Specification Information Note

OMA-WAP-211_105-WAPCert-SIN-20020520-a

Version 20-May-2002

for

Open Mobile Alliance
OMA-WAP-211-WAPCert-20010522-a

WAP Certificate and CRL Profiles
Version 22-May-2001

Continues the Technical Activities
Originated in the WAP Forum

.

A list of errata and updates to this document is available from the Open Mobile Alliance™ Web site, http://www.openmobilealliance.org/, in the form of SIN documents, which are subject to revision or removal without notice.

This document is available online in PDF format at http://www.openmobilealliance.org/.

Known problems associated with this document are published at http://www.openmobilealliance.org/.

Comments regarding this document can be submitted to the OMA™ in the manner published at http://www.wapforum.org/.

Contents

1. Scope

This document provides changes and corrections to the following document files:

- OMA-WAP-211-WAPCert-20010522-a

It includes changes from the following change requests:

- CR-VERISIGN-WAPCERT-TLSCERT.DOC

2. Notation

In the subsections describing the changes new text is underlined. Removed text has ~~strikethrough~~ marks. The presented text is copied from the specification. Text that is not presented is not affected at all. The change descriptions may also include editor's notes similar to the one below. The notes are not part of the actual changes and must not be included in the changed text.

Editor's note: Framed notes like these only clarify where and how the changes shall be applied.

3. Note To Process Server Certificates of at least 1500 bytes.

3.1 Change Classification

Class 2 – Bug Fixes

3.2 Change Summary

Add informative text to stress that ME's should have the ability to process TLS server certificates of at least 1500 bytes to ensure broad interoperability with existing TLS servers.

3.3 Change Description

Editor's note: On page 10, add the following text to the first paragraph of section 6.1 "General" as follows

This section defines WAP certificate profiles. The profiles are, unless otherwise mentioned, based on the Internet Certificate Profile [15], which in turn is based on the format defined in [7]. For full implementation of this section implementers are required to consult the underlying format and semantics defined in [7] and [15]. This specification provides, for each certificate type discussed, additional details regarding the contents of some individual fields in the certificate. Certificates issued in conformance with recommendations and requirements in this section will be reasonably compact, and MEs MUST be able to process certificates of size up to at least 700 bytes, while other certificate-processing entities MUST be able to process certificates of size up to at least 2000 bytes. MEs that support X.509-based server authentication MUST be able to process server certificates of size up to at least 1000 bytes and CA certificates of size up to at least 2000 bytes, in addition to requirements listed in Section 6.4, and SHOULD be able to process longer certificates. (**NOTE:** Because many TLS server certificates currently in use have a length over 1000 bytes it is highly recommended that MEs have the ability to process certificates of at least 1500 bytes to ensure broad interoperability with existing servers.) Certificate-processing clients MUST support a certificate chain depth of at least three (i.e., two subordinate CA certificates between the end-entity certificate and the CA root certificate in the chain). A client that encounters a certificate or certificate chain that does not conform to this profile must not fail the certificate processing in an uncontrolled manner. In addition, certificate-processing servers must also support a chain depth of at least three.

4. Note to process server certificates signed with md5WithRSAEncryption

4.1 Change Classification

Class 2 – Bug Fixes

4.2 Change Summary

Add informative text to stress that ME's should have the ability to process TLS server certificates that are signed with md5WithRSAEncryption to ensure broad interoperability with existing TLS servers.

4.3 Change Description

Editor's note: On Page 12, add a new paragraph at the end of Section 6.4.3 "Signature (Algorithm)" as follows

The only signature algorithms defined for use with this profile are **sha1WithRSAEncryption** and **ecdsa-with-SHA1** (see Section 9). Clients MUST support at least one of these algorithms. Clients that support server-authenticated TLS sessions MUST support **sha1WithRSAEncryption**. Clients MUST be able to process certificates signed with keys up to and including 2048 bits (RSA) or 233 bits (EC).

NOTE: Because many TLS server certificates currently in use are signed with the **md5WithRSAEncryption** signature algorithm, it is highly recommended that MEs have the ability to process server certificates signed with this algorithm to ensure broad interoperability with existing servers.