



VeriSign OMA DRMv2 Test PKI Overview



Alex Deacon
Principal Engineer
Advanced Products and Research Group
alex@verisign.com

January 13, 2006



Where it all comes together.™

Introduction

To assist in the interoperability and conformance testing of OMA DRMv2 devices and rights issuers, VeriSign has deployed a test public key infrastructure compatible with the OMA DRMv2 technical standards. This document specifies the details of the test OMA DRMv2 PKI service and indicates how interested parties can enroll for, obtain, manage and check the revocation status for device and rights issuer certificates.

All certificates used in and issued from this infrastructure are governed by the VeriSign Test CPS available from the following URL and thus should not be deployed on servers or devices used to protect content in production grade environments.

<http://www.verisign.com/repository/cps30/test-ca.html>

Any questions or comments regarding this service should be directed to alex@verisign.com

Enrolling for a certificate

Two Managed PKI accounts have been created to handle the certificate lifecycle operations (enrollment, pickup, revocation, search, renewal) for both rights issuer and device certificates.

End user certificate management functionality is available from the following Digital ID Center URLs

Rights Issuer:

<https://pilotsite.verisign.com/services/VeriSignOMADRMv2RITestCA/digitalidCenter.htm>

Device:

<https://pilotsite.verisign.com/services/VeriSignOMADRMv2DeviceTestCA/digitalidCenter.htm>

Two methods of certificate enrollment are available –

Using a Web Browser

Browser based certificate enrollment is supported using the following web browsers: **Internet Explorer**, **Firefox**, **Netscape**¹ and **Opera**²

This method leverages the native PKI capabilities of web browsers and is available by clicking on the “Enroll” link from the relevant URL above. When using this enrollment method you will be prompted to enter your email address, the certificate naming information and a challenge phrase via a web form. Once the “submit” button is selected the browser will generate an RSA key pair and send the public portion of this key along with the naming information to the CA. An email will be sent to the email address specified in the enrollment form confirming the enrollment. At the same time the CA administrator will be

¹ Netscape users must configure their browser to “Display like Firefox” from the Tools→ Options...→ Site Controls menu.

² Opera users must configure their browser to “Identity like Mozilla 5.0” from the Tools→ Preferences...→ Advanced→ Network→ Network Preferences Menu

notified of your enrollment and will approve and issue your certificate. Once issued, you will receive an email message containing instructions on how to retrieve your certificate.

NOTE: You must enroll for and pick up your certificate from the same web browser and machine.

Once your certificate has been delivered and installed in the browser, the certificate and private key can be exported to a PKCS#12 (PFX) for use within your application. If your device or rights issuer requires a raw PKCS#8 private key, see the “Extracting Private Keys and Certificates from a PKCS#12 File” below for details on how this can be accomplished.

Using a PKCS#10 Certificate Signing Request (CSR)

For those devices or rights issuers with the ability to generate an RSA key pair directly, a PKCS#10 enrollment web page is available by selecting the “Enroll for a Digital ID using a CSR”. You will first be asked to point to a base64 encoded PKCS#10 request on your local file system. Once submitted the system will validate your PKCS#10 request and prompt you to enter your email address, the relevant naming information and a challenge phrase. An email will be sent to the address specified in the enrollment form confirming your enrollment. At the same time the CA administrator will be notified of your enrollment and will approve and issue your certificate. .

NOTE: Only the public key data is retrieved from the CSR. As such, you must enter the naming information into the form presented after the CSR has been submitted and validated.

Once issued, you will receive an email message that contains your issued certificate which can then be imported onto your device or into your rights issuer server. The certificate is delivered as a PKCS#7 “certs-only” structure which includes both the end-entity certificate and the certificate of the associated issuing CA. This structure is included in the email message as a base64 encoded blob message in addition to being included as a separate binary attachment.

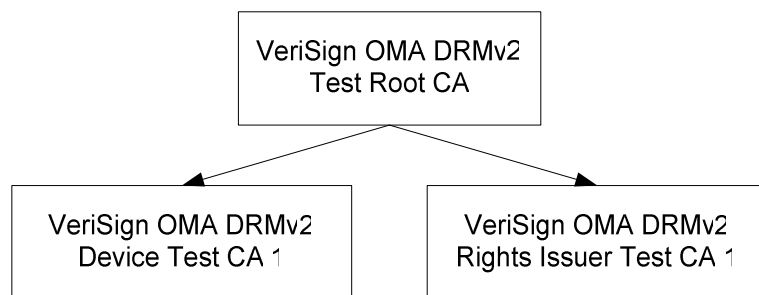
Revoking an End-Entity Certificate

Once issued, a certificate can be revoked by the enrollee by selecting the “Revoke” link from the relevant URL above. You will be asked to first “search” for the certificate you wish to revoke and once found it will be necessary to provide the “challenge phrase” specified during the certificate enrollment process. The CA administrator also has the capabilities to revoke any device or rights issuer certificate issued from the system.

Once revoked, the new status of rights issuer certificates is immediately available from the OCSP responder. Revoked device certificate information will be available on the next CRL update which occurs once every 24 hours.

CA Hierarchy and Certificate Profiles

A simple certificate hierarchy, depicted in the following diagram, has been created. The test PKI is made up of a root certificate and two issuing certificate authorities which have been signed by the root.



The root certificate can be obtained by selecting the “Install CA” link from either of the Digital ID Center URL’s listed above. Intermediate CA certificates are included along with the issued end-entity certificate after approval.

VeriSign OMA DRMv2 Test Root CA	
Issuer Name	CN = VeriSign OMA DRMv2 Test Root CA OU = For Testing Purposes Only O = VeriSign, Inc. C = US
Signature Algorithm	sha1WithRSAEncryption
Subject Name	CN = VeriSign OMA DRMv2 Test Root CA OU = For Testing Purposes Only O = VeriSign, Inc. C = US
Subject Public Key Info	2048-bit RSA
Extensions	SubjectKeyIdentifier (non-critical) Basic Constraints (critical) Key Usage (critical)

VeriSign OMA DRMv2 Device Test CA 1	
Issuer Name	CN = VeriSign OMA DRMv2 Test Root CA OU = For Testing Purposes Only O = VeriSign, Inc. C = US
Signature Algorithm	sha1WithRSAEncryption
Subject Name	CN = VeriSign OMA DRMv2 Device Test CA 1 OU = For Testing Purposes Only

	O = VeriSign, Inc. C = US
Subject Public Key Info	2048-bit RSA
Extensions	SubjectKeyIdentifier (non-critical) Authority Key Identifier (non-critical) Basic Constraints (critical) Key Usage (critical) CRL Distribution Points (non-critical)

VeriSign OMA DRMv2 Rights Issuer Test CA 1	
Issuer Name	CN = VeriSign OMA DRMv2 Test Root CA OU = For Testing Purposes Only O = VeriSign, Inc. C = US
Signature Algorithm	sha1WithRSAEncryption
Subject Name	CN = VeriSign OMA DRMv2 Rights Issuer Test CA 1 OU = For Testing Purposes Only O = VeriSign, Inc. C = US
Subject Public Key Info	2048-bit RSA
Extensions	SubjectKeyIdentifier (non-critical) Authority Key Identifier (non-critical) Basic Constraints (critical) Key Usage (critical) CRL Distribution Points (non-critical)

End-Entity Certificate Profile

The default validity period for device and rights issuer certificates is 90 days.

Rights Issuer Certificates	
Issuer Name	CN = VeriSign OMA DRMv2 Rights Issuer Test CA 1 OU = For Testing Purposes Only O = VeriSign, Inc.

	C = US
Signature Algorithm	sha1WithRSAEncryption
Subject Name	As defined in the OMA DRMv2 TS
Extensions	Authority Key Identifier (non-critical) Basic Constraints (non-critical) Key Usage (critical) Extended Key Usage (critical) OCSP Authority Information Access (non-critical) Certificate Policies (non-critical)

Device Certificates	
Issuer Name	CN = VeriSign OMA DRMv2 Device Test CA 1 OU = For Testing Purposes Only O = VeriSign, Inc. C = US
Signature Algorithm	sha1WithRSAEncryption
Subject Name	As defined in the OMA DRMv2 TS
Extensions	Authority Key Identifier (non-critical) Basic Constraints (non-critical) Key Usage (critical) Extended Key Usage (critical) CRL Distribution Points (non-critical) Certificate Policies (non-critical)

Certificate Validation

Certificate revocation status is available for all certificates within this test PKI hierarchy.

Validation of the issuing CA's revocation status is accomplished via a CRL managed by the root. This CRL is available from the following URL

<http://pilotsitecrl.verisign.com/OfflineCA/VeriSignIncVeriSignOMADRMv2TestRootCA.crl> . A CRL distribution point extension specifying this URL has been added to both the Device and Rights Issuer CA certificates.

Validation of Rights Issuer end-entity certificates is accomplished via the use of OCSP. The address of the OCSP server is <http://pilot-ocsp.verisign.com>. All rights issuer end-entity certificates include a pointer to the OCSP server in an Authority Information Access extension.³

Validation of Device end-entity certificates is accomplished via the use of a CRL managed by the Device CA. The CRL is available from the following URL

<http://pilotonsitecrl.verisign.com/VeriSignOMADRMv2DeviceTestCA/LatestCRL.crl>.

A CRL distribution point extension specifying this URL is included in all device end-entity certificates.

OCSP Responder Certificate	
Issuer Name	CN = VeriSign OMA DRMv2 Rights Issuer Test CA 1 OU = For Testing Purposes Only O = VeriSign, Inc. C = US
Signature Algorithm	sha1WithRSAEncryption
Subject Name	CN =VeriSign OMA DRMv2 Rights Issuer Test OCSP Responder 1 OU =For Testing Purposes Only O =VeriSign, Inc. C =US
Subject Public Key Info	1024-bit RSA
Extensions	Authority Key Identifier (non-critical) Subject Key Identifier (non-critical) Basic Constraints (non-critical) Key Usage (critical) Extended Key Usage (critical) Subject Alternative Name (non-critical)

³ Note that in addition to OCSP, rights issuer certificate status is also available via CRL at <http://pilotonsitecrl.verisign.com/VeriSignOMADRMv2DeviceTestCA/LatestCRL.crl>.

Extracting Private Keys and Certificates from a PKCS#12 File

The OpenSSL tool (<http://www.openssl.org>) can be used to extract certificates and the private key from a PKCS#12 file exported from most web browsers. See <http://www.openssl.org/docs/apps/pkcs12.html> for the OpenSSL documentation on this functionality.

The following OpenSSL command will convert a PKCS#12 file into a "PEM" formatted file that contains the base64 encoded certificate chain and an unencrypted (raw) PKCS#8 format private key.

```
% openssl pkcs12 -nodes -in <infile>.pfx -out <outfile>
```

You will be prompted to enter the password you used to "export" the PKCS#12 file from the browser. If an encrypted PKCS#8 structure is required you can omit the "-nodes" option from the command above. The utility will prompt you to enter a "PEM Passphrase" which will be used to encrypt the private key extracted to the "outfile".



Where it all comes together.™