

OVI Certificates



This article needs to be updated: If you found this article useful, please fix the problems below then delete the `{{ArticleNeedsUpdate}}` template from the article to remove this warning.

Reasons: Ipvalente (18 Apr 2012)

This article contains current information as of 2010.

Introduction

Publishing a Java ME app to OVI requires signing the app. Currently (Jan 2010) OVI accepts Java Verified, VeriSign and Thawte certificates. This wiki attempts to dig a little deeper into which certificates the average app should use and why.

Java Verified

It is assumed that Java Verified is not used in an average app, for the simple reason that software is a fluid product: maintenance releases are the norm. With Java Verified, the developer is not able to sign the app, it must be signed by a third party, thus adding time and expense for each and every release.

VeriSign

A VeriSign code signing certificate can be used to sign a Java App. VeriSign is more expensive than Thawte.

The original Nokia 6555 firmware does not allow any signed Java MIDlet to launch. Upgrading to the latest firmware resolves this issue. According to Ovi Publisher's Guide, Also, VeriSign signed MIDlets cannot be installed on many Nokia 5310 XpressMusic phones. All other handsets featured support VeriSign, including other Series 40 5th Edition, Feature Pack 1 devices.

Thawte

A Thawte code signing certificate can be used to sign a Java app too. It is almost half the price of VeriSign.

As at 27 Jan 2010, The following devices are mentioned as not working with a Thawte certificate:

5320 6210 6650 6555

Certificate Time

A code signing certificate can be purchased for one or two years. That is the time span during which it can be used to sign apps. A handset whose current date/time is set prior to the certificate's start date, will not install the signed app. Similarly a handset whose date/time is set to after the expiration date of the certificate will not install.

OVI expects the certificate end date to be specified.

Under most circumstances, I would expect an app to be installable forever, but the current situation with certificate authorities and their implementation makes this impossible. So it is advisable to purchase a certificate that is valid for two years.

Signing an App

The process of getting a certificate and installing it so that it can be used is quite cryptic. It involves running the java Keytool with lots of parameters from a command prompt, and hoping for the best. Thankfully, VeriSign's website has a very good guide on this, in their support section.

When it comes to signing an app, I switched from the Eclipse Java IDE to Netbeans, purely on the simplicity of signing and obfuscating an app (never managed to get either of these to work in Eclipse).

Related articles

- [Signing your application to run on 3rd Edition phones](#)
- [How to conform with Unified Testing Criteria](#)
- [Nokia Test Criteria for Java™ ME Applications v1.4](#)
- [MIDP 2.0: Signed MIDlet Developer's Guide v2.0](#)

