Signature de pilotes de périphériques en 2016

Par Martin Dubois, ing.

mdubois@kms-quebec.com

Présentation

L'arrivée de Windows 10 et la désuétude de SHA-1 apportent bien des changements à la signature numérique des pilotes de périphériques. Et malheureusement, la documentation de Microsoft est parfois un peu lourde et ardue à déchiffrer. J'ai donc décidé de réaliser un document regroupant les aspects importants.

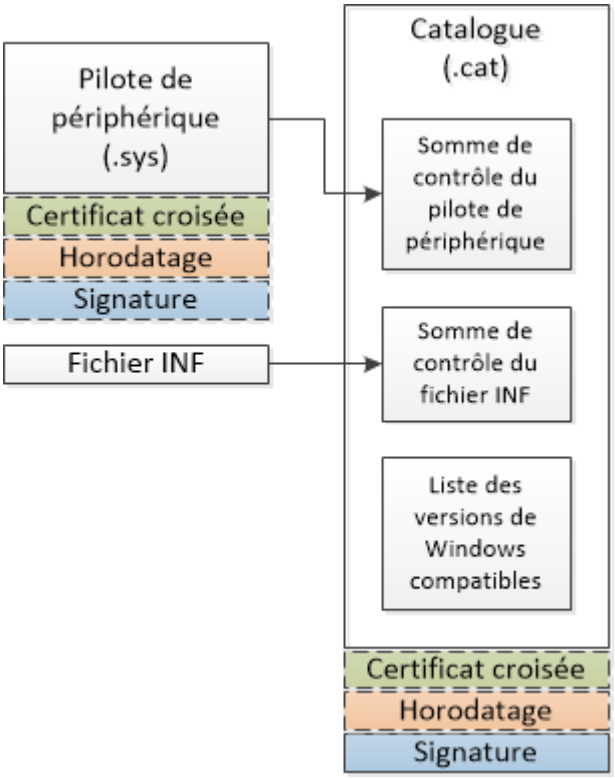
Table des matières

1 Introduction

Signer un pilote de périphérique pour Windows est presque aussi complexe qu'en programmer un. J'exagère un peu, mais il n'en reste pas moins que c'est le sujet qui revient le plus souvent dans mes discussions au sujet des pilotes de périphérique pour Windows. Ce document tente donc, comme bien d'autres avant lui, de clarifier et simplifier un peu les choses.

2 Comment un pilote est-il signé

Ce n'est pas le pilote lui-même qui est signé. C'est plutôt le fichier de catalogue. Celui-ci contient un « hash » du pilote, un « hash » tu fichier INF et un « hash » de tout autre fichier nécessaire pour l'installation telle que des co-installateurs, des installateurs de classe ou d'autres types de DLL.



Dans de rares cas, il faut aussi signer le pilote lui-même. Les pilotes qui doivent être chargés avant que le système de fichier ne soit disponible pour permettre de retrouver le fichier catalogue doivent aussi être signés de la même manière que le fichier catalogue. C'est le cas pour les pilotes des ports SATA ou des disques durs par exemple. La signature du pilote ne cause aucun problème, en cas de doute, il est donc préférable de signer, et le fichier catalogue, et le pilote.

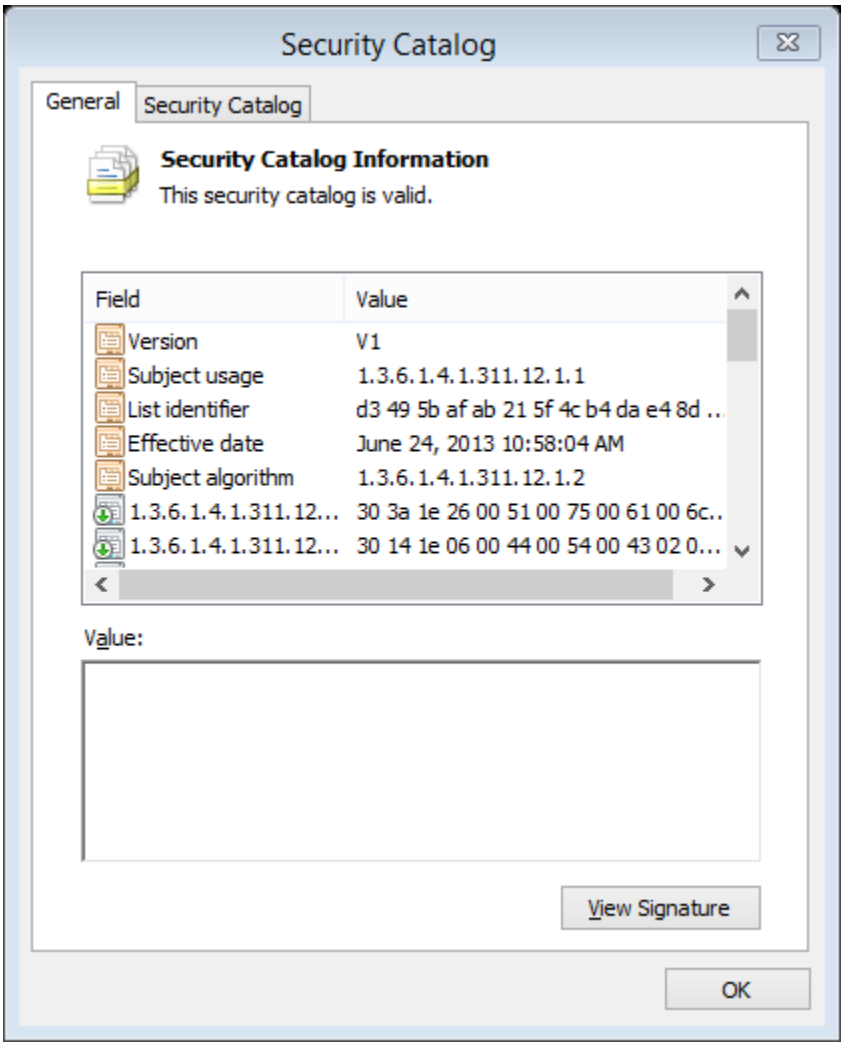
Si vous êtes un lecteur attentif, vous avez certainement remarqué la boite « Certificat croisé » dans le diagramme. Cette boite correspond au lien entre l'autorité qui a émis le certificat utilisé pour la signature et le certificat utilisé par Microsoft pour vérifier les signatures. Naturellement, si c'est Microsoft qui a signé le pilote, cette partie n'est pas présente. Elle est aussi absente si c'est une signature de test.

Il y a aussi un autre item nommé « Horodatage ». Cette information certifie la date de signature du pilote. Elle est créée par un serveur de temps. Celle-ci n'est pas présente dans une signature de test.

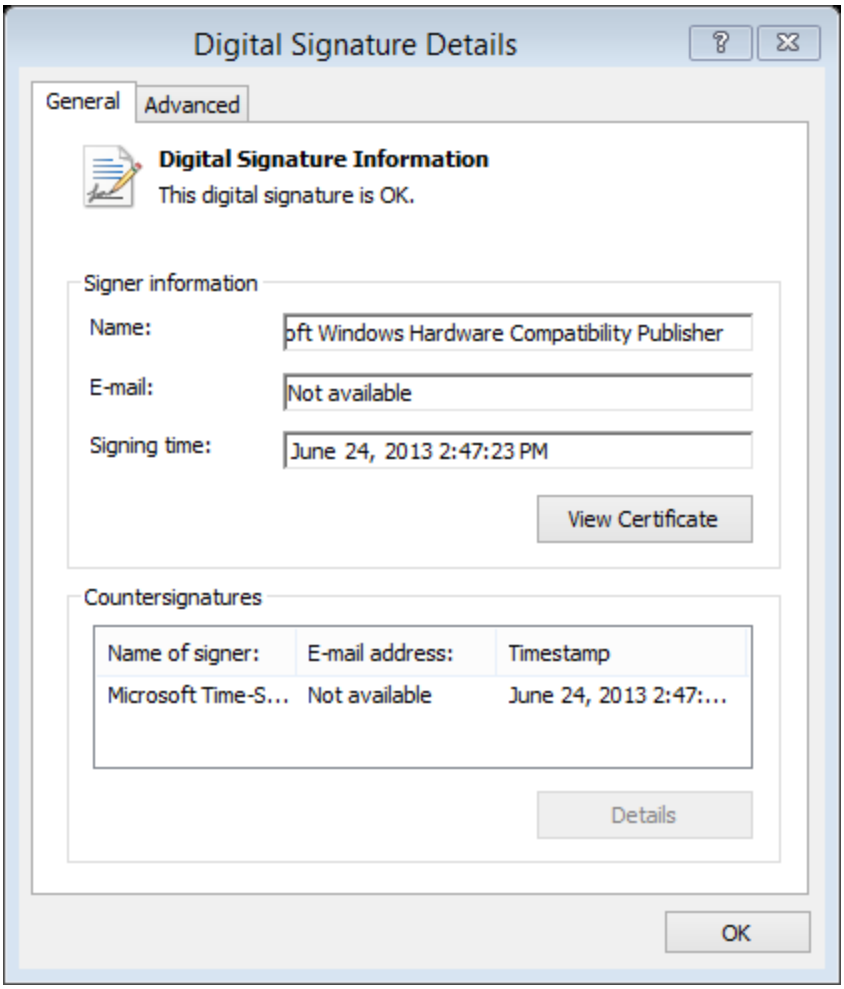
2.1 Comment visualiser la signature d'un pilote

Il y a deux moyens pour visualiser la signature d'un pilote.

Le premier est simplement de double cliquer sur le catalogue (fichier « .cat »).



Ensuite, le bouton « View Signature » permet de visualiser la signature.



Dans la figure, il est possible de voir que le pilote a été signé par Microsoft le 24 juin 2013 et que la date de signature est garantie par le serveur de temps de Microsoft.

Le principal défaut de cette première méthode est qu'elle ne permet pas de voir si un certificat croisé est présent ou non.

La seconde méthode repose sur un outil fourni par Microsoft avec le WDK. L'exécutable « SignTool.exe » permet de signer un fichier ou de vérifier sa signature. Dans le cas d'un pilote de périphérique, il faut utiliser la commande « `SignTool.exe Verify /kp` *`Fichier`*`.cat` » pour vérifier et afficher la signature en incluant le certificat croisé s'il est présent. Le document [2] décrit complètement l'outil « SignTool.exe ».

3 Vérification à l'installation et au chargement

Une des choses importantes à savoir sur la signature d'un pilote de périphérique est que celle-ci est vérifiée à deux moments distincts par Windows.

Elle est tout d'abord vérifiée lors de l'installation du pilote de périphérique. Si cette première vérification échoue, un utilisateur avec les privilèges administrateur peut choisir de quand même installer le pilote. Si la vérification réussit, mais que l'auteur du pilote n'est pas connue, un message demande à l'utilisateur de confirmer l'installation. Pour confirmer, l'utilisateur doit disposer des privilèges administrateur. Dans les deux cas, une installation complètement automatique du pilote est donc impossible. À ce niveau, c'est toujours la signature du fichier catalogue qui est vérifié.

La seconde vérification de la signature est faite par le noyau de Windows lui-même au chargement du pilote. Je devrais plutôt dire : à chaque chargement du pilote. Si cette vérification échoue, le pilote n'est tout simplement pas chargé. Consultez le tableau qui suit pour savoir quelles versions de Windows effectuent cette vérification de signature au chargement d'un pilote. À ce niveau, c'est généralement la signature du fichier catalogue qui est vérifié, mais dans le cas de pilotes chargé très tôt durant la séquence de démarrage, il est possible que ce soit la signature du pilote qui soit vérifiée.

	x86	X64
Windows XP	Non	
Windows Vista	Non	Oui
Windows 7		
Windows 8	Voir la note	
Windows 8.1		
Windows 10		

Note : Si le « Secure Boot » est activé, ces versions de Windows vérifient les signatures des pilotes au chargement

3.1 Désactiver la vérification des signatures au chargement

Il est possible de désactiver la vérification de la signature des pilotes lors de leur chargement. Il y a même deux manières de le faire.

La première est très simple, il suffit de connecter un débogueur noyau (Visual Studio avec le WDK installé, ou WinDbg) à l'ordinateur. La désactivation reste effective tant que le débogueur demeure connecté.

La seconde consiste à appuyer sur la touche « F8 » lors du démarrage de Windows pour entrer dans le menu des options de démarrage avancé et ensuite sélectionner l'option « Disable Driver Signature Enforcement ». La désactivation demeure effective jusqu'au prochain démarrage de l'ordinateur.

La section « How to Disable Signature Enforcement on a Test Computer », débutant à la cinquante-troisième page du document [1], décrit en détail les deux méthodes.

Note : Malgré qu'il soit possible de désactiver la vérification de la signature, je crois qu'il est préférable, lors du développement d'un pilote, d'utiliser une signature de test. Une telle signature sera décrite dans la prochaine section. Elle permet de charger le pilote sur un ordinateur de test sans que celui-ci soit connecté au dévermineur et sans modifier les options de démarrage avancés.

4 Types de signature possibles

4.1 Signature de test avec un certificat autosigné

C'est le type de signature à utiliser lors du développement de pilote. Pour signer un pilote de cette manière, il faut aller dans les propriétés du projet « Visual Studio » pour le « package » du pilote. Il faut ensuite activer la signature de test. Le WDK s'occupe alors de générer un certificat de test et de signer le fichier catalogue avec celui-ci. Si le pilote doit aussi être signé, c'est aussi simple. Il faut simplement modifier les propriétés au niveau du projet du pilote.

Pour utiliser un pilote signé de cette manière sur un ordinateur de test, il faut

- Installer le certificat de test sur l'ordinateur de test ;
- Activer l'acceptation des signatures de test sur l'ordinateur de test.

La section « Utilisation d'un pilote signé avec un certificat autosigné » explique ces deux étapes.

Note : Il ne faut pas utiliser une signature de test pour un pilote utilisé en « production ». Il serait dangereux de demander aux utilisateurs d'activer l'acceptation des signatures de test. D'ailleurs, quand la signature de test est activée sur un ordinateur, un message est inscrit dans les coins du fond d'écran pour l'indiquer.

Versions de Windows supportées	Tous

4.2 Signature de test Microsoft

Le « Dashboard » du « Hardware Dev Center » de Microsoft offre la possibilité de signer un pilote dans le but d'effectuer des tests. Je ne peux cependant pas en dire plus sur cette fonctionnalité, car la signature en utilisant un certificat autosigné a toujours très bien fonctionné et je n'ai jamais senti le besoin de, même, essayer cette option.

4.3 Signature propriétaire

Il est aussi possible pour l'auteur d'un pilote de périphérique de se procurer un certificat auprès d'une autorité et de l'utiliser pour signer un pilote. Microsoft donne une liste des fournisseurs de certificat avec lesquels il est possible de faire affaire et pour lesquels des certificats croisés sont disponibles (fournis avec le WDK à partir de la version 8).

Une fois le certificat obtenu, il suffit de l'installer sur l'ordinateur de développement et d'ensuite configurer le projet du « package » du pilote (et le projet du pilote au besoin) pour effectuer une signature réelle. Dans ce cas, il faut sélectionner le certificat à utiliser et spécifier le serveur de temps à utiliser.

Si un serveur de temps n'est pas spécifié, Visual Studio signera le fichier sans ajouter d'horodatage. Cela aura de graves conséquences sur la validité de la signature. La signature ne sera alors considérée valide que jusqu'à l'expiration du certificat.

Maintenant que la signature n'est plus manuelle et qu'elle est intégrée à l'environnement de Visual Studio, il n'est plus nécessaire de se préoccuper du certificat croisé. L'outil de signature s'en charge pour nous.

Versions de Windows supportées	Windows Vista, Windows 7, Windows 8 et Windows 8.1 Windows 10, si la signature comprend un horodatage et que la date de signature précède la date de sortie de Windows 10 (25 juillet 2015).
Avantages	• Pas de requis spécifique pour les tests du pilote ; • Pas d'obligation d'envoyer le pilote à Microsoft • Pas de frais pour la signature (il faut acheter un certificat, mais celui-ci est aussi nécessaire pour soumettre une demande signature ou de certification à Microsoft)
Désavantages	• Lors de l'installation, Windows affiche un message « inquiétant » mettant en doute la provenance du pilote ; • Impossible d'installer le pilote de manière entièrement automatique sans intervention d'un utilisateur disposant des privilèges administrateurs. • Le certificat utilisé pour la signature à une date d'expiration. Il devient donc nécessaire de périodiquement produire un autre ensemble signé avec un nouveau certificat.

4.4 Signature par Microsoft

Avec l'arrivée de Windows 10, Microsoft oblige tous les nouveaux pilotes à être signés par Microsoft. Les nouveaux pilotes signés directement par l'auteur ne fonctionnent pas sous Windows 10.

Versions de Windows supportées	Windows 10
Avantages	• Pour un nouveau pilote utilisé sous Windows 10, c'est une des deux solutions obligatoires ; • Pas de frais exigé par Microsoft ; • Pas de requis spécifique pour les tests du pilote.
Désavantage	• Obligation de faire parvenir le pilote à Microsoft pour la signature.

Le document [6] décrit la procédure pour ce que Microsoft appelle « Attestation Signing ».

4.5 Certification et signature par Microsoft

C'est la solution complète ! L'auteur du pilote doit exécuter les tests de certification définis par Microsoft et associé au type de pilote.

Microsoft fournit le HCK (Windows 8.1 et prédécesseurs) et le HLK (Windows 10) qui comprend l'ensemble des tests et un environnement d'exécution. Pour l'utiliser, il faut installer un serveur responsable de maintenir la base de données des tests et résultats et aussi responsable de gérer et surveiller l'ensemble des ordinateurs de tests.

Croyez-moi sur parole, l'installation et la maintenance du HCK représentent à eux seuls un travail important.

L'auteur est responsable de l'exécution des tests sur toutes les versions de Windows pour lesquels il demande une certification. Ensuite, il doit faire parvenir les résultats des tests ainsi que le pilote à Microsoft qui vérifie les résultats des tests et signe le pilote. Ici aussi, l'exécution des tests représente un travail important.

Il y a deux types de certification :

1. Signature seulement : Si le pilote ne correspond à aucune catégorie de périphérique supportée par le HCK ou le HLK, l'auteur exécute avec succès des tests de base montrant que le pilote ne compromet pas la stabilité du système et Microsoft signe le pilote sans pour autant permettre que celui-ci soit vendu comme « Conçu pour Windows ».
2. Programme de logo : Si le pilote correspond à une des catégories de périphérique supportées par le HCK ou le HLK, il faut alors que l'auteur exécute avec succès l'ensemble des tests associé à cette catégorie. Microsoft signe alors le pilote et permet que le produit soit vendu avec l'indication « Conçu pour Windows »

Versions de Windows supportées	Tous
Avantages	• Augmente la confiance du client dans le produit ; • Permets une installation complètement automatisée du pilote ; • Le certificat signant le pilote n'expire jamais.
Désavantages	• Il faut payer Microsoft pour chacune des certifications • Il est nécessaire d'exécuter l'ensemble des tests de certification chaque fois que le pilote change, même si le changement est mineur

5 Utiliser un pilote signé avec un certificat autosigné

5.1.1 Installation du certificat de test sur l'ordinateur de test

Si c'est le catalogue qui est signé, il est possible d'installer le certificat de test simplement en cochant la case indiquant de faire confiance à l'auteur du pilote quand Windows vous avertit de la signature non valide du pilote lors de son installation.

Si la signature du pilote est nécessaire, il faut copier sur l'ordinateur de test le fichier « .cer » que Visual Studio génère lors de la première signature du pilote (il est placé dans le répertoire du projet du pilote).

Ensuite, il suffit de double cliquer sur ce fichier et de suivre l'assistant d'installation de certificat en utilisant les réponses par défaut.

La section « Preparing the Test System » du document [1] explique comment installer le certificat sur l'ordinateur de test. Les explications restent valides, mais datent d'avant l'intégration du WDK à Visual Studio, ce qui les rend un peu plus lourdes.

5.1.2 Activation de l'acceptation des signatures de test

Il faut ouvrir une fenêtre de commande s'exécutant en tant qu'administrateur et exécuter la commande « `BCDEdit.exe /set TESTSIGNING ON` ».

La section « Preparing the Test System » du document [1] explique aussi comment active l'acceptation des signatures de test. Le document [3] décrit la commande « set » de l'outil « BCDEdit »

6 Obtention d'un certificat

Depuis le début de 2016, il n'est plus possible de se procurer un certificat utilisant l'algorithme SHA-1. Il ne reste donc que deux types de certificats qu'il est possible de se procurer.

2	**Code Signing Certificate (SHA-2)** Vous ne pouvez plus utiliser ce type de certificat pour interagir avec le WHQL. Il est cependant toujours possible de l'utiliser pour signer des pilotes qui fonctionneront avec les versions de Windows précédant la version 10.
3	**Extended Validation (EV) Code Signing Certificate** C'est le nouveau type de certificat à utiliser. Il permet l'ensemble d'interagir avec le WHQL et de signer de nouveaux pilotes pour les versions de Windows précédant la version 10. Il permet aussi bien d'autres opérations.

6.1 Fournisseurs

	Code Signing Certificate			EV Code Signing Certificate		
Fournisseur	**1 an**	**2 ans**	**3 ans**	**1 an**	**2 ans**	**3 ans**
Symantec www.symantec.com	499 $ US	873 $ US (436 $/an)	1248 $ US (416 $/an)	795 $ US	1249 $ US (624 $/an)	1549 $ US (775 $/an)
Global Sign www.globalsign.com	219 $ US	395 $ US (197 $/an)	525 $ US (175 $/an)	410 $ US	760 $ US (380 $/an)	950 $ US (316 $/an)
Digicert www.digicert.com	111 $ US	198 $ US (99 $/an)	267 $ US (89 $/an)	224 $ US	400 $ US (200 $/an)	498 $ US (166 $/an)
WoSign buy.wosign.com	Le site n'est pas disponible en anglais ou en français.					
Entrust	Le prix n'est pas directement accessible.					

Note : Prix en date du 12 janvier 2015.

7 Abréviations

DLL **D**ynamic **L**ink **L**ibrary

EV **E**xtended **V**alidation

HCK **H**ardware **C**ertification **K**it

HLK **H**ardware **L**ab **K**it

SATA **S**erial **A**dvanced **T**echnology **A**ttachment

SHA **S**ecure **H**ash **A**lgorithm

WDK **W**indows **D**river **K**it

WHQL **W**indows **H**ardware **Q**uality **L**ab

8 Références

1. Kernel-Mode Code Signing Walkthrough
   http://www.microsoft.com/whdc/winlogo/drvsign/kmcs_walkthrough.mspx
2. SignTool.exe (Sign Tool)
   https://msdn.microsoft.com/en-us/library/8s9b9yaz%28v=vs.110%29.aspx?f=255&MSPPError=-2147217396
3. BCDEdit /set
   https://msdn.microsoft.com/en-us/library/windows/hardware/ff542202(v=vs.85).aspx
4. Installing an Unsigned Driver during Development and Test
   https://msdn.microsoft.com/en-us/library/windows/hardware/ff547565(v=vs.85).aspx
5. Get a code signing certificate
   https://msdn.microsoft.com/library/windows/hardware/hh801887.aspx
6. Attestation Signing a Kernel Driver For Public Release
   https://msdn.microsoft.com/en-us/windows-drivers/develop/attestation_signing_a_kernel_driver_for_public_release?f=255&MSPPError=-2147217396