,
        options);

        request.setSignatureObject(signaturetype);

        // Creació de l'element amb un document
        InputDocuments documents = request.addNewInputDocuments();

        DocumentHash document = documents.addNewDocumentHash();
        document.setID("Doc1");
        document.setDigestValue(digest);

        DigestMethodType method = document.addNewDigestMethod();
        method.setAlgorithm("http://www.w3.org/2000/09/xmlldsig#sha1");
```

```
// Creació de l'element amb els paràmetres opcionals a consultar
OptionalInputs optInputs = OptionalInputs.Factory.newInstance();
optInputs.addNewReturnProcessingDetails();

request.setOptionalInputs(optInputs);

// Execució del servei per verificar
VerifyResponseDocument responseDocument = port.verify(requestDocument);

// Visualització de la petició
System.out.println(requestDocument.xmlText(options));

// visualització de la resposta
System.out.println(responseDocument.xmlText(options));

}
}
```

Figura 36 Exemple en Java de validació de segell de temps

#### Validació de certificats amb autenticació SSL

```
import java.util.Hashtable;
import java.security.Security;
import org.apache.xmlbeans.XmlBase64Binary;
import org.apache.xmlbeans.XmlOptions;

import org.w3.x2000.x09.xmlldsig.X509DataDocument;
import org.w3.x2000.x09.xmlldsig.X509DataType;

import oasis.names.tc.dss._1_0.core.wsdl.SOAPport;
import oasis.names.tc.dss._1_0.core.wsdl.digitalSignatureServiceClient;

import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument;
import x0CoreSchema.oasisNamesTcDss1.VerifyResponseDocument;
import x0CoreSchema.oasisNamesTcDss1.OptionalInputsDocument;
import x0CoreSchema.oasisNamesTcDss1.SignatureObjectType.Other;
import x0CoreSchema.oasisNamesTcDss1.VerifyRequestDocument.VerifyRequest;

public class ValidacioCertificat {

    @SuppressWarnings("unchecked")
    public static void main(String args[]) throws Exception {

        // Inicialització del client

        //configuració socket TSL

        //keystore del client

        System.setProperty("javax.net.ssl.keyStore", "path_to_pl2");
        System.setProperty("javax.net.ssl.keyStoreType", "pkcs12");
        System.setProperty("javax.net.ssl.keyStorePassword", "password_pl2");

        //trustore

        System.setProperty("javax.net.ssl.trustStore", "path_to_cacerts");
        System.setProperty("javax.net.ssl.trustStoreType", "JKS");
        System.setProperty("javax.net.ssl.trustStorePassord", "changeit");

        System.setProperty("java.protocol.handler.pkgs", "com.sun.net.ssl.internal.www.protocol");
        Security.addProvider(new com.sun.net.ssl.internal.ssl.Provider());

        digitalSignatureServiceClient proxy = new digitalSignatureServiceClient();
        SOAPport port = proxy.getdssPortSoap("http://psisbeta.catcert.net/psis/catcert-test/dss");
    }
}
```

```
// Composició del missatge

// Certificat que es verificarà
byte[] certificate = Utils.readBase64File("certificate.dat");

// Definició dels namespaces correctes
Hashtable prefixes = new Hashtable();
prefixes.put("urn:oasis:names:tc:dss:1.0:core:schema", "dss");
prefixes.put("http://www.w3.org/2000/09/xmldsig#", "ds");

XmlOptions options = new XmlOptions();
options.setSaveSuggestedPrefixes(prefixes);

// Creació del missatge DSS
VerifyRequestDocument requestDocument =
VerifyRequestDocument.Factory.newInstance(options);

VerifyRequest request = requestDocument.addNewVerifyRequest();
request.setProfile("urn:oasis:names:tc:dss:1.0:profiles:XSS");

// Creació de l'element amb el certificat a verificar
SignatureObjectType signature = request.addNewSignatureObject();

X509DataDocument x509doc = X509DataDocument.Factory.newInstance();
X509DataType x509data = x509doc.addNewX509Data();

XmlBase64Binary b64certificate = x509data.addNewX509Certificate();
b64certificate.setByteArrayValue(certificate);

Other any = signature.addNewOther();
any.set(x509doc);

signature.setOther(any);

// Creació de l'element amb els paràmetres opcionals a consultar
OptionalInputs optional = request.addNewOptionalInputs();
optional.addNewReturnProcessingDetails();

// Consulta de l'atribut X509 SubjectDN
ReturnX509CertificateInfo info = optional.addNewReturnX509CertificateInfo();

AttributeType SubjectDNcommonName = info.addNewAttributeDesignator();
SubjectDNcommonName.setName("urn:oasis:names:tc:dss:1.0:profiles:XSS:certificateAttributes:SubjectDistinguishedName:commonName");

// Execució del servei per verificar
VerifyResponseDocument responseDocument = port.verify(requestDocument);

// Visualització de la petició
System.out.println(requestDocument.xmlText(options));

// visualització de la resposta
System.out.println(responseDocument.xmlText(options));

}

}
```

Figura 36 Exemple en Java de validació certificats amb autenticació SSL

## 8.2 .NET (C#)

```
using System;
using System.Web;
using net.catcert.psis;

public class ValidacioCertificat
{
    public ValidacioCertificat()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        // Dades de prova
        byte[] certificate = utils.Base64File("c:/psis/certificate.dat");

        // Inicialització del client
        digitalSignatureService proxy = new digitalSignatureService();

        // Indicar l'adreça del servidor
        proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dss";

        // Composició del missatge
        X509DataType x509DataType = new X509DataType();

        Object[] certificates = new Object[1];
        certificates[0] = certificate;

        ItemsChoiceType[] certificateType = new ItemsChoiceType[1];
        certificateType[0] = ItemsChoiceType.X509Certificate;

        x509DataType = new X509DataType();
        x509DataType.Items = certificates;
        x509DataType.ItemsElementName = certificateType;

        SignatureObjectTypeOther other = new SignatureObjectTypeOther();
        other.X509Data = x509DataType;

        SignatureObjectType signature = new SignatureObjectType();
        signature.Item = other;

        // Creació del missatge DSS
        VerifyRequest requestDocument = new VerifyRequest();
        requestDocument.SignatureObject = signature;

        // Execució del servei
        VerifyResponse responseDocument = proxy.verify (requestDocument);

        // Visualització de la petició
        Console.WriteLine(utils.MessageToString(requestDocument));

        // Visualització de la resposta
        Console.WriteLine(utils.MessageToString(responseDocument));
    }
}
```

**Figura 37 Exemple en .net de validació de certificats**

**Validació de signatura PKCS#7 / CMS**

```
using System;
using System.Web;
using net.catcert.psis;

public class ValidacioSignaturaCMS
{
    public ValidacioSignaturaCMS()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        // Objectes temporals
        Object[] objects = null;
        ItemsChoiceType6[] types6 = null;

        // Inicialització del client
        digitalSignatureService proxy = new digitalSignatureService();

        // Indicar l'adreça del servidor
        proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dss";

        // Composició del missatge

        // Dades de prova
        byte[] signature = utils.Base64File("c:/psis/cms-signature.dat");
        byte[] doc = utils.Base64File("c:/psis/cms-doc.dat");

        // Creació de l'element amb la signatura CMS a verificar
        Base64Signature b64signature = new Base64Signature();
        b64signature.Type = "urn:ietf:rfc:3852";
        b64signature.Value = signature;

        SignatureObjectType signatureType = new SignatureObjectType();
        signatureType.Item = b64signature;

        // Creació de l'element amb els documents a enviar
        Base64Data b64data = new Base64Data();
        b64data.Value = doc;

        DocumentType[] document = new DocumentType[1];
        document[0] = new DocumentType();
        document[0].Item = b64data;

        InputDocuments inpDocuments = new InputDocuments();
        inpDocuments.Items = document;

        // Creació de l'element amb els paràmetres opcionals a consultar
        OptionalInputs optInputs = new OptionalInputs();

        objects = new Object[1];
        objects[0] = new Object();

        types6 = new ItemsChoiceType6[1];
        types6[0] = ItemsChoiceType6.ReturnProcessingDetails;

        optInputs.Items = objects;
        optInputs.ItemsElementName = types6;

        // Creació del missatge DSS
        VerifyRequest requestDocument = new VerifyRequest();
        requestDocument.SignatureObject = signatureType;
        requestDocument.InputDocuments = inpDocuments;
        requestDocument.OptionalInputs = optInputs;
```

```
// Execució del servei
VerifyResponse responseDocument = proxy.verify(requestDocument);

// Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument));

// Visualització de la resposta
Console.WriteLine(Utils.MessageToString(responseDocument));

    }
}
```

**Figura 38** Exemple en .net de validació de signatura CMS

### Validació de signatura XMLDsig

```
using System;
using System.Web;
using net.catcert.psis;

public class ValidacioSignaturaXMLDsig
{
    public ValidacioSignaturaXMLDsig()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        // Objectes temporals
        Object[] objects=null;
        ItemsChoiceType7[] types7 = null;

        // Inicialització del client
        digitalSignatureService proxy = new digitalSignatureService();

        // Indicar l'adreça del servidor
        proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dss";

        // Composició del missatge

        // Dades de prova
        System.Xml.XmlElement signature = utils.XMLFile("c:/psis/xmldsig-signature.dat");

        // Creació de l'element amb la signatura XMLDsig
        SignatureObjectType signaturetype = new SignatureObjectType();
        signaturetype.Any = signature;

        // Creació de l'element amb els paràmetres opcionals a consultar
        OptionalInputs optInputs = new OptionalInputs();

        objects = new Object[1];
        objects[0] = new Object();

        types7 = new ItemsChoiceType7[1];
        types7[0] = ItemsChoiceType7.ReturnProcessingDetails;

        optInputs.Items = objects;
        optInputs.ItemsElementName = types7;

        // Creació del missatge DSS
        VerifyRequest requestDocument = new VerifyRequest();
        requestDocument.SignatureObject = signaturetype;
    }
}
```

```
requestDocument.OptionalInputs = optInputs;

// Execució del servei per a verificar
VerifyResponse responseDocument = proxy.verify (requestDocument);

// Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument));

// Visualització de la resposta
Console.WriteLine(Utils.MessageToString(responseDocument));

}
}
```

Figura 39 Exemple en .net de validació de signatura XML

#### Validació de signatura XAdES

```
using System;
using System.Web;
using net.catcert.psis;

public class ValidacioSignaturaXAdES
{
    public ValidacioSignaturaXAdES()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        // Objectes temporals
        Object[] objects = null;
        ItemsChoiceType6[] types6 = null;

        // Inicialització del client
        digitalSignatureService proxy = new digitalSignatureService();

        // Indicar l'adreça del servidor
        proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dss";

        // Codi de composició del missatge

        // Dades de prova
        System.Xml.XmlElement signature = utils.XMLFile("c:/psis/xades-signature.dat");

        // Creació de l'element amb la signatura XMLDSig
        SignatureObjectType signaturetype = new SignatureObjectType();
        signaturetype.Any = signature;

        // Creació de l'element amb els paràmetres opcionals a consultar
        OptionalInputs optInputs = new OptionalInputs();

        objects = new Object[1];
        objects[0] = new Object();

        types6 = new ItemsChoiceType6[1];
        types6[0] = ItemsChoiceType6.ReturnProcessingDetails;

        optInputs.Items = objects;
        optInputs.ItemsElementName = types6;

        // Creació del missatge DSS
        VerifyRequest requestDocument = new VerifyRequest();
```

```
requestDocument.SignatureObject = signaturetype;
requestDocument.OptionalInputs = optInputs;

// Execució del servei per a verificar
VerifyResponse responseDocument = proxy.verify(requestDocument);

// Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument));

// Visualització de la resposta
Console.WriteLine(Utils.MessageToString(responseDocument));

}
```

Figura 40 Exemple en .net de validació de signatura XAdES

#### Validació de document PDF signat

```
using System;
using System.Web;

using net.catcert.psis;

public class ValidacioPDF
{
    public ValidacioPDF()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        // Objectes temporals
        Object[] objects=null;
        ItemsChoiceType7[] types7 = null;

        // Inicialització del client
        digitalSignatureService proxy = new digitalSignatureService();

        // Indicar l'adreça del servidor
        //
        // ATENCIO: http://.../dsspdf
        //
        proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dsspdf";

        // Codi de composició del missatge

        // Document PDF a validar
        byte[] doc = utils.Base64File("c:/psis/doc.pdf");

        // Creació de l'element amb els documents a enviar
        Base64Data b64data = new Base64Data();
        b64data.Value = doc;

        DocumentType[] document = new DocumentType[1];
        document[0] = new DocumentType();
        document[0].Item = b64data;

        InputDocuments inpDocuments = new InputDocuments();
        inpDocuments.Items = document;

        // Creació de l'element amb els paràmetres opcionals a consultar
        OptionalInputs optInputs = new OptionalInputs();
```

```

objects = new object[2];
objects[0] = new Object();
objects[1] = new Object();

types7 = new ItemsChoiceType7[2];
types7[0] = ItemsChoiceType7.ReturnProcessingDetails;
types7[1] = ItemsChoiceType7.ReturnSignatureReason;

optInputs.Items = objects;
optInputs.ItemsElementName = types7;

// Creació del missatge DSS
VerifyRequest requestDocument = new VerifyRequest();
requestDocument.InputDocuments = inpDocuments;
requestDocument.OptionalInputs = optInputs;

//
// !!! IMPORTANT !!!
//
// Activar perfil PDF
//
requestDocument.Profile = "urn:oasis:names:tc:dss:1.0:profiles:DSS_PDF";

// Execució del servei per a verificar
VerifyResponse responseDocument = proxy.verify (requestDocument);

// Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument));

// Visualització de la resposta
Console.WriteLine(Utils.MessageToString(responseDocument));

}
}

```

Figura 41 Exemple en .net de validació de document PDF signat

#### Creació de segells de temps

```

using System;
using System.Web;

using net.catcert.psis;

public class CreacioSegellDeTemps
{
    public CreacioSegellDeTemps()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        // Objectes temporals
        Object[] objects = null;
        ItemsChoiceType[] types = null;
        ItemsChoiceType1[] types1 = null;
        ItemsChoiceType6[] types6 = null;

        // Inicialització del client
        digitalSignatureService proxy = new digitalSignatureService();

        // Indicar l'adreça del servidor
        proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dss";
    }
}

```

```
// Codi de composició del missatge

// Dades de prova
byte[] certificate = utils.Base64File("c:/psis/timestamp-certificate.dat");
byte[] digest = utils.Base64File("c:/psis/timestamp-digest1.dat");

// Tipus de signatura disponibles
// TimeStamp amb signatura CMS/CADES:          urn:ietf:rfc:3161
// TimeStamp amb signatura XMLDsig:
oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken
// TimeStamp amb signatura XAdES:
oasis:names:tc:dss:1.0:core:schema:XAdESTimeStampToken
string signatureType = "oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken";

// Creació de l'element amb un document
DigestMethodType method = new DigestMethodType();
method.Algorithm = "http://www.w3.org/2000/09/xmldsig#sha1";

DocumentHash document = new DocumentHash();
document.ID = "Doc1";
document.DigestMethod = method;
document.DigestValue = digest;

// Creació de l'element amb els documents a signar
InputDocuments documents = new InputDocuments();
documents.Items = new DocumentHash[1] { document };

// Creació de l'element amb un certificat
objects = new Object[1];
objects[0] = certificate;

types = new ItemsChoiceType[1];
types[0] = ItemsChoiceType.X509Certificate;

X509DataType x509DataType = new X509DataType();
x509DataType.Items = objects;
x509DataType.ItemsElementName = types;

// Creació de l'element opcional amb el certificat per fer el segell de temps
// Continuarà el certificat de TSA de CATCert en Base64
objects = new Object[1];
objects[0] = x509DataType;

types1 = new ItemsChoiceType1[1];
types1[0] = ItemsChoiceType1.X509Data;

KeyInfoType keyInfo = new KeyInfoType();
keyInfo.Items = objects;
keyInfo.ItemsElementName = types1;

KeySelector keySelector = new KeySelector();
keySelector.Item = keyInfo;

// Creació de l'element opcional amb l'objecte
IncludeObject iObject = new IncludeObject();
iObject.ObjId = "Doc1";
iObject.WhichDocument = "Doc1";
iObject.HasObjectTagsAndAttributesSet = false;
iObject.CreateReference = true;

// Creació de l'element amb el conjunt de paràmetres opcionals a consultar
objects = new Object[3];
objects[0] = keySelector;
objects[1] = signatureType;
objects[2] = iObject;

types6 = new ItemsChoiceType6[3];
types6[0] = ItemsChoiceType6.KeySelector;
types6[1] = ItemsChoiceType6.SignatureType;
types6[2] = ItemsChoiceType6.IncludeObject;

OptionalInputs optInputs = new OptionalInputs();
optInputs.Items = objects;
optInputs.ItemsElementName = types6;
```

```
// Creació del missatge DSS
SignRequest requestDocument = new SignRequest();
requestDocument.OptionalInputs = optInputs;
requestDocument.InputDocuments = documents;

// Execució del servei per a verificar
SignResponse responseDocument = proxy.sign(requestDocument);

// Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument));

// Visualització de la resposta
Console.WriteLine(Utils.MessageToString(responseDocument));

    }
}
```

Figura 42 Exemple en .net de creació de segell de temps

#### Validació de segells de temps

```
using System;
using System.Web;

using net.catcert.psis;

public class ValidacioSegellDeTemps
{
    public ValidacioSegellDeTemps()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        // Objectes temporals
        Object[] objects=null;
        ItemsChoiceType6[] types6 = null;

        // Inicialització del client
        digitalSignatureService proxy = new digitalSignatureService();

        // Indicar l'adreça del servidor
        proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dss";

        // Composició del missatge

        // Dades de prova
        System.Xml.XmlElement timestamp = utils.XMLFile("c:/psis/timestamp-xml.dat");

        byte[] digest = utils.Base64File("c:/psis/timestamp-digest1.dat");

        // Creació de l'element amb un document
        DigestMethodType method = new DigestMethodType();
        method.Algorithm="http://www.w3.org/2000/09/xmldsig#sha1";

        DocumentHash document = new DocumentHash();
        document.ID = "Doc1";
        document.DigestMethod = method;
        document.DigestValue = digest;

        // Creació de l'element amb els documents a verificar
        InputDocuments documents = new InputDocuments();
```

```
documents.Items = new DocumentHash[1] {document};

// Creació de l'element amb el segell de temps
SignatureObjectType signaturetype = new SignatureObjectType();
signaturetype.Any=timestamp;

// Creació de l'element amb els paràmetres opcionals a consultar
OptionalInputs optInputs = new OptionalInputs();

objects = new Object[1];
objects[0] = new Object();

types6 = new ItemsChoiceType6[1];
types6[0] = ItemsChoiceType6.ReturnProcessingDetails;

optInputs.Items = objects;
optInputs.ItemsElementName = types6;

// Creació del missatge DSS
VerifyRequest requestDocument = new VerifyRequest();
requestDocument.SignatureObject = signaturetype;
requestDocument.OptionalInputs = optInputs;
requestDocument.InputDocuments = documents;

// Execució del servei per a verificar
VerifyResponse responseDocument = proxy.verify (requestDocument);

// Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument));

// Visualització de la resposta
Console.WriteLine(Utils.MessageToString(responseDocument));
}
}
```

Figura 43 Exemple en .net de validació de segell de temps

#### Validació de certificats amb autenticació SSL

```
using System;
using System.Web;
using System.Security;
using System.Security.Cryptography.X509Certificates;

using net.catcert.psis;

public class ValidacioCertificat
{
    public ValidacioCertificat()
    {
    }

    static void Main()
    {
        // Utilitats
        Utils utils = new Utils();

        //Keystore client
        X509Certificate2 p12client = null;
        p12client = new X509Certificate2("path_to_p12", "password_p12");

        // Dades de prova
        byte[] certificate = utils.Base64File("c:/psis/certificate.dat");
```

```
// Inicialització del client
digitalSignatureService proxy = new digitalSignatureService();

// Indicar l'adreça del servidor
proxy.Url = "http://psisbeta.catcert.net/psis/catcert-test/dss";
proxy.ClientCertificates.Add(p12client);

// Composició del missatge
X509DataType x509DataType = new X509DataType();

Object[] certificates = new Object[1];
certificates[0] = certificate;

ItemsChoiceType[] certificateType = new ItemsChoiceType[1];
certificateType[0] = ItemsChoiceType.X509Certificate;

x509DataType = new X509DataType();
x509DataType.Items = certificates;
x509DataType.ItemsElementName = certificateType;

SignatureObjectTypeOther other = new SignatureObjectTypeOther();
other.X509Data = x509DataType;

SignatureObjectType signature = new SignatureObjectType();
signature.Item = other;

// Creació del missatge DSS
VerifyRequest requestDocument = new VerifyRequest();
requestDocument.SignatureObject = signature;

// Execució del servei
VerifyResponse responseDocument = proxy.verify (requestDocument);

// Visualització de la petició
Console.WriteLine(Utils.MessageToString(requestDocument));

// Visualització de la resposta
Console.WriteLine(Utils.MessageToString(responseDocument));

}
}
```

## 8.3 Visual Basic 6

### Validació de certificats

```
Public Sub Main()

    'Utilitats
    Set Utils = New PSISClient.Utils

    'Inicialització del client
    Dim proxy As New PSISClient.digitalSignatureService
    proxy.url = "http://psisbeta.catcert.net/psis/catcert-test/dss"

    'Composició del missatge
    Dim certsObj(1)
    certsObj(0) = Utils.Base64File("c:/psis/certificate.dat")

    Dim types(1) As Integer
    types(0) = ItemsChoiceType.ItemsChoiceType_X509Certificate

    Dim certificate As New X509DataType
```

```
certificate.Items = certsObj
certificate.ItemsElementName = types

Dim other As New SignatureObjectTypeOther
other.X509Data = certificate

Dim signatureType As New SignatureObjectType
signatureType.Item = other

'Creació del missatge DSS
Dim requestDocument As New PSISClient.VerifyRequest
requestDocument.SignatureObject = signatureType

'Execució del servei
Dim responseDocument As PSISClient.VerifyResponse
Set responseDocument = proxy.verify(requestDocument)

'Visualització de la petició
Debug.Print Utils.MessageToString(requestDocument)

'Visualització de la resposta
Debug.Print Utils.MessageToString(responseDocument)

End Sub
```

**Figura 44 Exemple en Visual Basic de validació de certificats**

**Validació de signatura PKCS#7 / CMS**

```
Public Sub Main()

    'Utilitats
    Set Utils = New PSISClient.Utils

    'Inicialització del client
    Dim proxy As New PSISClient.digitalSignatureService
    proxy.url = "http://psisbeta.catcert.net/psis/catcert-test/dss"

    'Codi de composició del missatge

    'Creació de l'element amb la signatura CMS a verificar
    Dim signature As New PSISClient.Base64Signature
    signature.Type = "urn:ietf:rfc:3852"
    signature.Value = Utils.Base64File("c:/psis/cms-signature.dat")

    Dim sign As New PSISClient.SignatureObjectType
    Set sign.Item = signature

    'Documents a enviar
    Dim data As New PSISClient.Base64Data
    data.Value = Utils.Base64File("c:/psis/cms-doc.dat")

    Dim docType(1) As PSISClient.DocumentType
    Set docType(0) = New PSISClient.DocumentType
    Set docType(0).Item = data

    Dim clearText As New PSISClient.InputDocuments
    clearText.Items = docType

    'Creació de l'element amb els paràmetres opcionals a consultar
    Dim optInputs As PSISClient.OptionalInputs
    Set optInputs = New PSISClient.OptionalInputs

    Dim objects(1) As Object
    Dim type7(1) As Long

    Set objects(0) = New ObjectType
    type7(0) = PSISClient.ItemsChoiceType7_ReturnProcessingDetails

    optInputs.Items = objects
    optInputs.ItemsElementName = type7
```

```
'Creació del missatge DSS
Dim requestDocument As New PSISClient.VerifyRequest
Set requestDocument.SignatureObject = sign
Set requestDocument.InputDocuments = clearText
Set requestDocument.OptionalInputs = optInputs

'Execució del servei per verificar
Dim responseDocument As PSISClient.VerifyResponse
Set responseDocument = proxy.verify(requestDocument)

'Visualització de la petició
Debug.Print Utils.MessageToString(requestDocument)

'Visualització de la resposta
Debug.Print Utils.MessageToString(responseDocument)

End Sub
```

**Figura 45 Exemple en Visual Basic de signatura CMS**

**Validació de signatura XMLDsig**

```
Private Sub main()  
  
    'Utilitats  
    Set Utils = New PSISClient.Utils  
  
    'Inicialització del client  
    Dim proxy As New PSISClient.digitalSignatureService  
    proxy.url = "http://psisbeta.catcert.net/psis/catcert-test/dss"  
  
    'Composició del missatge  
  
    'Creació de l'element amb la signatura XMLDsig  
    Dim signatureType As New SignatureObjectType  
    signatureType.Any = Utils.XMLFile("c:/psis/xmldsig-signature.dat")  
  
    'Creació de l'element amb els paràmetres opcionals a consultar  
    Dim optInputs As New OptionalInputs  
  
    Dim types7(1) As Long  
    types7(0) = ItemsChoiceType7.ItemsChoiceType7_ReturnProcessingDetails  
  
    Dim objInputs(1)  
    Set objInputs(0) = New ObjectType  
  
    optInputs.Items = objInputs  
    optInputs.ItemsElementName = types7  
  
    'Creació del missatge DSS  
    Dim requestDocument As New PSISClient.VerifyRequest  
    requestDocument.SignatureObject = signatureType  
    requestDocument.OptionalInputs = optInputs  
  
    'Execució del servei per verificar  
    Set responseDocument = proxy.verify(requestDocument)  
  
    'Visualització de la petició  
    Debug.Print Utils.MessageToString(requestDocument)  
  
    'Visualització de la resposta  
    Debug.Print Utils.MessageToString(responseDocument)  
  
End Sub
```

**Figura 46 Exemple en Visual Basic de signatura XML**

**Validació de signatura XAdES**

```
Private Sub main()  
  
    'Utilitats  
    Set Utils = New PSISClient.Utils  
  
    'Inicialització del client  
    Dim proxy As New PSISClient.digitalSignatureService  
    proxy.url = "http://psisbeta.catcert.net/psis/catcert-test/dss"  
  
    'Composició del missatge  
  
    'Creació de l'element amb la signatura XMLDsig  
    Dim signatureType As New SignatureObjectType  
    signatureType.Any = Utils.XMLFile("c:/psis/xades-signature.dat")  
  
    'Creació de l'element amb els paràmetres opcionals a consultar  
    Dim optInputs As New OptionalInputs
```

```
Dim types7(1) As Long
types7(0) = ItemsChoiceType7.ItemsChoiceType7_ReturnProcessingDetails

Dim objInputs(1)
Set objInputs(0) = New ObjectType

optInputs.Items = objInputs
optInputs.ItemsElementName = types7

'Creació del missatge DSS
Dim requestDocument As New PSISClient.VerifyRequest
requestDocument.SignatureObject = signatureType
requestDocument.OptionalInputs = optInputs

'Execució del servei per verificar
Set responseDocument = proxy.verify(requestDocument)

'Visualització de la petició
Debug.Print Utils.MessageToString(requestDocument)

'Visualització de la resposta
Debug.Print Utils.MessageToString(responseDocument)

End Sub
```

**Figura 47 Exemple en Visual Basic de signatura XAdES**

#### Validació de document PDF signat

```
Private Sub Main()

'Utilitats
Set Utils = New PSISClient.Utils

'Inicialització del client. Atenció a la URL (http://.../dsspdf
Dim proxy As New PSISClient.digitalSignatureService
proxy.url = "http://psisbeta.catcert.net/psis/catcert-test/dsspdf"

'Codi de composició del missatge

'Documents a enviar
Dim data As New PSISClient.Base64Data
data.Value = Utils.Base64File("c:/psis/doc.pdf")

Dim docType(1) As PSISClient.DocumentType
Set docType(0) = New PSISClient.DocumentType
Set docType(0).Item = data

Dim clearText As New PSISClient.InputDocuments
clearText.Items = docType

'Creació de l'element amb els paràmetres opcionals a consultar
Dim optInputs As PSISClient.OptionalInputs
Set optInputs = New PSISClient.OptionalInputs

Dim objects(2) As Object
Dim type7(2) As Long

Set objects(0) = New ObjectType
type7(0) = PSISClient.ItemsChoiceType7_ReturnProcessingDetails

Set objects(1) = New ObjectType
type7(1) = PSISClient.ItemsChoiceType7_ReturnSignatureReason

optInputs.Items = objects
optInputs.ItemsElementName = type7

'Creació del missatge DSS
```

```
Dim requestDocument As New PSISClient.VerifyRequest
Set requestDocument.InputDocuments = clearText
Set requestDocument.OptionalInputs = optInputs

'!!! IMPORTANT !!!
'
'Activar perfil PDF
request.Profile = "urn:oasis:names:tc:dss:1.0:profiles:DSS_PDF"

'Execució del servei per verificar
Set responseDocument = proxy.verify(requestDocument)

'Visualització de la petició
Debug.Print Utils.MessageToString(requestDocument)

'Visualització de la resposta
Debug.Print Utils.MessageToString(responseDocument)

End Sub
```

Figura 48 Exemple en Visual Basic de document PDF signat

#### Creació de segells de temps

```
Private Sub main()

'Utilitats
Set Utils = New PSISClient.Utils

'Tipus de signatura disponibles
'TimeStamp amb signatura CMS/CADES: urn:ietf:rfc:3161
'TimeStamp amb signatura XMLDsig: oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken
'TimeStamp amb signatura XAdES:
oasis:names:tc:dss:1.0:core:schema:XAdESTimeStampToken
Dim signatureType As String
signatureType = "oasis:names:tc:dss:1.0:core:schema:XMLTimeStampToken"

'Inicialització del client
Dim proxy As New PSISClient.digitalSignatureService
proxy.url = "http://psisbeta.catcert.net/psis/catcert-test/dss"

'Codi de composició del missatge

'Creació de l'element amb un document
Dim method As New DigestMethodType
method.Algorithm = "http://www.w3.org/2000/09/xmlldsig#sha1"

Dim document(1) As DocumentHash
Set document(0) = New DocumentHash
document(0).ID = "Doc1"
document(0).digestMethod = method
document(0).DigestValue = Utils.Base64File("c:/psis/timestamp-digest1.dat")

'Creació de l'element amb els documents a signar
Dim documents As New InputDocuments
documents.Items = document

'Creació de l'element amb un certificat
Dim certificates(1)
certificates(0) = Utils.Base64File("c:/psis/timestamp-certificate.dat")

Dim types(1) As Long
types(0) = ItemsChoiceType.ItemsChoiceType_X509Certificate

Dim certData As New X509DataType
certData.Items = certificates
certData.ItemsElementName = types

'Creació de l'element opcional amb el certificat per fer el segell de temps
Dim dataObject(1) As Object
```

```

Set dataObject(0) = certData

Dim types1(1) As Long
types1(0) = ItemsChoiceType1.ItemsChoiceType1_X509Data

Dim keyInfo As New KeyInfoType
keyInfo.Items = dataObject
keyInfo.ItemsElementName = types1

Dim keySlt As New KeySelector
Set keySlt.Item = keyInfo

'Creació de l'element amb el conjunt de paràmetres opcionals a consultar
Dim optObjects(2)
Set optObjects(0) = keySlt
optObjects(1) = signatureType
optObjects(2) = iObject

Dim types7(2) As Long
types7(0) = ItemsChoiceType7.ItemsChoiceType7_KeySelector
types7(1) = ItemsChoiceType7.ItemsChoiceType7_SignatureType

Dim optInputs As New OptionalInputs
optInputs.Items = optObjects
optInputs.ItemsElementName = types7

'Creació del missatge DSS
Dim requestDocument As New PSISClient.SignRequest
Set requestDocument.InputDocuments = documents
Set requestDocument.OptionalInputs = optInputs

'Execució del servei per signar
Set responseDocument = proxy.sign(requestDocument)

'Visualització de la petició
Debug.Print Utils.MessageToString(requestDocument)

'Visualització de la resposta
Debug.Print Utils.MessageToString(responseDocument)

End Sub

```

**Figura 49 Exemple en Visual Basic de creació de segell de temps**

#### Validació de segells de temps

```

Private Sub main()

'Utilitats
Set Utils = New PSISClient.Utils

'Inicialització del client
Dim proxy As New PSISClient.digitalSignatureService
proxy.url = "http://psisbeta.catcert.net/psis/catcert-test/dss"

'Composició del missatge

'Creació de l'element amb un document
Dim method As New DigestMethodType
method.Algorithm = "http://www.w3.org/2000/09/xmldsig#sha1"

Dim document(1) As DocumentHash
Set document(0) = New DocumentHash
document(0).ID = "Doc1"
document(0).digestMethod = method
document(0).DigestValue = Utils.Base64File("c:/psis/timestamp-digest1.dat")

'Creació de l'element amb els documents a verificar
Dim documents As New InputDocuments

```

```
documents.Items = document

' Creació de l'element amb el segell de temps
Dim signatureType As New SignatureObjectType
signatureType.Any = Utils.XMLFile("c:/psis/timestamp-xml.dat")

'Creació de l'element amb els paràmetres opcionals a consultar
Dim optInputs As New OptionalInputs

Dim types7(1) As Long
types7(0) = ItemsChoiceType7.ItemsChoiceType7_ReturnProcessingDetails

Dim objInputs(1)
Set objInputs(0) = New ObjectType

optInputs.Items = objInputs
optInputs.ItemsElementName = types7

'Creació del missatge DSS
Dim requestDocument As New PSISClient.VerifyRequest
requestDocument.SignatureObject = signatureType
requestDocument.OptionalInputs = optInputs
requestDocument.InputDocuments = documents

'Execució del servei per verificar
Set responseDocument = proxy.verify(requestDocument)

'Visualització de la petició
Debug.Print Utils.MessageToString(requestDocument)

'Visualització de la resposta
Debug.Print Utils.MessageToString(responseDocument)

End Sub
```

**Figura 50 Exemple en Visual Basic de creació de segell de temps**

## 9. Annexes

### 9.1 Referències

Fitxer	Títol
<a href="#">dss-v1[1].0-spec-cd-Core-r03.pdf</a>	<i>Digital Signature Service Core Protocols, Elements, and Bindings</i>
<a href="#">OASIS-dss-1.0-core-profiles-XSS-spec-wd02.doc</a>	<i>eXtended Signature Services (XSS) Profile of the OASIS Digital Signature Service (DSS)</i>

URL	Descripció
<a href="http://www.OASIS-open.org/committees/tc_home.php?wg_abbrev=dss">http://www.OASIS-open.org/committees/tc_home.php?wg_abbrev=dss</a>	Pàgina web oficial d'estandardització del protocol DSS
<a href="http://www.webservices.org/">http://www.webservices.org/</a>	Pàgina web d'informació general referent als serveis web
<a href="http://www.w3.org/TR/soap/">http://www.w3.org/TR/soap/</a>	Pàgina web d'estandardització del protocol SOAP
<a href="http://www.w3.org/TR/XAdES/">http://www.w3.org/TR/XAdES/</a>	Pàgina web d'estandardització de signatures digitals XAdES
<a href="http://www.w3.org/TR/XMLDsig-core/">http://www.w3.org/TR/XMLDsig-core/</a>	Pàgina web d'estandardització de signatures digitals DSIG
<a href="http://ws.apache.org/axis/">http://ws.apache.org/axis/</a>	Pàgina web oficial d'AXIS
<a href="http://XMLbeans.apache.org/">http://XMLbeans.apache.org/</a>	Pàgina web oficial de XMLBeans
<a href="http://xfire.codehaus.org/">http://xfire.codehaus.org/</a>	Pàgina web oficial de XFire

### 9.2 Codis de resposta genèrics

Conjunt de codis de retorn més comuns que els clients poden trobar com a resultat de les operacions realitzades a la plataforma PSIS. Tot i això, per a tenir la totalitat dels mateixos, així com una descripció totalment estàndard, pot consultar els documents adjuntats amb la guia mateixa.

urn: oasis:names:tc:dss:1.0:resultmajor: + [valor]		
Valor	Tipus	Descripció
<i>Success</i>	OK	Protocol complert correctament
<i>RequesterError</i>	Error	El missatge que conté la petició del client es incorrecte, ja sigui sintàcticament o semànticament
<i>ResponderError</i>	Error	La petició no s'ha pogut completar per un error en el servidor

urn: oasis:names:tc:dss:1.0:resultminor: + [valor]		
Valor	Tipus	Descripció
<i>SignaturePropertiesNotSupported</i>	Error	El tipus de signatura no suporta les propietats demanades.

<i>InvalidSignatureAttribute</i>	Error	L'atribut de la signatura no és conegut pel servidor.
<i>UnsupportedSignatureAttribute</i>	Error	L'atribut de la signatura no és suportat pel servidor.
<i>InvalidCertificateAttribute</i>	Error	L'atribut del certificat no és conegut pel servidor.
<i>UnsupportedCertificateAttribute</i>	Error	L'atribut del certificat no és suportat pel servidor.
<i>NotAuthorized</i>	Error	El client no està autoritzat a fer la petició.
<i>NotSupported</i>	Error	No es suporta o reconeix la petició sol·licitada pel client.
<i>NotParseableXMLDocument</i>	Error	No es pot <i>parsejar</i> el document com a un XML vàlid.
<i>XMLDocumentNotValid</i>	Error	No es pot validar el document contra l'esquema que el valida.
<i>XPathEvaluationError</i>	Error	El resultat d'avaluar l'expressió XPath indicada és erroni.
<i>MoreThanOneRefUriOmitted</i>	Error	En el cas de signatures detached s'ha adjuntat més d'un document amb URI nul·la. Prohibit per DSS.

### 9.3 Codis de resposta de generació

urn:oasis:names:tc:dss:1.0:resultminor: + [valor]		
Valor	Tipus	Descripció
<i>KeyNotFound_InvalidIdentifier</i>	Error	No es pot trobar la clau indicada a la petició.
<i>KeyNotFound_MoreThanOneKeyFound</i>	Error	Hi ha més d'una clau disponible amb la clau indicada a la petició.
<i>IncompatibleSignatureForms</i>	Error	El "Signature Form" és incompatible amb la "Signature Policy" indicada a la petició.
<i>SignatureFormsNotSupported</i>	Error	S'ha sol·licitat la generació d'un tipus de signatura no suportada pel servidor.

### 9.4 Codis de resposta de validació

urn:oasis:names:tc:dss:1.0:resultminor: + [valor]		
Valor	Tipus	Descripció
<i>SchemaNotFound</i>	Error	No s'ha trobat l'esquema indicat al servidor.
<i>valid:signature:onAllDocuments</i>	OK	Signatura o segell de temps vàlids.
<i>valid:signature:onTransformedDocuments</i>	OK	Signatura o segell de temps vàlids sobre documents enviats amb transformacions no especificades pel client.
<i>valid:signature:notAllDocumentsReferenced</i>	OK	Signatura o segell de temps vàlids sobre algun dels documents enviats pel client però no tots els documents adjunts dintre dels InputDocuments estan referenciats per la signatura.
<i>Invalid:referencedDocumentNotPresent</i>	Error	La petició de validació no conté algun dels documents referenciats per la signatura.
<i>Invalid:indeterminateKey</i>	Error	El servidor no pot determinar la validesa del certificat de la signatura. Això pot ser degut a que no pot construir el path fins a una arrel de confiança, o bé que no pot validar aquest path. La no validació ve causada normalment per l'absència d'informació de revocació vàlida.
<i>Invalid:untrustedKey</i>	Error	El servidor no considera que el certificat de la signatura sigui vàlid. Això vol dir que està revocat o <i>suspès</i> .
<i>Invalid:incorrectSignature</i>	Error	La signatura no és correcta.
<i>inappropriate:signature</i>	Error	La signatura no és correcta en el context actual.
<i>indetermined:checkOptionalOutput</i>	Error	El client ha de verificar la resposta obtinguda en els elements

		OptionalOutput per a determinar l'error.
--	--	------------------------------------------

urn:oasis:names:tc:dss:1.0:profiles:XSS:resultminor: + [valor]		
Valor	Tipus	Descripció
<i>InvalidCertificateAttribute</i>	Error	S'ha demanat un atribut inexistent per a ésser extret del certificat
<i>InvalidSignatureAttribute</i>	Error	S'ha demanat un atribut inexistent per a ésser extret de la signatura

urn:oasis:names:tc:dss:1.0:profiles:XSS:resultminor:valid:certificate: + [valor]		
Valor	Tipus	Descripció
<i>Definitive</i>	Ok	El certificat d'entitat final enviat a validar es vàlid
<i>Temporal</i>	Ok	El certificat d'entitat final enviat a validar es vàlid però el servidor no té certesa absoluta de que aquesta validesa sigui definitiva

urn:oasis:names:tc:dss:1.0:profiles:XSS:resultminor:invalid:certificate: + [valor]		
Valor	Tipus	Descripció
<i>OnHold</i>	Error	El certificat d'entitat final enviat a validar està en esta suspès
<i>Revoked</i>	Error	El certificat d'entitat final enviat a validar està revocat
<i>Expired</i>	Error	El certificat d'entitat final enviat a validar ha expirat
<i>NotYetValid</i>	Error	El certificat d'entitat final enviat a validar encara no ha començat el seu període de validesa
<i>QualifiedCertificateRequired</i>	Error	El servidor requeria que el certificat a validar fos qualificat i l'enviat no ho és
<i>CertificatePolicyNotSupported</i>	Error	El certificat enviat està estampat seguint una política de certificació no suportada pel servidor

urn:oasis:names:tc:dss:1.0:profiles:XSS:resultminor:unknown:certificate: + [valor]		
Valor	Tipus	Descripció
<i>Status_NoCertificatePathFound</i>	Error	El servidor no ha pogut construir una cadena de certificats vàlida a partir del certificat d'entitat final a validar. Això pot ser perquè no disposa d'accés als certificats de les autoritats de certificació intermitges, la informació de revocació sobre aquests certificats o be l'arrel de confiança en el que acaba aquesta cadena.
<i>RevocationStatusInfoNotFound</i>	Error	El servidor no pot trobar la informació de revocació d'algun dels certificats de la cadena
<i>UntrustedRevocationStatusInfo</i>	Error	El servidor pot trobar la informació de revocació però no confia en ella donat que no és vàlida criptogràficament o be el seu període de validesa ha expirat
<i>BadCertificateFormat</i>	Error	El certificat a validar no està codificat correctament
<i>BadCertificateSignature</i>	Error	La signatura que protegeix el certificat no és vàlida

## 9.5 Atributs de consulta d'un certificat

El perfil XSS que permet l'extracció de camps dels certificats validats a PSIS (ja sigui per validació directa o per estar inclosos dins d'una signatura a validar) tanmateix possibilita un processat semàntic del contingut dels mateixos, ja que no es tracta únicament d'una extracció simple. Això propicia que la informació recuperada sigui la mateixa, independentment de la política presa per l'emissor del certificat. Per exemple, ens permetria extreure el DNI del titular del certificat independentment d'on i de com ho hagi inclòs l'emissor del certificat.

urn:oasis:names:tc:dss:1.0:profiles:XSS:certificateAttributes:+[Paràmetre]		
Paràmetre	Descripció	Tipus
<i>SubjectDistinguishedName:organizationName</i>	Nom de l'organització de la qual forma part el subjecte del certificat	Global
<i>SubjectDistinguishedName:commonName</i>	Nom de pila del subjecte del certificat	Global
<i>SubjectDistinguishedName:surname</i>	Cognom del subjecte del certificat	Global
<i>SubjectDistinguishedName:givenName</i>	Nom del subjecte del certificat	Global
<i>SubjectDistinguishedName:serialNumber</i>	Nombre de sèrie del certificat del subjecte del certificat	Global
<i>SubjectDistinguishedName:title</i>	Títol del subjecte del certificat	Global
<i>SubjectDistinguishedName:countryName</i>	Nom del país del subjecte del certificat	Global
<i>SubjectDistinguishedName:stateOrProvinceName</i>	Nom de la província del subjecte del certificat	Global
<i>IssuerDistinguishedName:commonName</i>	Nom de l'emissor del certificat	Global
<i>Extension:authorityKeyIdentifier</i>	Identificador de la clau de l'emissor	Extensió
<i>Extension:subjectKeyIdentifier</i>	Identificador de la clau del subjecte del certificat	Extensió
<i>Extension:privateKeyUsagePeriod</i>	Sol·licitud de canvi del temps de validesa d'una clau pública	Extensió
<i>Extension:certificatePolicies</i>	Sol·licitud de les polítiques del certificat	Extensió
<i>Extension:policyMappings</i>	Sol·licitud dels parells de claus del certificat	Extensió
<i>Extension:subjectAltName</i>	Nom alternatiu del subjecte	Extensió
<i>Extension:issuerAltName</i>	Nom alternatiu de l'emissor	Extensió
<i>Extension:subjectDirectoryAttributes</i>	No disponible	Extensió
<i>Extension:basicConstraints</i>	No disponible	Extensió
<i>Extension:nameConstraints</i>	No disponible	Extensió
<i>Extension:policyConstraints</i>	No disponible	Extensió
<i>Extension:keyUsage</i>	No disponible	Extensió
<i>Extension:extKeyUsage</i>	No disponible	Extensió
<i>Extension:cRLDistributionPoints</i>	Punts de distribució de la CRL per al certificat	Extensió
<i>Extension:inhibitAnyPolicy</i>	No disponible	Extensió
<i>Extension:freshestCRL</i>	No disponible	Extensió
<i>Extension:authorityInfoAccess</i>	Sol·licitud d'informació del mètode d'accés a la informació i serveis del CA per l'emissor del certificat	Extensió
<i>Extension:subjectInfoAccess</i>	No disponible	Extensió
<i>Version</i>	Sol·licitud de la versió del certificat	Global

<i>SerialNumber</i>	Sol·licitud del nombre de sèrie del certificat	Global
<i>Signature</i>	Sol·licitud de la signatura del certificat	Global
<i>SignatureAlgorithm</i>	Sol·licitud de l'algoritme usat per a signar el certificat	Global
<i>IssuerDistinguishedName</i>	Nom de l'emissor del certificat	Global
<i>SubjectDistinguishedName</i>	Nom del subjecte del certificat	Global
<i>NotBefore</i>	Sol·licitud de la data d'inici de validesa del certificat	Global
<i>NotAfter</i>	Sol·licitud de la data de finalització de validesa del certificat	Global
<i>SubjectPublicKeyAlgorithm</i>	Sol·licitud de l'algoritme de generació de la clau pública	Global
<i>SubjectPublicKey</i>	Sol·licitud de la clau pública del certificat	Global
<i>KeyUsages</i>	Sol·licitud dels usos permesos de la clau del certificat	Global
<i>SubjectEmail</i>	Direcció de correu electrònic del subjecte	Global
<i>CertificatePolicies</i>	Polítiques de certificació	Global

urn:catcert:psis:certificateAttributes:notaries:+[Paràmetre]		
Paràmetre	Descripció	Tipus
<i>AuthorizingNotary</i>	No disponible	Particular CATCERT
<i>RepresentationDocumentLocationData</i>	No disponible	Particular CATCERT
<i>EntitlementsRegistryLocationData</i>	No disponible	Particular CATCERT

urn:catcert:psis:certificateAttributes:professionalAssociations:+[Paràmetre]		
Paràmetre	Descripció	Tipus
<i>ProfessionalAssociationName</i>	No disponible	Particular CATCERT
<i>ProfessionalAssociationInitials</i>	No disponible	Particular CATCERT
<i>ProfessionalAssociationNumber</i>	No disponible	Particular CATCERT
<i>ProfessionalAssociationZone</i>	No disponible	Particular CATCERT
<i>ProfessionalAssociationEmployeeNumber</i>	No disponible	Particular CATCERT
<i>ProfessionalAssociationCIF</i>	No disponible	Particular CATCERT

urn:catcert:psis:certificateAttributes:+[Paràmetre]		
Paràmetre	Descripció	Tipus
<i>KeyOwnerNIF</i>	Consulta del NIF incorporat al certificat	Particular CATCERT
<i>Title</i>	No disponible	Particular CATCERT
<i>LegalEntityCIF</i>	No disponible	Particular CATCERT
<i>LegalEntityGlobalCIF</i>	No disponible	Particular CATCERT
<i>SubjectName</i>	No disponible	Particular CATCERT
<i>ClassificationLevel</i>	No disponible	Particular CATCERT
<i>Attribute</i>	No disponible	Particular CATCERT
<i>Department</i>	No disponible	Particular CATCERT
<i>QuantitativeUsageLimitations</i>	No disponible	Particular CATCERT
<i>QualitativeUsageLimitations</i>	No disponible	Particular CATCERT

CertIssuerName	No disponible	Particular CATCERT
----------------	---------------	--------------------

## 9.6 Atributs de consulta d'una signatura

urn:OASIS:names:tc:dss:1.0:profiles:XSS:signatureAttributes:+[Paràmetre]	
Paràmetre	Descripció
DigestAlgorithm	Sol·licitud de consulta de l'algoritme utilitzat per a generar el valor de Digest
DigestEncryptionAlgorithm	Algoritme d'encryptació aplicat al digest per a generar la signatura
SignatureAlgorithm	Sol·licitud de consulta de l'algoritme utilitzat per a generar la signatura
SignatureValue	Sol·licitud de consulta del valor de signatura

## 9.7 Esquema del protocol DSS i el seu perfil XSS

Protocol DSS
<pre> &lt;?XML version="1.0" encoding="UTF-8"?&gt; &lt;xs:schema Xmlns:dss="urn:OASIS:names:tc:dss:1.0:core:schema"   Xmlns:ds="http://www.w3.org/2000/09/XMLDsig#"   Xmlns:xs="http://www.w3.org/2001/XMLSchema"   Xmlns:saml="urn:OASIS:names:tc:SAML:1.0:assertion"   targetNamespace="urn:OASIS:names:tc:dss:1.0:core:schema" elementFormDefault="qualified"   attributeFormDefault="unqualified"&gt;   &lt;!-- --&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation XML:lang="en"&gt;       This Schema defines the Digital Signature Service Core Protocols,       Elements, and Bindings Working Draft 34     &lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;!-- --&gt;   &lt;xs:import namespace="http://www.w3.org/2000/09/XMLDsig#"     schemaLocation="http://www.w3.org/TR/XMLDsig-core/XMLDsig-core-schema.xsd"/&gt;   &lt;xs:import namespace="urn:OASIS:names:tc:SAML:1.0:assertion"     schemaLocation="http://www.OASIS-open.org/committees/download.php/3408/OASIS-sstc-saml-     schema-protocol-1.1.xsd"/&gt;   &lt;xs:import namespace="http://www.w3.org/XML/1998/namespace"     schemaLocation="http://www.w3.org/2001/XML.xsd"/&gt;   &lt;!-- COMMON PROTOCOL STRUCTURES --&gt;   &lt;xs:complexType name="AnyType"&gt;     &lt;xs:annotation&gt;       &lt;xs:documentation XML:lang="en"&gt;         This Type type is used to match optional inputs, optional         outputs and to make the Schema extensible where         &lt;xs:any namespace="##other" processContents="lax"/&gt;         is not possible due to unique particle attribution rules.       &lt;/xs:documentation&gt;     &lt;/xs:annotation&gt;     &lt;xs:sequence&gt;       &lt;xs:any processContents="lax" minOccurs="0" maxOccurs="unbounded"/&gt;     &lt;/xs:sequence&gt;   &lt;/xs:complexType&gt;   &lt;!-- --&gt;   &lt;xs:complexType name="InlineXMLType"&gt;     &lt;xs:annotation&gt;       &lt;xs:documentation XML:lang="en"&gt;         This Type clearly expresses the fact that content of         InlineXML should be       &lt;/xs:documentation&gt;     &lt;/xs:annotation&gt;   &lt;/xs:complexType&gt; </pre>

<p>one</p> <p>Comments before</p> <p>ignorePisComments indicates</p> <p>will also</p> <p>PI's and</p> <p>would have</p>	<p>equivalent to a complete XML Document. I.e. having only</p> <p>DocumentElement and not allowing anything but PI's and</p> <p>and after this one element. The attribute</p> <p>how to deal with PI's and Comments as a number of parsers</p> <p>ignore them and a server will have to be able to know if</p> <p>Comments have gone missing after parsing and if the client</p> <p>wanted them to be signed.</p> <pre> &lt;/xs:documentation&gt; &lt;/xs:annotation&gt; &lt;xs:sequence&gt;   &lt;xs:any processContents="lax"/&gt; &lt;/xs:sequence&gt; &lt;xs:attribute name="ignorePis" type="xs:boolean" use="optional" default="true"/&gt; &lt;xs:attribute name="ignoreComments" type="xs:boolean" use="optional" default="true"/&gt; &lt;/xs:complexType&gt; &lt;!-- --&gt; &lt;xs:complexType name="InternationalStringType"&gt;   &lt;xs:simpleContent&gt;     &lt;xs:extension base="xs:string"&gt;       &lt;xs:attribute ref="XML:lang" use="required"/&gt;     &lt;/xs:extension&gt;   &lt;/xs:simpleContent&gt; &lt;/xs:complexType&gt; &lt;!-- --&gt; &lt;xs:element name="InputDocuments"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation XML:lang="en"&gt;       &lt;!-- Re: UPA Problem rationale behind these changes [FW: FROM JC THROUGH KONRAD] --&gt;       &lt;!--       &lt;xs:any namespace="##other" processContents="lax"/&gt; allows to introduce new top level elements from other namespaces --&gt;       &lt;!-- Solution consistent with other places --&gt;       &lt;xs:element name="Other" type="dss:AnyType"/&gt; allows to introduce new top level elements from namespaces including future. dss to support other types of input documents in the future.         </pre>
<pre> &lt;/xs:documentation&gt; &lt;/xs:annotation&gt; &lt;xs:complexType&gt;   &lt;xs:sequence&gt;     &lt;xs:choice maxOccurs="unbounded"&gt;       &lt;xs:element ref="dss:Document"/&gt;       &lt;xs:element ref="dss:TransformedData"/&gt;       &lt;xs:element ref="dss:DocumentHash"/&gt;       &lt;xs:element name="Other" type="dss:AnyType"/&gt;     &lt;/xs:choice&gt;   &lt;/xs:sequence&gt; &lt;/xs:complexType&gt; &lt;/xs:element&gt; &lt;!-- --&gt; &lt;xs:complexType name="DocumentBaseType" abstract="true"&gt;   &lt;xs:attribute name="ID" type="xs:ID" use="optional"/&gt;   &lt;xs:attribute name="RefURI" type="xs:anyURI" use="optional"/&gt;   &lt;xs:attribute name="RefType" type="xs:anyURI" use="optional"/&gt;   &lt;xs:attribute name="SchemaRefs" type="xs:IDREFS" use="optional"/&gt; &lt;/xs:complexType&gt; &lt;!-- --&gt; &lt;xs:element name="Document" type="dss:DocumentType"/&gt; &lt;xs:complexType name="DocumentType"&gt;   &lt;xs:complexContent&gt;     &lt;xs:extension base="dss:DocumentBaseType"&gt;       &lt;xs:sequence&gt; </pre>	

```

        <xs:choice>
            <xs:element name="InlineXML"
type="dss:InlineXMLType"/>
            <xs:element name="Base64XML"
type="xs:base64Binary"/>
            <xs:element name="EscapedXML"
type="xs:string"/>
            <xs:element ref="dss:Base64Data"/>
        </xs:choice>
    </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<!-- -->
<xs:element name="Base64Data">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:base64Binary">
                <xs:attribute name="MimeType" type="xs:string"
use="optional"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="DocumentHash">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="dss:DocumentBaseType">
                <xs:sequence>
                    <xs:element ref="ds:Transforms"
minOccurs="0"/>
                    <xs:element ref="ds:DigestMethod"/>
                    <xs:element ref="ds:DigestValue"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="TransformedData">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="dss:DocumentBaseType">
                <xs:sequence>
                    <xs:element ref="ds:Transforms"
minOccurs="0"/>
                    <xs:element ref="dss:Base64Data"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="SignatureObject" type="dss:SignatureObjectType"/>
<xs:complexType name="SignatureObjectType">
    <xs:annotation>
        <xs:documentation XML:lang="en">
            &lt;xs:any namespace="##other" processContents="lax"/&gt;
            is not
            namespace
            (i.e. ds, dss) are used in the choice hence
            &lt;xs:element name="Other" type="dss:AnyType"/&gt;
            allows to introduce new top level elements from
            namespaces including
            dss to support other types of signatures in the future.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:choice>
            <xs:element ref="ds:Signature"/>
            <xs:element ref="dss:Timestamp"/>
            <xs:element ref="dss:Base64Signature"/>
            <xs:element ref="dss:SignaturePtr"/>

```

```

        <xs:element name="Other" type="dss:AnyType"/>
    </xs:choice>
</xs:sequence>
<xs:attribute name="SchemaRefs" type="xs:IDREFS" use="optional"/>
</xs:complexType>
<!-- -->
<xs:element name="Base64Signature">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:base64Binary">
                <xs:attribute name="Type" type="xs:anyURI"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="SignaturePtr">
    <xs:complexType>
        <xs:attribute name="WhichDocument" type="xs:IDREF"/>
        <xs:attribute name="XPath" type="xs:string" use="optional"/>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="Result">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="ResultMajor" type="xs:anyURI"/>
            <xs:element name="ResultMinor" type="xs:anyURI"
minOccurs="0"/>
            <xs:element name="ResultMessage"
type="dss:InternationalStringType" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="OptionalInputs" type="dss:AnyType">
    <xs:annotation>
        <xs:documentation XML:lang="en">
            "dss:AnyType"/> matches any top level element of any
elements.
            It should however not contain elements that are not
declared as
optional inputs by normative text of the dss-core or dss-
profiles.
        </xs:documentation>
    </xs:annotation>
</xs:element>
<!-- -->
<xs:element name="OptionalOutputs" type="dss:AnyType">
    <xs:annotation>
        <xs:documentation XML:lang="en">
            "dss:AnyType"/> matches any top level element of any
elements.
            It should however not contain elements that are not
declared as
optional outputs by normative text of the dss-core or dss-
profiles.
        </xs:documentation>
    </xs:annotation>
</xs:element>
<!-- -->
<xs:element name="ServicePolicy" type="xs:anyURI"/>
<!-- -->
<xs:element name="ClaimedIdentity">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Name" type="saml:NameIdentifierType"/>
            <xs:element name="SupportingInfo" type="dss:AnyType"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- -->

```

```

<xs:element name="Language" type="xs:language"/>
<!-- -->
<xs:element name="AdditionalProfile" type="xs:anyURI"/>
<!-- COMMON PROTOCOL STRUCTURES -->
<!-- PROTOCOL MESSAGES BEGIN -->
<!-- -->
<xs:complexType name="RequestBaseType">
  <xs:sequence>
    <xs:element ref="dss:OptionalInputs" minOccurs="0"/>
    <xs:element ref="dss:InputDocuments"/>
  </xs:sequence>
  <xs:attribute name="RequestID" type="xs:string" use="optional"/>
  <xs:attribute name="Profile" type="xs:anyURI" use="optional"/>
</xs:complexType>
<!-- -->
<xs:element name="SignRequest">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="dss:RequestBaseType">
        <xs:attribute name="Type" type="xs:anyURI"/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="IncludeObject">
  <xs:complexType>
    <xs:attribute name="WhichDocument" type="xs:IDREF"/>
    <xs:attribute name="hasObjectTagsAndAttributesSet"
type="xs:boolean" default="false"/>
    <xs:attribute name="ObjId" type="xs:string" use="optional"/>
    <xs:attribute name="createReference" type="xs:boolean"
use="optional" default="true"/>
  </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="SignaturePlacement">
  <xs:complexType>
    <xs:sequence>
      <xs:choice>
        <xs:element name="XPathAfter" type="xs:string"/>
        <xs:element name="XPathFirstChildOf"
type="xs:string"/>
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="WhichDocument" type="xs:IDREF"/>
    <xs:attribute name="createEnvelopedSignature" type="xs:boolean"
default="true"/>
  </xs:complexType>
</xs:element>
<!-- -->
<xs:complexType name="ResponseBaseType">
  <xs:sequence>
    <xs:element ref="dss:Result"/>
    <xs:element ref="dss:OptionalOutputs" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="RequestID" type="xs:string" use="optional"/>
  <xs:attribute name="Profile" type="xs:anyURI" use="required"/>
</xs:complexType>
<!-- -->
<xs:element name="Response" type="dss:ResponseBaseType"/>
<!-- -->
<xs:element name="SignResponse">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="dss:ResponseBaseType">
        <xs:sequence>
          <xs:element ref="dss:SignatureObject"
minOccurs="0"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!-- -->

```

<pre> &lt;!-- SIGNRESPONSE OPTIONAL OUTPUTS START --&gt; &lt;xs:element name="DocumentWithSignature"&gt;   &lt;xs:complexType&gt;     &lt;xs:sequence&gt;       &lt;xs:element ref="dss:Document"/&gt;     &lt;/xs:sequence&gt;   &lt;/xs:complexType&gt; &lt;/xs:element&gt; &lt;!-- SIGNRESPONSE OPTIONAL OUTPUTS END --&gt; &lt;xs:element name="VerifyRequest"&gt;   &lt;xs:complexType&gt;     &lt;xs:complexContent&gt;       &lt;xs:extension base="dss:RequestBaseType"&gt;         &lt;xs:sequence&gt;           &lt;xs:element ref="dss:SignatureObject" minOccurs="0"/&gt;         &lt;/xs:sequence&gt;       &lt;/xs:extension&gt;     &lt;/xs:complexContent&gt;   &lt;/xs:complexType&gt; &lt;/xs:element&gt; &lt;!-- --&gt; &lt;xs:element name="VerifyResponse"&gt;   &lt;xs:complexType&gt;     &lt;xs:complexContent&gt;       &lt;xs:extension base="dss:ResponseBaseType"/&gt;     &lt;/xs:complexContent&gt;   &lt;/xs:complexType&gt; &lt;/xs:element&gt; &lt;!-- --&gt; &lt;!-- PROTOCOL MESSAGES END --&gt; &lt;!-- SIGNREQUEST OPTIONAL INPUTS START --&gt; &lt;xs:element name="SignatureType" type="xs:anyURI"/&gt; &lt;xs:element name="AddTimestamp"&gt;   &lt;xs:complexType&gt;     &lt;xs:attribute name="Type" type="xs:anyURI" use="optional"/&gt;   &lt;/xs:complexType&gt; &lt;/xs:element&gt; &lt;!-- --&gt; &lt;xs:element name="IntendedAudience"&gt;   &lt;xs:complexType&gt;     &lt;xs:sequence&gt;       &lt;xs:element name="Recipient" type="saml:NameIdentifierType" maxOccurs="unbounded"/&gt;     &lt;/xs:sequence&gt;   &lt;/xs:complexType&gt; &lt;/xs:element&gt; &lt;!-- --&gt; &lt;xs:element name="KeySelector"&gt;   &lt;xs:annotation&gt;     &lt;xs:documentation XML:lang="en"&gt;       &lt;!-- is not than namespaces including target namespace. other namespaces however as we cannot --&gt;       &lt;!-- possible here to allow extensibility as another namespace the target namespace is used in the choice hence allows to introduce new top level elements from dss to support other types of key selectors in the future. Note that namespace="##other" is the complement of the Note also that XML schema does not support complements for or sets of namespaces which is a defect in XML schema. It only supports sets of namespaces which is not useful know which namespaces might be relevant in the future. --&gt;     &lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:complexType&gt;     &lt;xs:sequence&gt;       &lt;xs:choice&gt;         &lt;xs:element ref="ds:KeyInfo"/&gt;         &lt;xs:element name="Other" type="dss:AnyType"/&gt;       &lt;/xs:choice&gt;     &lt;/xs:sequence&gt;   &lt;/xs:complexType&gt; -- </pre>	<pre>       &lt;!-- is not than namespaces including target namespace. other namespaces however as we cannot --&gt;       &lt;!-- possible here to allow extensibility as another namespace the target namespace is used in the choice hence allows to introduce new top level elements from dss to support other types of key selectors in the future. Note that namespace="##other" is the complement of the Note also that XML schema does not support complements for or sets of namespaces which is a defect in XML schema. It only supports sets of namespaces which is not useful know which namespaces might be relevant in the future. --&gt;     &lt;/xs:documentation&gt;   &lt;/xs:annotation&gt;   &lt;xs:complexType&gt;     &lt;xs:sequence&gt;       &lt;xs:choice&gt;         &lt;xs:element ref="ds:KeyInfo"/&gt;         &lt;xs:element name="Other" type="dss:AnyType"/&gt;       &lt;/xs:choice&gt;     &lt;/xs:sequence&gt;   &lt;/xs:complexType&gt; -- </pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

```

        </xs:choice>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!-- -->
<xs:element name="SignedReferences">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="dss:SignedReference"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="Properties">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="SignedProperties"
type="dss:PropertiesType" minOccurs="0"/>
            <xs:element name="UnsignedProperties"
type="dss:PropertiesType" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="Property">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Identifier" type="xs:anyURI"/>
            <xs:element name="Value" type="dss:AnyType"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:complexType name="PropertiesType">
    <xs:sequence>
        <xs:element ref="dss:Property" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<!-- -->
<xs:element name="SignedReference">
    <xs:annotation>
        <xs:documentation XML:lang="en">
            RefURI overrides the of <code>dss:Document</code>
        </xs:documentation>
    </xs:annotation>
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="ds:Transforms" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="WhichDocument" type="xs:IDREF"
use="required"/>
        <xs:attribute name="RefURI" type="xs:anyURI" use="optional"/>
        <xs:attribute name="RefId" type="xs:string" use="optional"/>
    </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="Schema" type="dss:DocumentType"/>
<!-- -->
<xs:element name="Schemas" type="dss:SchemasType"/>
<xs:complexType name="SchemasType">
    <xs:sequence>
        <xs:element ref="dss:Schema" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<!-- SIGNREQUEST OPTIONAL INPUTS END -->
<!-- VERIFYREQUEST OPTIONAL INPUTS START -->
<xs:element name="VerifyManifests"/>
<xs:element name="VerificationTime" type="xs:dateTime"/>
<xs:element name="AdditionalKeyInfo">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="ds:KeyInfo"/>
        </xs:sequence>
    </xs:complexType>

```

```

        </xs:complexType>
    </xs:element>
    <!-- -->
    <xs:element name="ReturnProcessingDetails"/>
    <!-- -->
    <xs:element name="ReturnSigningTime"/>
    <!-- -->
    <xs:element name="ReturnTimestampTime"/>
    <!-- -->
    <xs:element name="ReturnSignerIdentity"/>
    <!-- -->
    <xs:element name="ReturnUpdatedSignature">
        <xs:complexType>
            <xs:attribute name="Type" type="xs:anyURI" use="optional"/>
        </xs:complexType>
    </xs:element>
    <!-- -->
    <xs:element name="ReturnTransformedDocument">
        <xs:complexType>
            <xs:attribute name="WhichReference" type="xs:integer"
use="required"/>
        </xs:complexType>
    </xs:element>
    <!-- VERIFYREQUEST OPTIONAL INPUTS END -->
    <!-- VERIFYRESPONSE OPTIONAL OUTPUTS START -->
    <xs:element name="ProcessingDetails">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="ValidDetail" type="dss:DetailType"
minOccurs="0" maxOccurs="unbounded"/>
                <xs:element name="IndeterminateDetail"
type="dss:DetailType" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element name="InvalidDetail" type="dss:DetailType"
minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <!-- -->
    <xs:element name="SigningTime">
        <xs:complexType>
            <xs:simpleContent>
                <xs:extension base="xs:dateTime">
                    <xs:attribute name="ThirdPartyTimestamp"
type="xs:boolean" use="required"/>
                </xs:extension>
            </xs:simpleContent>
        </xs:complexType>
    </xs:element>
    <!-- -->
    <xs:element name="TimestampTime" type="xs:dateTime"/>
    <!-- -->
    <xs:element name="SignerIdentity" type="saml:NameIdentifierType"/>
    <!-- -->
    <xs:element name="UpdatedSignature">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="dss:SignatureObject"/>
            </xs:sequence>
            <xs:attribute name="Type" type="xs:anyURI" use="optional"/>
        </xs:complexType>
    </xs:element>
    <!-- -->
    <xs:element name="TransformedDocument">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="dss:Document"/>
            </xs:sequence>
            <xs:attribute name="WhichReference" type="xs:integer"
use="required"/>
        </xs:complexType>
    </xs:element>
    <!-- -->
    <xs:complexType name="DetailType">
        <xs:sequence>
            <xs:element name="Code" type="xs:anyURI" minOccurs="0"/>

```

```

    <xs:element name="Message" type="dss:InternationalStringType"
minOccurs="0" />
    <xs:any namespace="##other" minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>
  <xs:attribute name="Type" type="xs:anyURI" use="required" />
</xs:complexType>
<!-- VERIFYRESPONSE OPTIONAL OUTPUTS END -->
<!-- TIMESTAMP BEGIN -->
<xs:element name="Timestamp">
  <xs:complexType>
    <xs:sequence>
      <xs:choice>
        <xs:element ref="ds:Signature" />
        <xs:element name="RFC3161TimeStampToken"
type="xs:base64Binary" />
        <xs:element name="Other" type="dss:AnyType" />
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- -->
<xs:element name="TstInfo">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SerialNumber" type="xs:integer" />
      <xs:element name="CreationTime" type="xs:dateTime" />
      <xs:element name="Policy" type="xs:anyURI" minOccurs="0" />
      <xs:element name="ErrorBound" type="xs:duration"
minOccurs="0" />
      <xs:element name="Ordered" type="xs:boolean"
default="false" minOccurs="0" />
      <xs:element name="TSA" type="saml:NameIdentifierType"
minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- TIMESTAMP END -->
<!-- REQUESTER IDENTITY BEGIN -->
<xs:element name="RequesterIdentity">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Name" type="saml:NameIdentifierType" />
      <xs:element name="SupportingInfo" type="dss:AnyType"
minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- REQUESTER IDENTITY END -->
<xs:element name="VerifyManifestResults" type="dss:VerifyManifestResultsType" />
<xs:complexType name="VerifyManifestResultsType">
  <xs:sequence>
    <xs:element ref="dss:ManifestResult" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:element name="ManifestResult">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ReferenceXpath" type="xs:string" />
      <xs:element name="Status" type="xs:anyURI" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

#### Protocol XSS

```

<?XML version="1.0" encoding="UTF-8"?>
<!-- XSS Profile of the OASIS DSS Schema v1.0-->
<!--Author: Carlos González-Cadenas-->
<!--Date: December 2005-->
<xs:schema targetNamespace="urn:OASIS:names:tc:dss:1.0:profiles:XSS"

```

```

Xmlns:archp="urn:OASIS:names:tc:dss:1.0:profiles:archive"
Xmlns:xsp="http://uri.etsi.org/2038/v1.1.1#" Xmlns:tsl="http://uri.etsi.org/02231/v1.0bis
2005-04#" Xmlns:dss="http://www.docs.OASIS-open.org/dss/OASIS-dss-1.0-core-schema-cd-02.xsd"
Xmlns:xs="http://www.w3.org/2001/XMLSchema" Xmlns:ds="http://www.w3.org/2000/09/XMLDsig#"
Xmlns:XAdES="http://uri.etsi.org/01903/v1.2.2#"
Xmlns:saml20="urn:OASIS:names:tc:SAML:2.0:assertion"
Xmlns="urn:OASIS:names:tc:dss:1.0:profiles:XSS" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:import namespace="http://uri.etsi.org/02231/v1.0bis 2005-04#"
schemaLocation="TS101231v1_2_1.xsd"/>
  <xs:import namespace="urn:OASIS:names:tc:dss:1.0:profiles:archive"
schemaLocation="OASIS-dss-1.0-profiles-archive-schema-wd01-errata01.xsd"/>
  <xs:import namespace="urn:OASIS:names:tc:SAML:2.0:assertion"
schemaLocation="http://docs.OASIS-open.org/security/saml/v2.0/saml-schema-assertion-
2.0.xsd"/>
  <xs:import namespace="http://uri.etsi.org/2038/v1.1.1#"
schemaLocation="SigPolicy.xsd"/>
  <xs:element name="SignaturePolicy">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="XAdES:ObjectIdentifierType">
          <xs:attribute name="allowPolicyMappings"
type="xs:boolean" use="optional" default="false"/>
        </xs:extension>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <xs:element name="SignaturePolicyInfo">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="SignaturePolicyIssuer" type="xs:string"/>
        <xs:element name="SignaturePolicyIdentifier"
type="XAdES:ObjectIdentifierType"/>
        <xs:element name="SignaturePolicyDigestAlgorithm"
type="XAdES:ObjectIdentifierType"/>
        <xs:element name="SignaturePolicyDigestValue"
type="ds:DigestValueType"/>
        <xs:element ref="ds:Transforms" minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ReturnSignedResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="RequiredCommitments" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="CommitmentType"
type="xsp:CommitmentType" maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ResponseSignature">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="ds:Signature"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ReturnSignatureInfo">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="AttributeDesignator"
type="saml20:AttributeType" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SignatureInfo">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Attribute" type="saml20:AttributeType"
maxOccurs="unbounded"/>

```

```

        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:complexType name="BinaryAttributeValueType">
      <xs:simpleContent>
        <xs:extension base="xs:base64Binary">
          <xs:attribute name="Attribute" type="xs:anyURI"
use="required"/>
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
    <xs:element name="ReturnX509CertificateInfo">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="AttributeDesignator"
type="saml20:AttributeType" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="X509CertificateInfo">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Attribute" type="saml20:AttributeType"
maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="Scheme">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="SchemeName"
type="tsl:InternationalNamesType"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="SchemeInfo">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="SchemeName"
type="tsl:InternationalNamesType"/>
          <xs:element name="TSLSequenceNumber" type="xs:integer"/>
          <xs:element name="TSLDigestAlgorithm"
type="XAAdES:ObjectIdentifierType"/>
          <xs:element name="TSLDigestValue" type="ds:DigestValueType"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="X509CertificateValidationOptions"
type="xsp:CertificateTrustTreesType"/>
    <xs:element name="RequireQualifiedCertificate"/>
    <xs:element name="Archive">
      <xs:complexType>
        <xs:sequence>
          <xs:choice>
            <xs:element ref="archp:ArchivePolicy" minOccurs="0"/>
            <xs:element ref="archp:RetentionPeriod" minOccurs="0"/>
          </xs:choice>
          <xs:element ref="archp:UpdateSignature" minOccurs="0"/>
          <xs:element ref="archp:ArchiveMode" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="ArchiveInfo">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="ArchiveIdentifier"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="CounterSignature">
      <xs:complexType>
        <xs:attribute name="WhichDocument" type="xs:IDREF" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="ParallelSignature"/>

```

---

```
</xs:schema>
```

---

