TC APPLIED TECHNOLOGIES

DICE Software

Driver Signing Guide

Doc Rev 1.1

7/25/2009

TC APPLIED TECHNOLOGIES

CONTENTS

Revision history		
When	Who	What
2008-12-03	CC	1.0 Initial document.
2009-07-10	CC	1.1 Added signing of system components.

1. Overview

As of driver version 3.4.0, the pre-built Windows driver binaries that we distribute are now signed with TC Applied Technologies Ltd's code-signing certificate. The driver's system components, catalog file, and the installer application are all signed. This allows installation on Vista 64-bit. If you'd prefer to have your company's name appear in the Window's driver installation warning dialog box, you will need to re-sign the provided binaries with your company's code-signing certificate. This document explains how.

2. Requirements

For a complete description of the signing process please download and read the **Digital Signatures for Kernel Modules on Systems Running Windows Vista** white paper from Microsoft:

http://www.microsoft.com/whdc/winlogo/drvsign/kmsigning.mspx

More details are also available in the **Kernel-Mode Code Signing Walkthrough** white paper from Microsoft:

http://www.microsoft.com/whdc/winlogo/drvsign/kmcs_walkthrough.mspx

1.) If you don't already have one, you will need to obtain a code-signing certificate from a commercial Certificate Authority (CA). Note that not all CA's code-signing certificates can be used for kernel-mode code signing. We use and recommend Verisign. The below examples are based on using a code-signing certificate issued from Verisign.

2.) Download the corresponding cross-certificate from Microsoft for the CA that issued your code-signing certificate. For Verisign, the file is "MSCV-VSClass3.cer".

http://www.microsoft.com/whdc/winlogo/drvsign/crosscert.mspx

3.) Download and install Windows Driver Kit (WDK) 6001 or later from MSDN. We currently are using WDK 6001.18001.

4.) If you wish to rebuild the installer that we provide, download and install Inno Setup.

http://www.innosetup.com/isinfo.php

3. Configuration

If you haven't already, install your code-signing certificate in your build machine's local Personal certificate store. See page 33 in Microsoft's **Kernel-Mode Code Signing Walkthrough** white paper for details.

Set the environment variable WLHBASE to the base directory of the WDK. In a default installation that would be "C:\WINDDK\6001.18001".

Set the environment variable SELFSIGN_CERT_NAME to your code-signing certificate's subject name. For example, our code-signing certificate's subject name is "TC Applied Technologies Ltd.".

Set the environment variable SELFSIGN_CROSS_CERT to the path of your CA's cross-certificate file. An example for Verisign would be "C:\MSCV-VSClass3.cer".

Set the environment variable SELFSIGN_TIMESTAMP_URL to your CA's timestamp server URL. An example for Verisign would be "http://timestamp.verisign.com/scripts/timstamp.dll".

If you wish to rebuild the installer that we provide, add the Inno Setup directory to the PATH environment variable. In a default installation that would be "C:\Program Files\Inno Setup 5", or "C:\Program Files (x86)\Inno Setup 5" on a 64-bit OS.

4. Driver Signing

You will need to sign the driver's files with your code-signing certificate.

Open a Windows Command Prompt (note that on Vista you will need to right click on the Command Prompt icon and select "Run as administrator" as Signtool.exe requires Administrator rights) and change to the top-level directory of your locally checked-out Subversion release tag.

Sign the 32-bit driver components:

```
%WLHBASE%\bin\SelfSign\signtool.exe sign /v /ac
"%SELFSIGN_CROSS_CERT%" /n "%SELFSIGN_CERT_NAME%" /t
"%SELFSIGN_TIMESTAMP_URL%" binary\drv\win32\release\*.sys
```

Recreate the 32-bit driver catalog file:

```
%WLHBASE%\bin\SelfSign\inf2cat.exe /v
/drv:binary\drv\win32\release /os:XP_X86,Vista_X86
```

Sign the 32-bit driver catalog file:

```
%WLHBASE%\bin\SelfSign\signtool.exe sign /v /ac
"%SELFSIGN_CROSS_CERT%" /n "%SELFSIGN_CERT_NAME%" /t
"%SELFSIGN_TIMESTAMP_URL%" binary\drv\win32\release\*.cat
```

Sign the 64-bit driver components:

```
%WLHBASE%\bin\SelfSign\signtool.exe sign /v /ac
"%SELFSIGN_CROSS_CERT%" /n "%SELFSIGN_CERT_NAME%" /t
"%SELFSIGN_TIMESTAMP_URL%" binary\drv\x64\release\*.sys
```

Recreate the 64-bit driver catalog file:

```
%WLHBASE%\bin\SelfSign\inf2cat.exe /v
/drv:binary\drv\x64\release /os:XP_X64,Vista_X64
```

Sign the 64-bit driver catalog file:

```
%WLHBASE%\bin\SelfSign\signtool.exe sign /v /ac
"%SELFSIGN_CROSS_CERT%" /n "%SELFSIGN_CERT_NAME%" /t
"%SELFSIGN_TIMESTAMP_URL%" binary\drv\x64\release\*.cat
```

5. Installer Creation and Signing

After signing the driver's files you will need to rebuild the installer application to include the newly signed driver files, and then sign the installer application with your code-signing certificate.

Open a Windows Command Prompt (note that on Vista you will need to right click on the Command Prompt icon and select "Run as administrator" as Signtool.exe requires Administrator rights) and change to the sub-directory host/install/win in your locally checked-out Subversion release tag.

Rebuild the installer application using Inno Setup:

```
iscc.exe custom.iss
```

Sign the installer application:

```
%WLHBASE%\bin\SelfSign\signtool.exe sign /v /n
"%SELFSIGN_CERT_NAME%" /t "%SELFSIGN_TIMESTAMP_URL%"
..\..\..\binary\*Installer.exe
```

Note: For convenience, there is a build script "build.cmd" in the host/install/win directory that automates the steps above for rebuilding and signing the installer application. Note that it requires that Python is installed. http://www.python.org