EV Kernel-Mode Driver Signing

Microsoft Documentation: https://msdn.microsoft.com/en-us/library/windows/hardware/hh967734%28v=vs.85%29.aspx
Video Tutorial of the signing process: 1-31-2013 Kernel-Mode Driver Signing (this did not use a token but the process should still be the same)
Basic Token Set up Steps: EV Code Signing Token Preparation

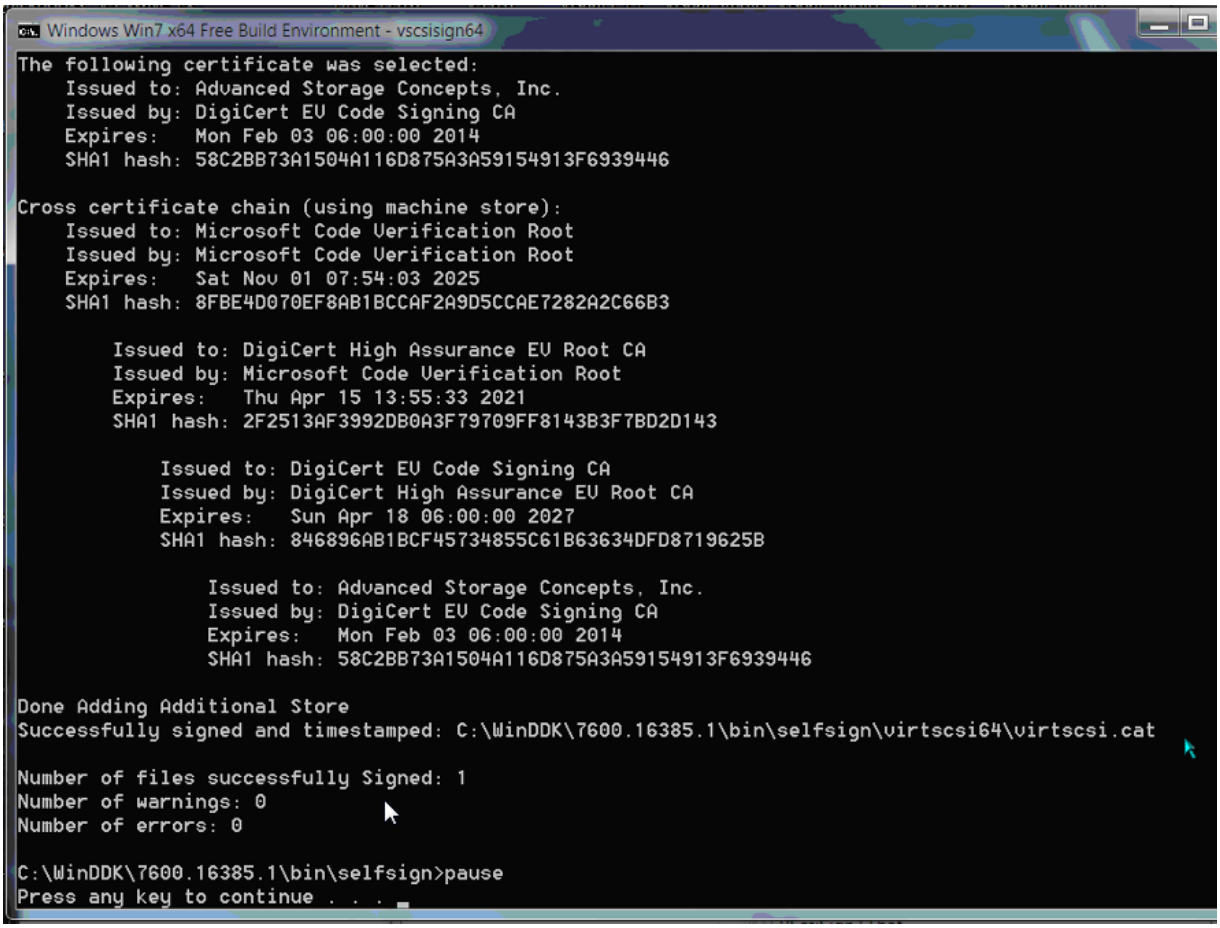
Signing Process:

1. They need to create a .cat file. This requires both a .sys and a .inf file. (They should have both) To do so, they will need to run the **Inf2cat** command listed in the instructions above. For details, see Taylor's video above. Note: Even if you have admin privileges, sometimes you have to right click and choose Run as Administrator anyway.

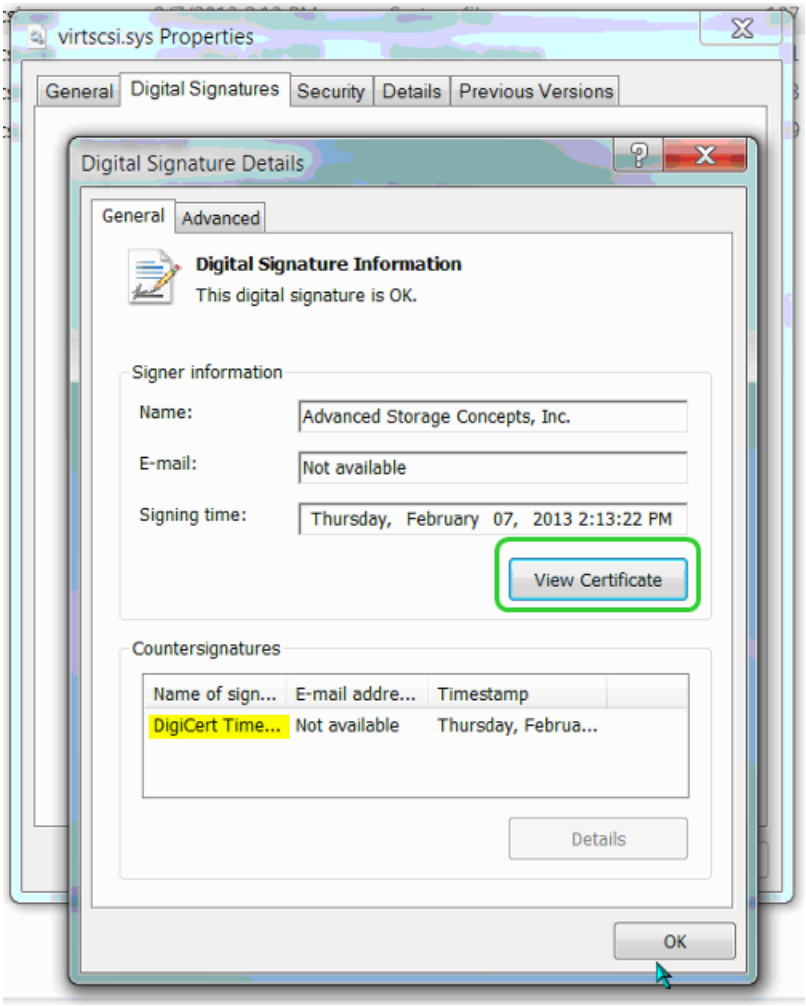
2. Sign the sys files (if needed). Then, create the .cat file and sign the .cat file. To do so, they need to run something like this from their command line:
    a. SHA1 signing option:
        i. signtool sign /v /ac "C:\test\DigiCert High Assurance EV Root CA.crt" /s my /n "Company Name, Inc" /t http://timestamp.digicert.com DRIVER.SYS
    b. SHA2 signing option:
        i. signtool sign /v /ac "DigiCert High Assurance EV Root CA.crt" /tr http://timestamp.digicert.com /td sha256 /fd sha256 /s my /n "Company Name, Inc" DRIVER.SYS
    c. NOTE:
    "Company Name, Inc." is the Common Name of the code signing certificate
    "DigiCert High Assurance EV Root CA.crt" is the cross-signed certificate, which is available for download here
    DRIVER.SYS is the driver file you want to sign
    For all of the command-line options for using SignTool, please see Microsoft's Signtool Documentation Page.
    When using the SHA2 timestamp or /fd sha256 please make sure the latest version of signtool (6.3 or newer) and safenet are used.

    d. **IMPORTANT NOTE**: The order in which they sign can be very important. If they are signing the .sys files, they must do so **before** they generate and sign the .cat.

    The Globalsign Documentation may be insightful here: https://support.globalsign.com/customer/portal/articles/1491089-kernel-mode-driver-signing-%E2%80%93-windows-7-8



3. Note that in the command above, we are specifically referencing the cross root cert signed by Microsoft. It's still called Digicert High Assurance EV Root CA, but its signed by their root. If you don't see that, then they're signing with the wrong root.
   They can grab the correct one here: https://www.digicert.com/code-signing/driver-signing-in-windows-using-signtool.htm#download_cross_certificate
4. To test it, they can go to the properties of the cat file. Under 'Digital Signatures' it should show the new signature:



5. Verify that it signed correctly by running the signtool verify command (it should show the Microsoft root in the Cross-Certificate section): signtool.exe verify /kp /v /c catalog.cat driver.sys
6. You're done!

Known Issues

- The customer gets a '**smartcard**' error, or something about '**the drivers could not be found**'. This typically indicates that the SafeNet drivers have not been installed, so it doesn't recognize the token.
- The signature no longer shows valid. This can happen if they modified the certificate after it was signed. (I know, you would think this would be obvious, but checking this one thing could have saved us a 3 hour phone call.)
- The Safenet Authentication client says, '**This is an unlicensed copy for evaluation use only**'. The client, although a trial, is fully functional and has no expiration date. If the customer throws a fuss about it, we do have a key they can use to authenticate it. This is it:

MQAzADMANQA0ADYANAAzADEANAA7ADIAOQA5ADQAMwA7AEQAaQBnAGkAYwBlAHIAdAAgAEkAbgBjAC4AOwAxADAAMQA7ADAAOwA1AEIAUAB3AHUAeABoAEgAYQBwAGMAQgBpAFoAbABkAC8ARwBMADUAKwBtADMAYQBnAFEAdABkAGkAawAxAHAASwAyAHAARwB4AGQAbgBiADMALwB2ADkAcgB1AFEARwBEAFQAQgBWAHoAdQBhAEoAdwBJAFQATQBYAGMAQQBBAE4AWABxAFQAQgBPADcAVQBJAGEARwBRAFoAVQByAFUAaQBVADgAVgBvAEEA

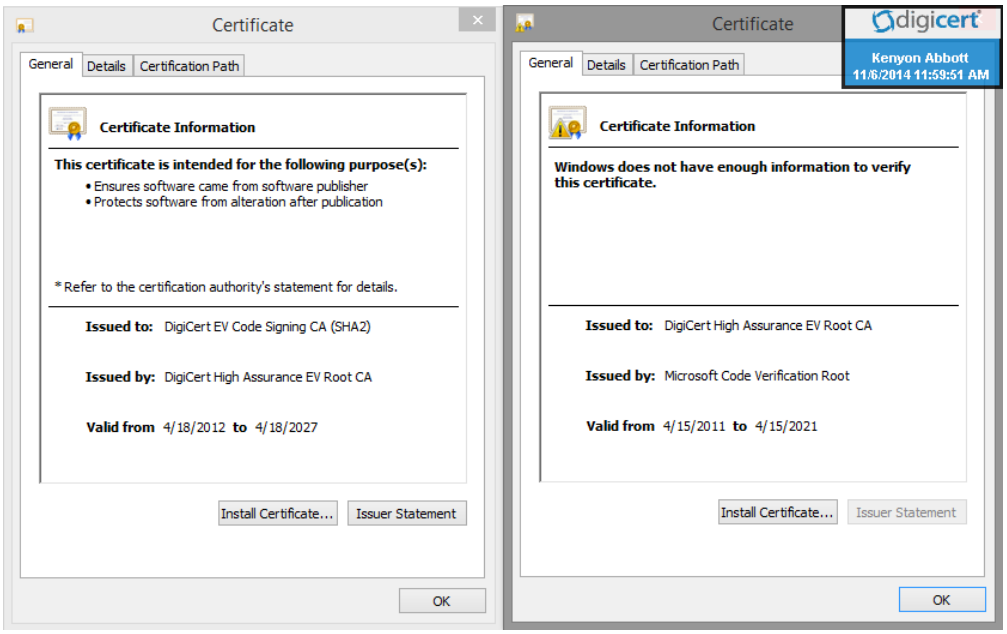
Note: I have put it on two lines so you can see the whole string, but it's one long string, so you may have to backspace once.

Other Notes

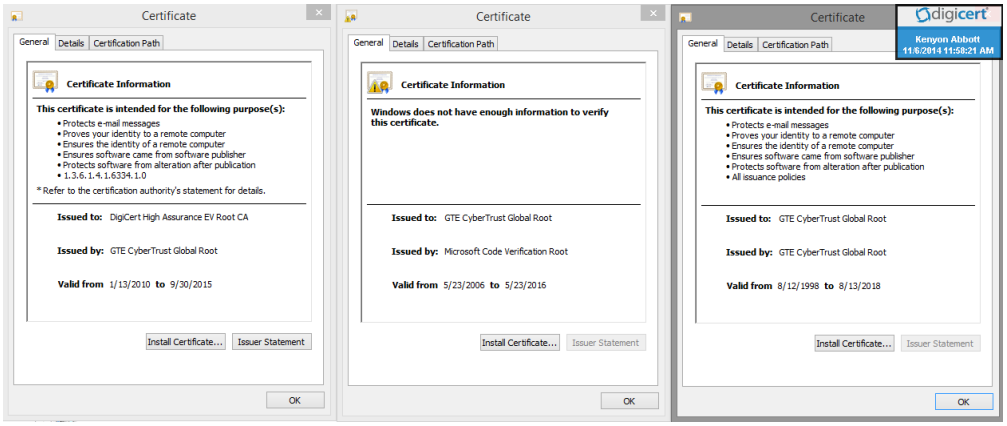
We can cross chain code signing certificates as well, additionally we can cross-chain to Baltimore or GTE CyberTrust, then use their Microsoft verification root.

Please see here for the three chains: kernel-chains.zip

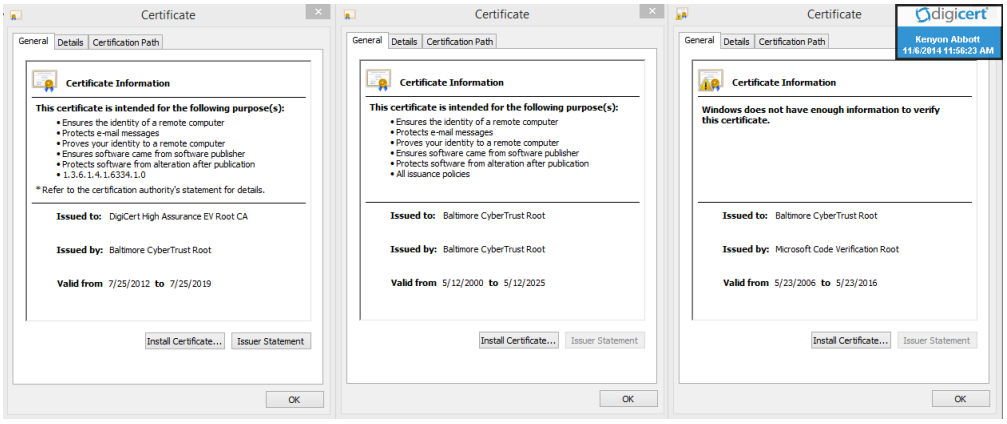
DigiCert



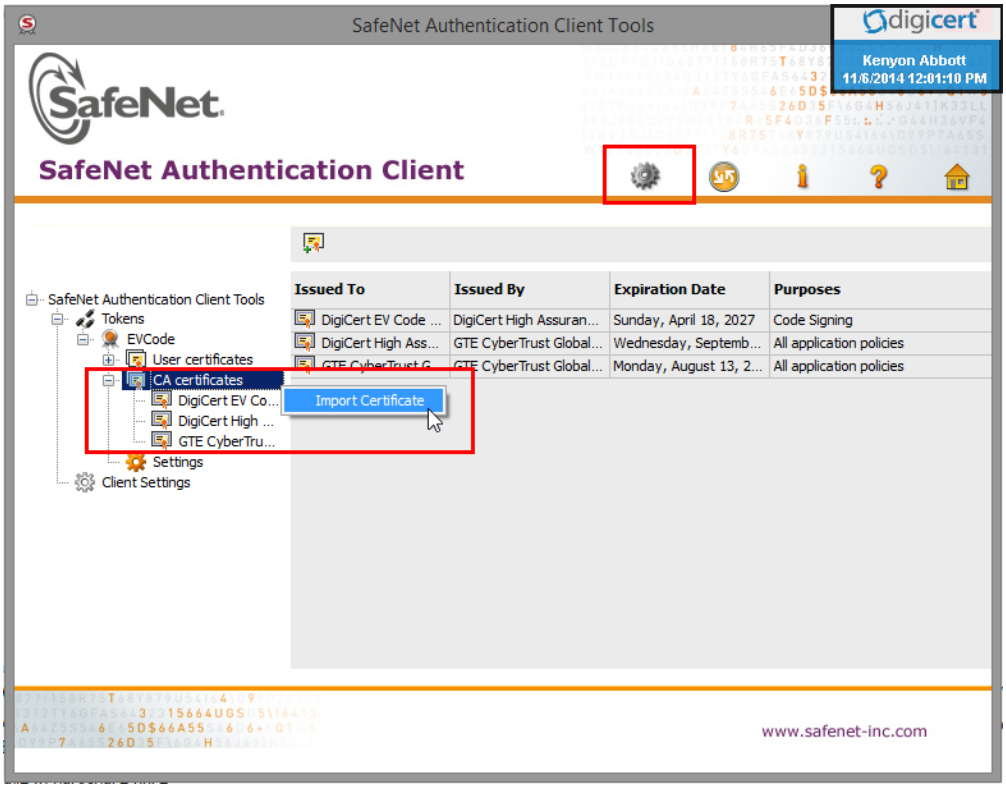
GTE



Baltimore



To use them on the token you can import them manually in the SAC Tools



- Customer tries signing on Windows 7 and gets:

Signtool Error: An Unexpected Internal Error
Error information: "CryptQueryObject" (-2147024773/0x8007007b)
Something is either wrong with their syntax, or the filename itself. This customer renamed the file to rick.crt and then it worked fine.