



1

2

3

---

# J2ME Code-Signing Profile of the OASIS Digital Signature Services

4

## Committee Draft, 24 December 2004 (Working Draft 03)

5

6

### Document identifier:

7

oasis-dss-1.0-profiles-codesigning-j2me-spec-cd-01

8

### Location:

9

<http://docs.oasis-open.org/dss/>

10

### Editor:

11

Pieter Kasselmann, *Cybertrust* <[pieter.kasselmann@cybertrust.com](mailto:pieter.kasselmann@cybertrust.com)>

12

### Contributors:

13

*Trevor Perrin*

14

### Abstract:

15

This draft profiles the OASIS DSS core protocols and the OASIS DSS Abstract Code-Signing Profile for the purpose of creating J2ME code-signing signatures.

16

17

### Status:

18

This is a **Committee Draft** produced by the OASIS Digital Signature Service Technical Committee. Committee members should send comments on this draft to [dss@lists.oasis-open.org](mailto:dss@lists.oasis-open.org).

19

20

21

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Digital Signature Service TC web page at <http://www.oasis-open.org/committees/dss/ipr.php>.

22

23

24

25

---

25 **Table of Contents**

26 1 Introduction ..... 3  
27 1.1 Notation ..... 3  
28 1.2 Namespaces ..... 3  
29 1.3 Overview (Non-normative) ..... 3  
30 2 Profile Features..... 5  
31 2.1 Identifier..... 5  
32 2.2 Scope ..... 5  
33 2.3 Relationship To Other Profiles ..... 5  
34 2.4 Signature Object..... 5  
35 2.5 Transport Binding..... 5  
36 2.6 Security Binding ..... 5  
37 3 Profile of Signing Protocol..... 6  
38 3.1 Element <dss:SignRequest>..... 6  
39 3.1.1 Element <dss:OptionalInputs>..... 6  
40 3.1.2 Element <dss:InputDocuments>..... 6  
41 3.2 Element <dss:SignResponse> ..... 7  
42 3.2.1 Element <dss:Result> ..... 7  
43 3.2.2 Element <dss:OptionalOutputs>..... 7  
44 3.2.3 Element <dss:SignatureObject> ..... 8  
45 4 Profile of Verifying Protocol..... 9  
46 5 Profile of J2ME MIDP 2.0 Signatures ..... 10  
47 6 Profile of Server Processing Rules ..... 11  
48 7 Profile of Client Processing Rules ..... 12  
49 8 Editorial Issues..... 13  
50 9 References..... 14  
51 9.1 Normative ..... 14  
52 Appendix A. Revision History ..... 15  
53 Appendix B. Notices ..... 16

---

## 54 1 Introduction

55 The DSS signing and verifying protocols are defined in [DSS Core] and the code-signing profile  
56 of the DSS signing and verification protocols are defined in [DSS CS]. As defined in those  
57 documents, these protocols have a fair degree of flexibility and extensibility. This document  
58 profiles these protocols to limit their flexibility and extend them in concrete ways. It also profiles  
59 the processing rules followed by clients and servers when using these protocols, and profiles the  
60 J2ME signature format for use with these protocols. The resulting profile is suitable for  
61 implementation and interoperability.

62 The following sections describe how to understand the rest of this document.

### 63 1.1 Notation

64 The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD",  
65 "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this specification are to be  
66 interpreted as described in IETF RFC 2119 [RFC 2119]. These keywords are capitalized when  
67 used to unambiguously specify requirements over protocol features and behavior that affect the  
68 interoperability and security of implementations. When these words are not capitalized, they are  
69 meant in their natural-language sense.

70 This specification uses the following typographical conventions in text: <ns:Element>,  
71 Attribute, **Datatype**, OtherCode.

### 72 1.2 Namespaces

73 The structures described in this specification are contained in the schema file [J2ME-CS-XSD].  
74 All schema listings in the current document are excerpts from the schema file. In the case of a  
75 disagreement between the schema file and this document, the schema file takes precedence.

76 This schema is associated with the following XML namespace:

```
77 urn:oasis:names:tc:dss:1.0:profiles:codesigning:1.0:J2ME:1.0
```

78 If a future version of this specification is needed, it will use a different namespace.

79 Conventional XML namespace prefixes are used in this document:

- 80 • The prefix `dsscscj2me:` (or no prefix) stands for the DSS code-signing namespace [CS-  
81 XSD].
- 82 • The prefix `dsscsc:` stands for the DSS code-signing namespace [CS-XSD].
- 83 • The prefix `asyncc:` stands for this profiles namespace [Async-XSD].
- 84 • The prefix `dssc:` stands for the DSS core namespace [Core-XSD].
- 85 • The prefix `ds:` stands for the W3C XML Signature namespace [XMLSig].

86 Applications MAY use different namespace prefixes, and MAY use whatever namespace  
87 defaulting/scoping conventions they desire, as long as they are compliant with the Namespaces  
88 in XML specification [XML-ns].

### 89 1.3 Overview (Non-normative)

90 The [DSS-CS] abstract profile provides a profile of [DSS-Core] and combines it with the [DSS-  
91 Async] profile. The [DSS-CS] profile allow for the generation of signatures on content, including

92 software programs, and is flexible enough to accommodate the typical scenarios encountered in  
93 the software development lifecycle.

94 This specification provides a concrete profile based on **[DSS-CS]** for requesting the generation of  
95 signatures as specified in the Java 2 Micro Edition (J2ME), Mobile Information Device Profile 2.0  
96 **[MIDP 2.0]**.

---

## 97 2 Profile Features

### 98 2.1 Identifier

99 urn:oasis:names:tc:dss:1.0:profiles:codesigning:1.0:J2ME:1.0

### 100 2.2 Scope

101 This document further profiles the abstract profile for code-signing as described in [DSS CS],  
102 which is a profile of the DSS signing protocol defined in [DSS Core] in combination with [DSS  
103 Async].

### 104 2.3 Relationship To Other Profiles

105 This profile is a concrete profile of the abstract code-signing profile defined in [DSS CS].

### 106 2.4 Signature Object

107 This profile supports the creation of signatures as defined in [MIDP 2.0]. [MIDP 2.0] defines the  
108 use of EMSA-PKCS1-v1\_5 as defined in [RFC 2437].

### 109 2.5 Transport Binding

110 This profile is transported using the HTTP POST Transport Binding defined in [DSS Core].

### 111 2.6 Security Binding

112 This profile is secured using the TLS X.509 Mutual Authentication Binding defined in [DSS Core].

---

## 113 3 Profile of Signing Protocol

### 114 3.1 Element <dss:SignRequest>

#### 115 3.1.1 Element <dss:OptionalInputs>

116 Optional inputs MUST be used as defined in [DSS CS].

117 The following optional inputs defined in the [DSS Core] will not be understood by a server  
118 implementing this profile:

- 119 • <dss:AddTimeStamp>
- 120 • <dss:SignedReference>
- 121 • <dss:Properties>
- 122 • <dss:SignaturePlacement>
- 123 • <dss:EnvelopingSignature>

124 In addition the following constraints are placed on the optional inputs as described below.

#### 125 3.1.1.1 Element <dss:SignatureType>

126 The <dss:SignatureType> MUST contain the identifier `urn:ietf:rfc:2437:RSASSA-`  
127 `PKCS1-v1_5`. This refers to PKCS #1 version 1.5 signatures as defined in [RFC 2437].

#### 128 3.1.1.2 Element <dss:ServicePolicy>

129 The <dss:ServicePolicy> SHOULD be used to indicate a specific server signing policy. The  
130 server signing policy is mapped to the recommended security policy for GSM/UMTS compliant  
131 devices in [MIDP 2.0]. The following URIs may be used to specify the service policy and  
132 corresponding domain under which the MIDlet must be signed.

133 For code that should execute in the manufacturer domain use:

134 `urn:oasis:names:tc:dss:1.0:profiles:codesigning:1.0:J2ME:1.0:manufactur`  
135 `er`

136 For code that should execute in the operator domain use:

137 `urn:oasis:names:tc:dss:1.0:profiles:codesigning:1.0:J2ME:1.0:operator`

138 For code that should execute in the trusted third party domain use:

139 `urn:oasis:names:tc:dss:1.0:profiles:codesigning:1.0:J2ME:1.0:trustedisv`

#### 140 3.1.2 Element <dss:InputDocuments>

141 The server MUST accept <dss:Document> inputs and MUST NOT accept  
142 <dss:DocumentHash> inputs. A server that implements this profile MUST respond with a  
143 <dss:ResultMajor> code of

144 `urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError` as defined in [DSS  
145 Core] if it receives a <dss:DocumentHash> input.

146 The <dss:Document> element MUST include the Base64 encoded J2ME JAR file on which the  
147 signature must be calculated within a <dss:Base64Data> element. The `MimeType` attribute

148 MUST be set to `application/java-archive`. Only one `<Document>` element MUST be  
149 submitted.

## 150 **3.2 Element `<dss:SignResponse>`**

### 151 **3.2.1 Element `<dss:Result>`**

152 This profile defines no additional `<dss:ResultMinor>` codes.

### 153 **3.2.2 Element `<dss:OptionalOutputs>`**

154 None of the optional outputs specified in the **[DSS Core]** are precluded in this abstract profile. In  
155 addition this profile defines the following `<dss:OptionalOutputs>`:

- 156 • `<X509CertificatePath>`

157 In addition, the `<dss:OptionalOutputs>` element MAY contain a `<dss:Document>` element.

#### 158 **3.2.2.1 Element `<X509CertificatePath>`**

159 This element defines the certificate path including the certificate containing the public key  
160 required to verify the signature generated on the JAR file submitted by the client and all  
161 intermediary certificates, excluding the root certificate. The client MAY use this information to  
162 determine the appropriate entries in the Java Application Descriptor file (JAD) file that is  
163 distributed with the JAR file containing the MIDP 2.0 application. The server may return multiple  
164 `<X509CertificatePath>` elements. The orders of the `<X509CertificatePath>` elements are  
165 significant. The first `<X509CertificatePath>` element corresponds to the first certificate path,  
166 identified by  $n=1$  in the JAD file, the second `<X509CertificatePath>` element corresponds to  
167 the second certificate path, identified by  $n=2$ , in the JAD file, the  $j$ 'th `<X509CertificatePath>`  
168 element corresponds to the  $j$ 'th certificate path, identified by  $n=j$ , in the JAD file. The  
169 `<X509CertificatePath>` element contains the following elements:

170 `<X509Certificate>`

171 The `<X509Certificate>` element contains a base64-encoded X.509 v3 certificate.  
172 The order of the `<X509Certificate>` elements are significant. The first  
173 `<X509Certificate>` element contains the signing certificate and corresponds to  $m=1$   
174 in the JAD file for the current `<X509CertificatePath>` element, the second  
175 `<X509Certificate>` element contains the first intermediary certificate and  
176 corresponds to  $m=2$  the current `<X509CertificatePath>` element, the  $k$ 'th  
177 `<X509Certificate>` element contains the  $k-1$ 'st intermediary certificate that issued  
178 the  $k-2$ 'nd intermediary cert.

179

```
180 <xs:element name="X509CertificatePath"  
181         type="dsscsj2me:X509CertificatePathType" />  
182  
183 <xs:complexType name="X509CertificatePathType">  
184     <xs:sequence maxOccurs="unbounded">  
185         <xs:element ref="dsscsj2me:X509Certificate" />  
186     </xs:sequence>  
187 </xs:complexType>
```

188

```
189 <xs:element name="X509Certificate"  
190         type="dsscsj2me:X509CertificateType" />  
191  
192 <xs:simpleType name="X509CertificateType">  
193     <xs:restriction base="xs:base64Binary" />  
194 </xs:simpleType>
```

### 195 **3.2.2.2 Element <dss:Documents>**

196 The server MAY include the J2ME JAR file on which the signature was created as an optional  
197 output using the <dss:Documents> element. If the <dss:Document> element is included in  
198 the response as an optional output, it MUST include the Base64 encoded J2ME JAR file within a  
199 <dss:Base64Data> element. The included J2ME JAR file MUST be the file on which the  
200 signature included in the <dss:SignatureObject> was calculated. The MimeType attribute  
201 MUST be set to application/java-archive.

### 202 **3.2.3 Element <dss:SignatureObject>**

203 The server MUST return a Base64 encoded PKCS #1 signature within the <Base64Signature>  
204 element. The <dss:SignatureObject> element MUST NOT contain any other elements.



---

205 **4 Profile of Verifying Protocol**

206 This **[DSS CS]** profile does not provide a profile of the DSS verification messages and  
207 consequently a server implementing this profile **MUST NOT** respond to any  
208 `<dss:VerifyRequest>` messages.

---

209 **5 Profile of J2ME MIDP 2.0 Signatures**

210 The J2ME MIDP 2.0 signature format is fully defined in **[MIDP 2.0]** and no further profiling is  
211 required.

212

213

214

---

215 **6 Profile of Server Processing Rules**

216 The signature must be calculated on the Base64 decoded JAR file. The server processing rules  
217 defined in **[DSS CS]** SHOULD be followed.

---

218 **7 Profile of Client Processing Rules**

219 Client processing rules as defined in **[DSS CS]** SHOULD be followed.



221

---

## 9 References

222

### 9.1 Normative

- 223     **[Core-XSD]**     T. Perrin et al. *DSS Schema*. OASIS, **(MONTH/YEAR TBD)**
- 224     **[DSSCore]**     T. Perrin et al. *Digital Signature Service Core Protocols and Elements*.  
225     OASIS, **(MONTH/YEAR TBD)**
- 226     **[RFC 2119]**     S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*.  
227     IETF RFC 2396, August 1998.  
228     <http://www.ietf.org/rfc/rfc2396.txt>.
- 229     **[XML-ns]**     T. Bray, D. Hollander, A. Layman. *Namespaces in XML*. W3C  
230     Recommendation, January 1999.  
231     <http://www.w3.org/TR/1999/REC-xml-names-19990114>
- 232     **[XMLSig]**     D. Eastlake et al. *XML-Signature Syntax and Processing*. W3C  
233     Recommendation, February 2002.  
234     <http://www.w3.org/TR/1999/REC-xml-names-19990114>
- 235     **[DSS CS]**     Abstract Code-Signing Profile of the OASIS Digital Signature Services  
236     Working Draft 03, 13 October 2004
- 237     **[DSS Async]**    Asynchronous Processing Abstract Profile of the OASIS Digital Signature  
238     Services, Working Draft 04, 21 August 2004
- 239     **[CS-XSD]**     P. Kasselmann, *Codesigning Schema*. OASIS, **(MONTH/YEAR TBD)**
- 240     **[Async-XSD]**    A, Kuehne. *Asynchronous Processing Profile Schema*. OASIS,  
241     **(MONTH/YEAR TBD)**
- 242     **[J2ME-CS-XSD]** P. Kasselmann, *J2ME Codesigning Schema*. OASIS, **(MONTH/YEAR**  
243     **TBD)**
- 244     **[MIDP 2.0]**     Mobile Information Device Profile for Java™ 2 Micro Edition Version 2.0,  
245     JSR 118 Expert Group
- 246     **[RFC 2437]**     RFC 2437 PKCS #1: RSA Cryptography Specifications Version 2.0, B.  
247     Kaliski, J. Staddon, <http://www.ietf.org/rfc/rfc2437.txt>
- 248

---

## Appendix A. Revision History

Rev	Date	By Whom	What
wd-01	2004-07-16	Pieter Kasselmann	Initial version based oasis-dss-1.0-profiles-XYZ-spec-wd-04.doc by Trevor Perrin
wd-02	2004-10-13	Pieter Kasselmann	Revised version includes <X509CertificatePath> element, clerical corrections and refinements.
wd-03	2004-11-24	Pieter Kasselmann	Clerical corrections (name change etc)
cd-01	2004-12-24	Pieter Kasselmann	Approved Committee Draft

---

## Appendix B. Notices

251 OASIS takes no position regarding the validity or scope of any intellectual property or other rights  
252 that might be claimed to pertain to the implementation or use of the technology described in this  
253 document or the extent to which any license under such rights might or might not be available;  
254 neither does it represent that it has made any effort to identify any such rights. Information on  
255 OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS  
256 website. Copies of claims of rights made available for publication and any assurances of licenses  
257 to be made available, or the result of an attempt made to obtain a general license or permission  
258 for the use of such proprietary rights by implementors or users of this specification, can be  
259 obtained from the OASIS Executive Director.

260 OASIS invites any interested party to bring to its attention any copyrights, patents or patent  
261 applications, or other proprietary rights which may cover technology that may be required to  
262 implement this specification. Please address the information to the OASIS Executive Director.

263 Copyright © OASIS Open 2003. *All Rights Reserved.*

264 This document and translations of it may be copied and furnished to others, and derivative works  
265 that comment on or otherwise explain it or assist in its implementation may be prepared, copied,  
266 published and distributed, in whole or in part, without restriction of any kind, provided that the  
267 above copyright notice and this paragraph are included on all such copies and derivative works.  
268 However, this document itself does not be modified in any way, such as by removing the  
269 copyright notice or references to OASIS, except as needed for the purpose of developing OASIS  
270 specifications, in which case the procedures for copyrights defined in the OASIS Intellectual  
271 Property Rights document must be followed, or as required to translate it into languages other  
272 than English.

273 The limited permissions granted above are perpetual and will not be revoked by OASIS or its  
274 successors or assigns.

275 This document and the information contained herein is provided on an "AS IS" basis and OASIS  
276 DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO  
277 ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE  
278 ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A  
279 PARTICULAR PURPOSE.