Windows 8 Sideloading

Table des matières

[1 Le concept 2](#_Toc399770678)

[1.1 Le Sideloading – Théorie en vrac (collecte d’infos) 2](#_Toc399770679)

[1.2 Configurations requises 3](#_Toc399770680)

[1.2.1 Avec un domaine 3](#_Toc399770681)

[1.2.2 Sans utiliser de domaine 3](#_Toc399770682)

[1.3 Préparer un environnement de test 4](#_Toc399770683)

[1.3.1 Besoin matériel 4](#_Toc399770684)

[2 Les étapes techniques pour le déploiement Sideloading 5](#_Toc399770685)

[2.1 Résumé des étapes 5](#_Toc399770686)

[2.2 Signer le package avec un certificat valide 5](#_Toc399770687)

[2.2.1 Créer un certificat valide 5](#_Toc399770688)

[2.2.2 Convertir pvk et cer en pfx : 5](#_Toc399770689)

[2.2.3 Installer le certificat 6](#_Toc399770690)

[2.3 Créer le package d’application 8](#_Toc399770691)

[2.3.1 Configurer le manifest 8](#_Toc399770692)

[2.3.2 Créer l’appx via Visual Studio 2013 8](#_Toc399770693)

[2.3.3 Créer l’appx en ligne de commande 8](#_Toc399770694)

[2.4 Préparer le poste cible 9](#_Toc399770695)

[2.4.1 Installer le certificat public (cer) 9](#_Toc399770696)

[2.4.2 Installer la clé de sideloading 10](#_Toc399770697)

[2.4.3 Autoriser les applications approuvées 10](#_Toc399770698)

[2.4.4 Déployer la solution 10](#_Toc399770699)

[2.4.5 Créer un BAT pour le poste client 10](#_Toc399770700)

[2.4.6 Créer un exécutable 11](#_Toc399770701)

[3 Questions: 12](#_Toc399770702)

[4 Liens utiles 12](#_Toc399770703)

# Le concept

## Le Sideloading – Théorie en vrac (collecte d’infos)

L’idée centrale qui régit le monde de l’application Windows est le principe de la « **confiance totale** ». L’idée est de faire en sorte qu’il soit aussi difficile que possible d’installer une application sans aucune forme de contrôle. (Fini les .exe qui font tout et n’importe quoi !)

Attention, le processus est différent entre Windows 8 et Windows RT.

Par défaut, quand une application est compilée, elle est signée avec une clé temporaire spécifique aux tests sur la machine locale. Elle est obtenue par l’activation de la licence de développeur Microsoft (gratuite sur inscription)

Toute machine qui comporte une licence développeur est « grande ouverte ».

Pour installer une application sur un poste client, Vous devez avoir un certificat que le poste client qui corresponde à celui utilisé pour signer l’application lorsqu’elle a été générée. (Dans le cas de test local, il s’agit d’un certificat de test généré par Visual studio)

Selon ce que je comprends, un certificat installé sur le poste client n’est que la partie publique (Vérification seulement) du certificat total comprenant clé privée (Signature de l’application lors de son développement) et clé publique (vérification de l’application lors de son installation (et de son exécution ?)). Du coup, dans le cas du build de test dans Visual studio, l’application est signée par une clé privée de VS et vérifié par un certificat publique installé en même temps que VS. Donc pour installer notre application sur les PC cibles, nous devons créer notre propre certificat (paire de clé), signer le package d’application avec notre clé privée et diffuser la partie publique pour qu’elle soit installée sur tous les PCs cibles.

**A propos du système de licences Microsoft :**

You should know that I'm not an expert on Microsoft licensing, and I don't give advice on it to anyone, especially people I don't know. Getting Microsoft licensing right is intensely difficult and you should always seek specialised advice. But I do need to say something…

There are, broadly, two ways two buy Microsoft licenses. You can either buy them at retail (e.g. a boxed product form a shop, or an OEM version installed by a system vendor), or you can buy them on a "volume license" (VL) program from Microsoft. The idea of VL is that it makes it easier to license your whole organisation. You also get some benefits in terms of extra rights and tools if you go down the VL route.

## Configurations requises

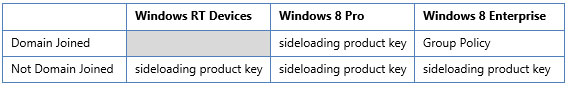
**2 possibilités**, avec un domaine et sans domaine. Le domaine remplace la clé de sideloading en activant le « **Allow all trusted apps to install** » dans les options de sécurités. Cela permet à tous les PCs dans le domaine (Et exécutant Win 8 Enterprise) d’installer des applications **A LA CONDITION SUIVANTE** : elles ont été signées par un certificat qui dépend d’un certificat racine approuvé.

### Avec un domaine

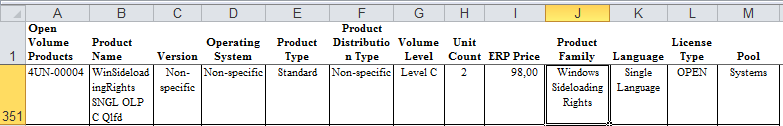
* Windows **Server 2012** et **8 Enterprise** sont classés « enterprise sideloading enabled ». Ce qui signifie qu’ils sont prêts à recevoir des applications déployés sans passer par le Windows store, cependant, les machines doivent avoir rejoint un domaine.
* Le **réseau local** doit disposer d’un « **Domain Controller** » (Un serveur exécutant Active Directory Domain Services)
* La **machine cible** doit être rejoindre le domaine et tourner sous Server 2012 ou Windows 8 Enterprise

### Sans utiliser de domaine

* Utilisation d’une clé de sideloading même avec Windows 8 Enterprise



**Important ! Changement dans les licences depuis mai 2014**



Depuis Mai 2014, Microsoft a changé son fonctionnement au niveau des licences de Sideloading.

Les postes ayant reçu la mise à jour « **Windows 8.1 Update** » bénéficient du nouveau système de Licence pour le Sideloading. Le tableau des prérequis ci-dessus à un peu évolué du fait que l’activation du sideloading via un domaine a été étendu à **Windows 8.1 Pro**.

Au niveau des licences, il existe un nouveau type de licence permettant d’activer le sideloading sur les postes n’étant pas dans un réseau.

Les informations sur les prérequis selon les OS ont légèrement changés comme le montre le tableau ci-dessous :

|  |  |  |
| --- | --- | --- |
| **Système** | **Joint à un domaine** | **Workgroup** |
| **Windows 8.1 Update Entreprise/Pro** | Clé de sideloading débloqué | Clé de sideloading nécessaire :  Incluse dans les licences, EA, ESA,EES,  Select et Select Plus, SE  Ou achat d’une clé à 100 $ pour un nombre illimité d’appareils via le programme Open License |
| **Windows RT 8.1 Update** | N/A |
| **Windows 8.1 Update  (Core)** | N/A | Non |

Comme indiqué dans le tableau, cette nouvelle possibilité permettrait d’avoir une clé unique qui permettrait d’activer un nombre illimité de postes clients pour 100$.

Le programme OpenLicence est très mal documenté sur le net. Il semblerait qu’il soir accessible après l’achat de 5 licences de produits Microsoft auprès d’un revendeur. Il est possible qu’Elipce l’ait déjà suite à nos achats de licences Windows 7. Il faut se renseigner auprès de notre revendeur pour connaitre les modalités d’accès à ce service je pense.

Mais que sont ces licences EA, ESA, EES, Seletc et Select plus ?

* **EA** : signifie Enterprise Agreement, ces licences sont destinées aux entreprises de plus de 250 postes.
* **ESA** : Enterprise Subscription Agreement, peu de docs là-dessus, un peu comme EA mais en mode abonnement apparemment.
* **EES** : Enrollment Education Solutions, réservées aux structures d’éducation.
* **Select**: Select License est un programme d'acquisition de licences en volume Microsoft de type transactionnel qui s'avère souple et abordable ; il s'adresse aux grandes organisations, publiques ou privées, disposant d'au minimum 250 ordinateurs de bureau et de besoins logiciels mixtes et qui souhaitent se procurer les dernières technologies Microsoft sur la base d'un paiement à l'utilisation.

## Préparer un environnement de test

### Besoin matériel

* **Serveur de domaine** : Windows 8 Enterprise avec Active Directory Domain Services (AD DS)
* **PC cible :** Windows 8 Enterprise OU pro avec clé de sideloading
* **PC de développement** : Windows 8 avec Visual studio (Nécessairement un PC ou une VM différente du PC cible et du serveur de domaine)

Ou bien sans utiliser de domaine, il nous faut un PC de développement et un PC de déploiement avec une clé de Sideloading.

# Les étapes techniques pour le déploiement Sideloading

## Résumé des étapes

Voici les étapes à suivre pour un déploiement en Sideloading

1. **Générer un certificat valide**
   1. *Créer un certificat*
   2. *Convertir un certificat en PFX*
   3. *Installer le certificat*
2. **Créer le package d’application (appx)**
   1. *Configurer le manifest*
   2. *Créer appx Via visual studio (conseillé)*
   3. *Créer appx En ligne de commande*
3. **Préparer le poste cible**
   1. *Installer le certificat*
   2. *Installer la clé de Sideloading*
   3. *Autoriser les applications approuvées*
   4. *Déployer l’application*
   5. *Créer un BAT pour le poste client*
   6. *Créer un EXE*

## Signer le package avec un certificat valide

### Créer un certificat valide

Pour créer un certificat, il faut utiliser les outils Visual studio **MakeCert.exe** et **PvkPfx.exe**.

Pour cela, sur l’interface métro entrer « visual studio tools » puis ouvrir le dossier. Ouvrir, en tant qu’administrateur (clic droit), l’invite de commandes développeur pour VS2013.

L’application doit être signée avant d’être déployée.

MakeCert /n "CN=elipce" /r /h 0 /eku " 1.3.6.1.5.5.7.3.3" /e 09/23/2024 /a sha256 /sv E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.pvk E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.cer

### Convertir pvk et cer en pfx :

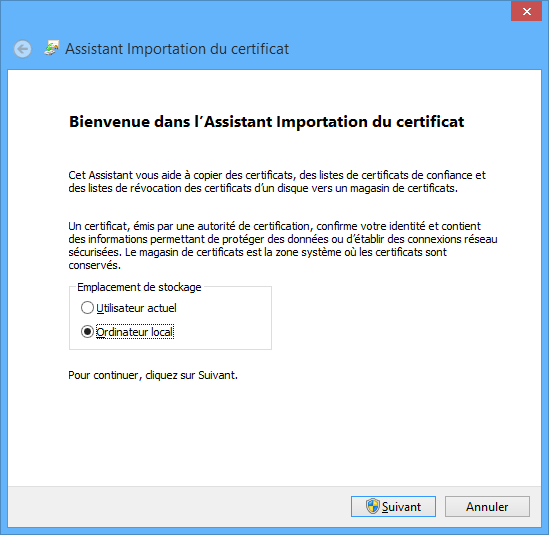
Pvk2Pfx /pvk E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.pvk /pi elipce05antipodes08 /spc E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.cer /pfx E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.pfx /po elipce05antipodes08

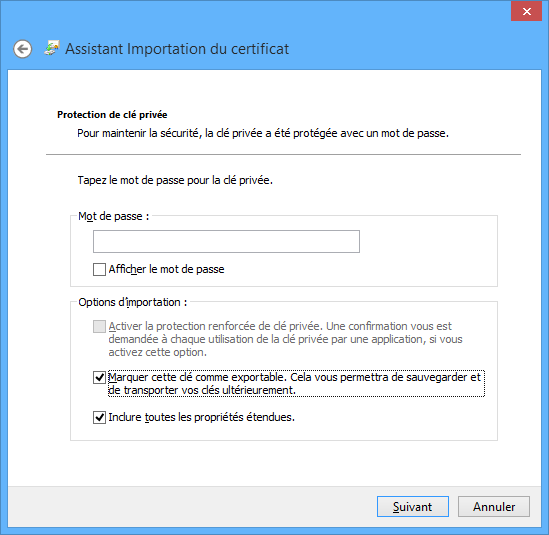
IDEM sans PWD :

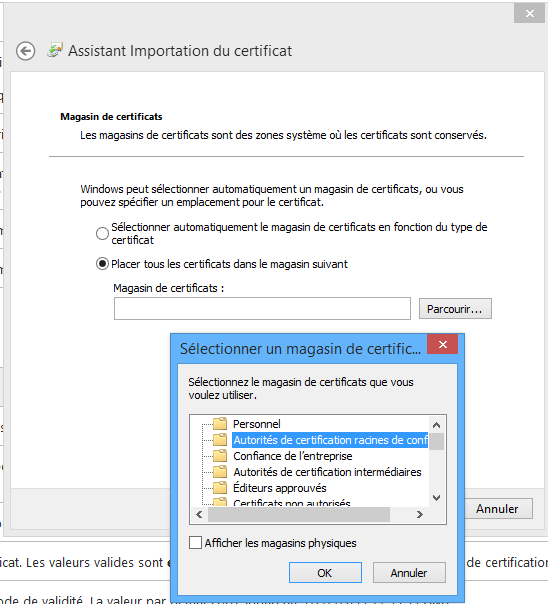
Pvk2Pfx /pvk E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.pvk /spc E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.cer /pfx E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\cert\elipcebibliodoc.pfx

### Installer le certificat

Double clic sur le PFX puis :







## Créer le package d’application

### Configurer le manifest

Ouvrir le manifest de l’application dans Visual studio. Puis aller dans l’onglet « Packages ». Cliquer sur « Choisir un certificat… » puis « Configurer le certificat … à partir d’un fichier ». Puis sélectionner le PFX et faire « OK »

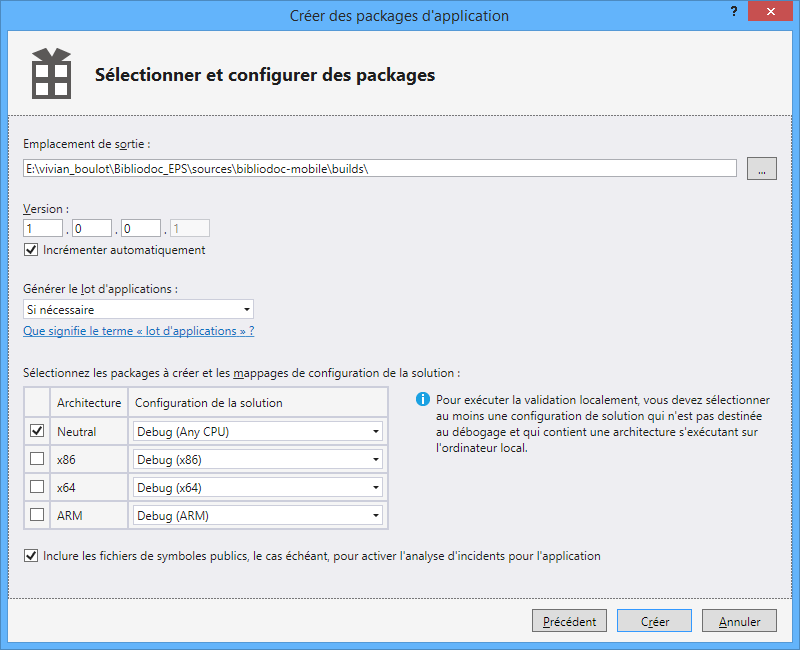
### Créer l’appx via Visual Studio 2013

Ouvrir le projet concerné dans Visual studio.

Cliquer sur « projet / Windows store / créer des packages d’application »

Ecran créer vos packages, cocher non puis cliquer sur « suivant »

Définir la version et l’emplacement de sortie puis cliquer sur « Créer ».



### Créer l’appx en ligne de commande

Ouvrir l’invite de commande VS2013 :

Sur l’écran métro taper « visual studio tools » puis choisir « Invite de commandes développeur pour VS2013. »

Entrer la commande : MSBuild MyProject.csproj /p:OutDir=C:\builds\droplocation\ /p :EnableSigningChecks=False

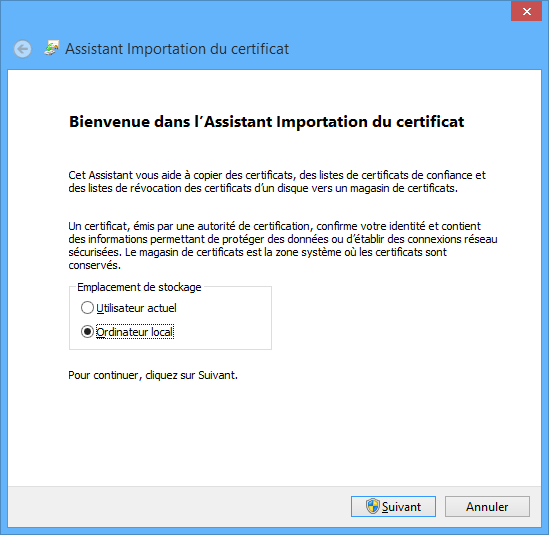
Attention : ne pas mettre d’espaces dans les chemins. Visual Studio doit être fermé.

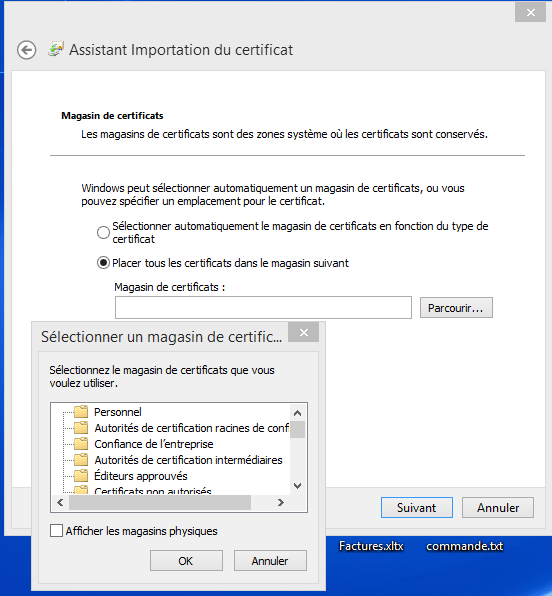
## Préparer le poste cible

Les étapes 2.4.1, 2.4.2, 2.4.3 et 2.4.4 sont réalisées en ligne de commande dans la partie 2.4.5.

### Installer le certificat public (cer)

Dans le dossier de génération de l’APPX, clic droit sur le CER puis « intaller le certificat ».





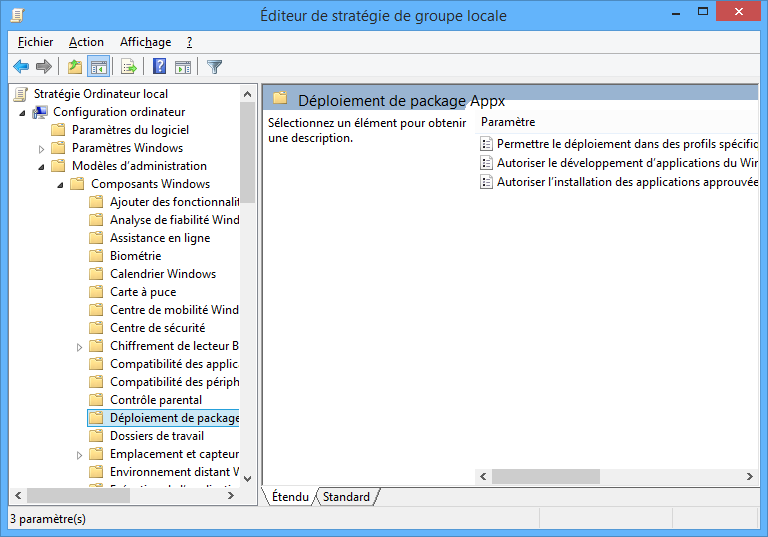
### Installer la clé de sideloading

Ouvrir une commande en mode administrateur puis :

* Slmgr /ipk <sideloading product key>
* slmgr /ato ec67814b-30e6-4a50-bf7b-d55daf729d1e

### Autoriser les applications approuvées

Ouvrir la stratégie de groupe : « Windows + r » puis « gpedit.msc »

****

**Choisir « autoriser l’installation des applications approuvées … » puis « Activé ».**

### Déployer la solution

Ouvrir powershell : sur l’interface metro taper « powershell » puis clic droit sur « windows powershell » puis « executer en tant qu’administrateur ».

Entrer la commande :

add-appxpackage E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\builds\bibliodoc-mobile\_v4\bibliodoc-mobile\_1.0.0.7\_AnyCPU\_Debug\_Test\bibliodoc-mobile\_1.0.0.7\_AnyCPU\_Debug.appx -DependencyPath E:\vivian\_boulot\Bibliodoc\_EPS\sources\bibliodoc-mobile\builds\bibliodoc-mobile\_v4\bibliodoc-mobile\_1.0.0.7\_AnyCPU\_Debug\_Test\Dependencies\Microsoft.WinJS.2.0.appx

### Créer un BAT pour le poste client

regedit /s allow.reg

Certutil -addStore TrustedPeople **%~dp0**%bibliodoc-mobile\_1.0.0.7\_AnyCPU\_Debug.cer

Slmgr /ipk 2RHXN-7CFVC-7HGCB-G2R97-M4DH7

Slmgr /ato ec67814b-30e6-4a50-bf7b-d55daf729d1e

powershell.exe -ExecutionPolicy Bypass -Command "add-appxpackage **%~dp0bibliodoc-mobile\_1**.0.0.7\_AnyCPU\_Debug.appx -DependencyPath **%~dp0Dependencies**\Microsoft.WinJS.2.0.appx"

### Créer un exécutable

A l’aide de l’outil « inno setup », créer un exécutable qui :

* Copie les fichiers sources, générés à partir de 2.4.4, dans un répertoire temporaire
* Lancer le BAT généré en partie 2.4.5
* Supprime les fichiers copiés précédemment

; Script generated by the Inno Setup Script Wizard.

; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO SETUP SCRIPT FILES!

**#define** MyAppName "Bibliodoc"

**#define** MyAppVersion "1.0"

**#define** MyAppPublisher "Elipce"

**#define** MyAppURL "www.elipce.com"

**#define** MyAppExeName "MyProg.exe"

**[Setup]**

; NOTE: The value of AppId uniquely identifies this application.

; Do not use the same AppId value in installers for other applications.

; (To generate a new GUID, click Tools | Generate GUID inside the IDE.)

**AppId**={{570DDA4C-4D8E-483C-B072-5A01A57E644F}

**AppName**=**{#MyAppName}**

**AppVersion**=**{#MyAppVersion}**

;AppVerName={#MyAppName} {#MyAppVersion}

**AppPublisher**=**{#MyAppPublisher}**

**AppPublisherURL**=**{#MyAppURL}**

**AppSupportURL**=**{#MyAppURL}**

**AppUpdatesURL**=**{#MyAppURL}**

**CreateAppDir**=no

**OutputBaseFilename**=setup

**Compression**=lzma

**SolidCompression**=yes

**[Languages]**

**Name**: "french"; **MessagesFile**: "compiler:Languages\French.isl"

**[Files]**

**Source**: "allow.reg"; **DestDir**: "{tmp}"; **Flags**: ignoreversion

**Source**: "bibliodoc-mobile\_1.0.0.7\_AnyCPU\_Debug\_Test\\*"; **DestDir**: "{tmp}"; **Flags**: ignoreversion recursesubdirs createallsubdirs

**Source**: "install.bat"; **DestDir**: "{tmp}"; **Flags**: ignoreversion

; NOTE: Don't use "Flags: ignoreversion" on any shared system files

**[Run]**

;Filename: "{src}\allow.reg"; Flags: shellexec; Parameters:"/S";

**Filename**: "{tmp}\install.bat";

# Questions:

**In the real world, if you were deploying an actual LOB app, the developer would sign it with a certificate that is chained to a trusted root certificate**

* C’est-à-dire ? Comment fait-on pour signer notre application avec un certificat dépendant d’un certificat racine approuvé ?

# Liens utiles

* **Article Microsoft Technet sur le sideloading :** <http://technet.microsoft.com/en-us/windows/jj874388.aspx>
* **Guide Sideloading Zdnet** : <http://www.zdnet.com/the-enterprise-sideloading-story-on-windows-8-its-complicated-7000006742/>
* **Nouvelle politique de déploiement** (mai 2014): <http://blogs.msdn.com/b/editeurs_de_logiciels/archive/2014/04/24/nouvelle-politique-de-d-233-ploiement-d-applications-windows-8-1-en-entreprise.aspx>
* **Création d’un certificat :** <http://msdn.microsoft.com/en-us/library/windows/desktop/jj835832(v=vs.85).aspx>
* **Guide de déploiement (MSDN) :** <http://blogs.msdn.com/b/devosaure/archive/2013/08/06/d-233-ploiement-d-une-application-windows-8-interface-moderne-en-entreprise.aspx>