

Cloner nativement Windows 8.x (mode EFI)

Dans ce tutoriel, je vais expliquer comment cloner un Windows 8.x x64 sur PC avec EFI.

L'utilité peut être par exemple :

- Changement de disque dur vers un plus gros ou plus petit
- Changement de SSD vers un plus gros ou plus petit
- Basculement d'un disque dur vers SSD ou vice-versa
- Une seule installation pour plusieurs PC à matériel identique

Il existe des logiciels gratuits comme CloneZilla, même si pour Windows la fiabilité n'est pas garantie (car le support NTFS n'est que partiel).

Il y a aussi des logiciels payants comme Paragon Drive Copy, que je trouve très peu coûteux et pratique dans le cas d'une copie directe d'un lecteur à un autre (sans passer par un réseau).

Pour un déploiement dans un réseau, j'utilise plutôt la méthode native via les outils Microsoft.

Si votre installation de Windows contient une partition spéciale de restauration et que vous voulez la conserver (facultatif, je recommande une vraie installation de Windows, non modifiée), il vous faudra adapter ce tutoriel à votre usage.

Pré-requis

Pour utiliser nativement ce clonage de PC Windows, il faudrait par exemple :

- Un Live Windows de type Windows PE 5 ([tutoriel ici pour sa création](#))
- Un disque dur ou SSD autre que celui vers lequel cloner (peut être externe), ou même une clé USB de taille suffisante

Note : on peut aussi utiliser un NAS ou un serveur de fichiers Windows (ou Linux Samba) pour centraliser la ou les images Windows.

Aide : si vous utilisez un serveur ou NAS comme stockage d'images Windows au lieu d'un disque dur, SSD ou Clé USB, ce type de commande peut vous aider (c'est mon cas) :

net use /?

Cela donne la syntaxe à utiliser pour créer un lecteur réseau.

Exemple :

```
net use S: \\192.168.0.2\Captures /user:Maxime
```

Cette commande permet d'utiliser la lettre (qui n'était pas utilisée, à vérifier auparavant) S: qui va pointer vers le partage "Captures" du serveur Windows (ou Samba) 192.168.0.2 avec le compte Maxime.

Le mot de passe est demandé, puis le lecteur est créé.

Attention : l'utilisateur pour le lecteur réseau doit avoir les droits suffisants sur le partage utilisé (droits de modification, au minimum).

Démarrage de Windows PE 5, et récolte d'informations

Une fois Windows PE démarré, par défaut il y a seulement un Invite de commandes (cmd.exe), ce qui conviendra parfaitement pour ce cas.

Si vous avez besoin du réseau et que Windows PE ne reconnaît pas votre carte réseau nativement, [voici un tutoriel expliquant comment régler cela](#).

Commençons à récolter les informations, avec **diskpart** par exemple :

Lancer diskpart, puis rentrer les commandes suivantes :

```
list disk
```

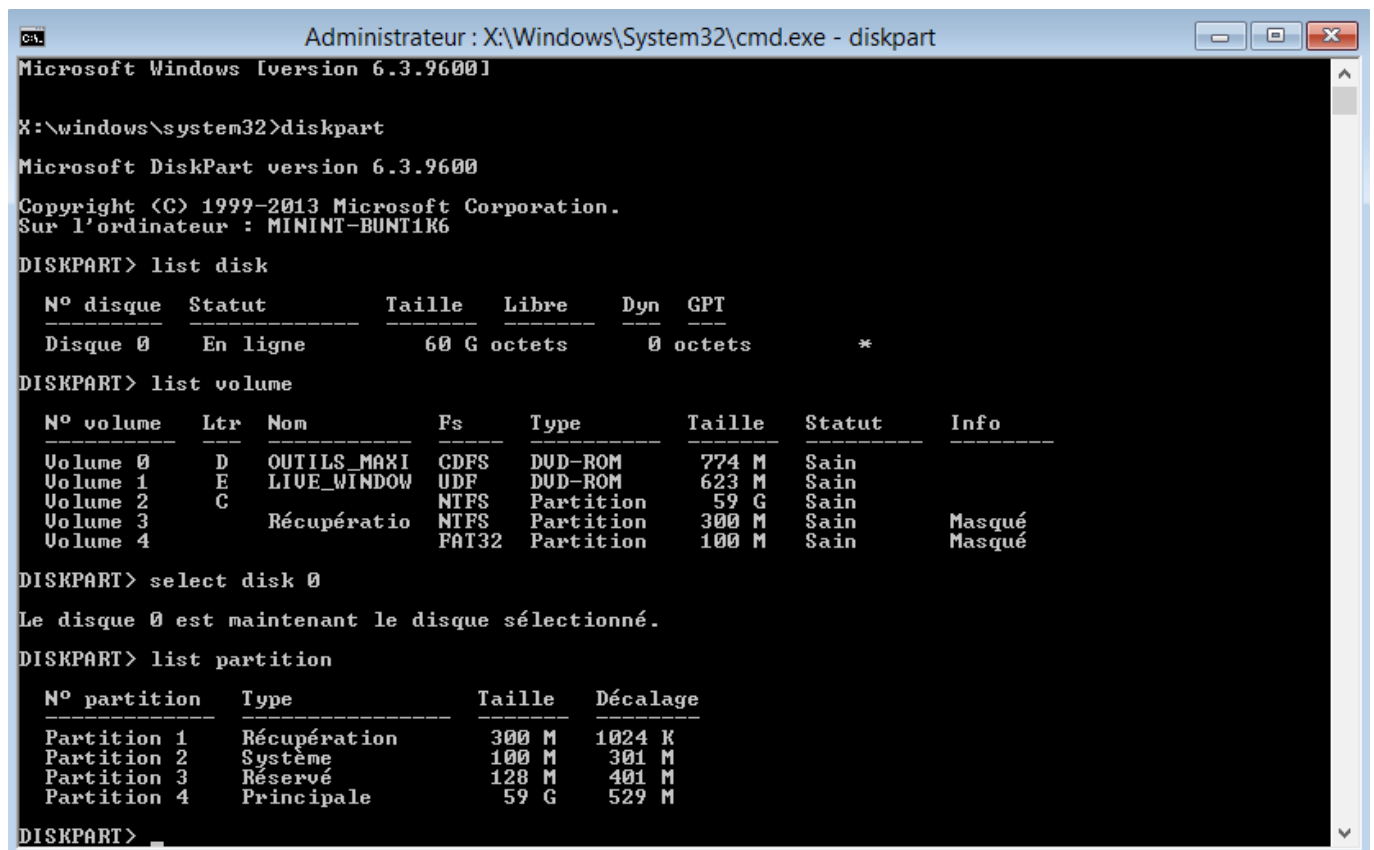
On peut aussi lister les volumes :

```
list volume
```

Si le disque ou SSD à cloner est le 0 dans la liste de **list disk**, alors :

```
select disk 0
```

```
list partition
```



```
Administrateur : X:\Windows\System32\cmd.exe - diskpart
Microsoft Windows [version 6.3.9600]

X:\windows\system32>diskpart

Microsoft DiskPart version 6.3.9600

Copyright (C) 1999-2013 Microsoft Corporation.
Sur l'ordinateur : MININT-BUNT1K6

DISKPART> list disk

   N° disque   Statut           Taille   Libre   Dyn   GPT
-----
Disque 0      En ligne        60 G octets   0 octets      *

DISKPART> list volume

   N° volume   Ltr  Nom           Fs      Type           Taille   Statut   Info
-----
Volume 0      D    OUTILS_MAXI  CDFS    DUD-ROM        774 M    Sain
Volume 1      E    LIVE_WINDOW  UDF     DUD-ROM        623 M    Sain
Volume 2      C    Récupératio  NTFS    Partition      59 G     Sain
Volume 3      C    Récupératio  NTFS    Partition      300 M    Sain    Masqué
Volume 4      C    Récupératio  FAT32   Partition      100 M    Sain    Masqué

DISKPART> select disk 0

Le disque 0 est maintenant le disque sélectionné.

DISKPART> list partition

   N° partition  Type           Taille   Décalage
-----
Partition 1    Récupération  300 M    1024 K
Partition 2    Système       100 M    301 M
Partition 3    Réservé       128 M    401 M
Partition 4    Principale    59 G     529 M

DISKPART>
```

Dans mon cas un seul "disque" est visible, car j'utilise un lecteur réseau pour les images.

Attention à bien choisir le bon disque ou SSD, et idem pour les volumes (avec leurs lettres), quitte à explorer un peu chaque lecteur via "dir" pour identification.

Dans ce "disque" (Windows écrit "disque" même pour un SSD), on peut voir 4 partitions.

Elles ont été créées par Windows lors de l'installation, voici ce qu'on trouve :

- une partition 1 "**Récupération**" de type Récupération de 300 Mo (c'est une partition principale NTFS)
- une partition 2 "**Système**" de type EFI de 100 Mo (c'est une partition FAT32, un des rares systèmes de fichiers que l'EFI peut lire)
- une partition 3 "**Réservé**" de type MSR (Microsoft Reserved Partition) de 128 Mo (en réalité, pas de données utilisateur dedans ni Windows lui-même, cette partition semble réservée à un usage non documenté de Microsoft, et elle est présente sur les systèmes EFI)
- une partition 4 "**Principale**" (en GPT de toute façon on peut créer un grand nombre de partitions principales) de 59 Go, formatée en NTFS

En correspondance on peut trouver ici dans la liste des volumes :

- Volume 2 en NTFS, 59 Go → c'est le **système entier**, la partition 4 de tout à l'heure ; lettre C
- Volume 3 en NTFS, 300 Mo → c'est la partition "**Récupération**", sans lettre par défaut
- Volume 4 en FAT32, 100 Mo → c'est la partition "**Système**" EFI, sans lettre par défaut

Note : la partition MSR de 128 Mo n'a pas de volume correspondant, elle ne semble pas formatée ou accessible.

Elle ne sera pas à copier.

Pour cloner votre disque ou SSD entier et si vous avez plusieurs autres partitions, il y aurait besoin d'une lettre par partition à copier.

Si c'est votre cas, voici le type de commande à utiliser, pour un volume sans lettre :

Dans ce cas, toujours dans "diskpart", exemple avec un volume 10 (imaginaire ici) :

```
select volume 10  
assign
```

Ainsi, une lettre serait attribuée au volume 10 imaginaire, et avec "list volume" on verrait la lettre.

Aide : la partition EFI FAT32 de 100 Mo n'est pas nécessairement à sauvegarder, car nous pourrions facilement générer de nouveaux fichiers de démarrage ensuite.

Copie du Windows PE de réparation de Windows 8.x (facultatif)

Si vous désirez copier (pour ensuite le restaurer) le Windows PE de réparation intégré à Windows, voici comment vous pouvez procéder.

En effet, depuis Windows 8.x le dossier "**Recovery**" contenant ce Windows PE a été placé dans une autre partition que celle du système.

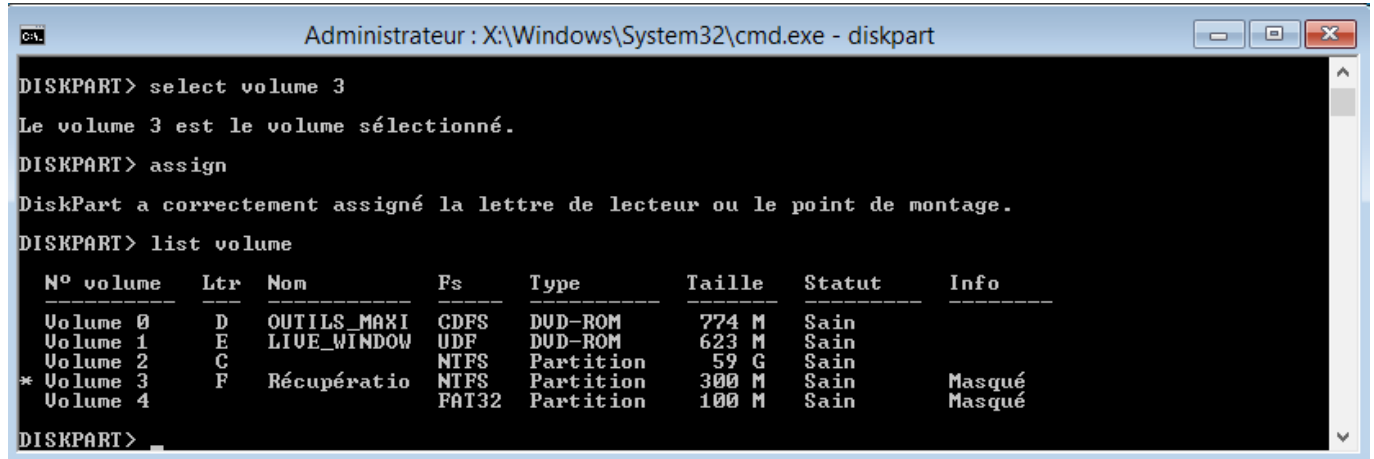
Bien sûr, il est possible de copier la partition spéciale "**Récupération**", la restaurer, et modifier ensuite le catalogue BCD du nouveau disque dur ou SSD.

J'ai néanmoins choisi une méthode que j'estime plus simple : garder uniquement les fichiers nécessaires et placer le dossier "**Recovery**" dans la partition système de Windows, comme avec Windows 7.

Dans ce cas, attribuer une lettre au **volume 3** (dans mon exemple) qui correspond à la partition "**Récupération**", via les commandes suivants dans diskpart :

```
select volume 3
assign
```

Cela donne alors :



```
DISKPART> select volume 3
Le volume 3 est le volume sélectionné.
DISKPART> assign
DiskPart a correctement assigné la lettre de lecteur ou le point de montage.
DISKPART> list volume
```

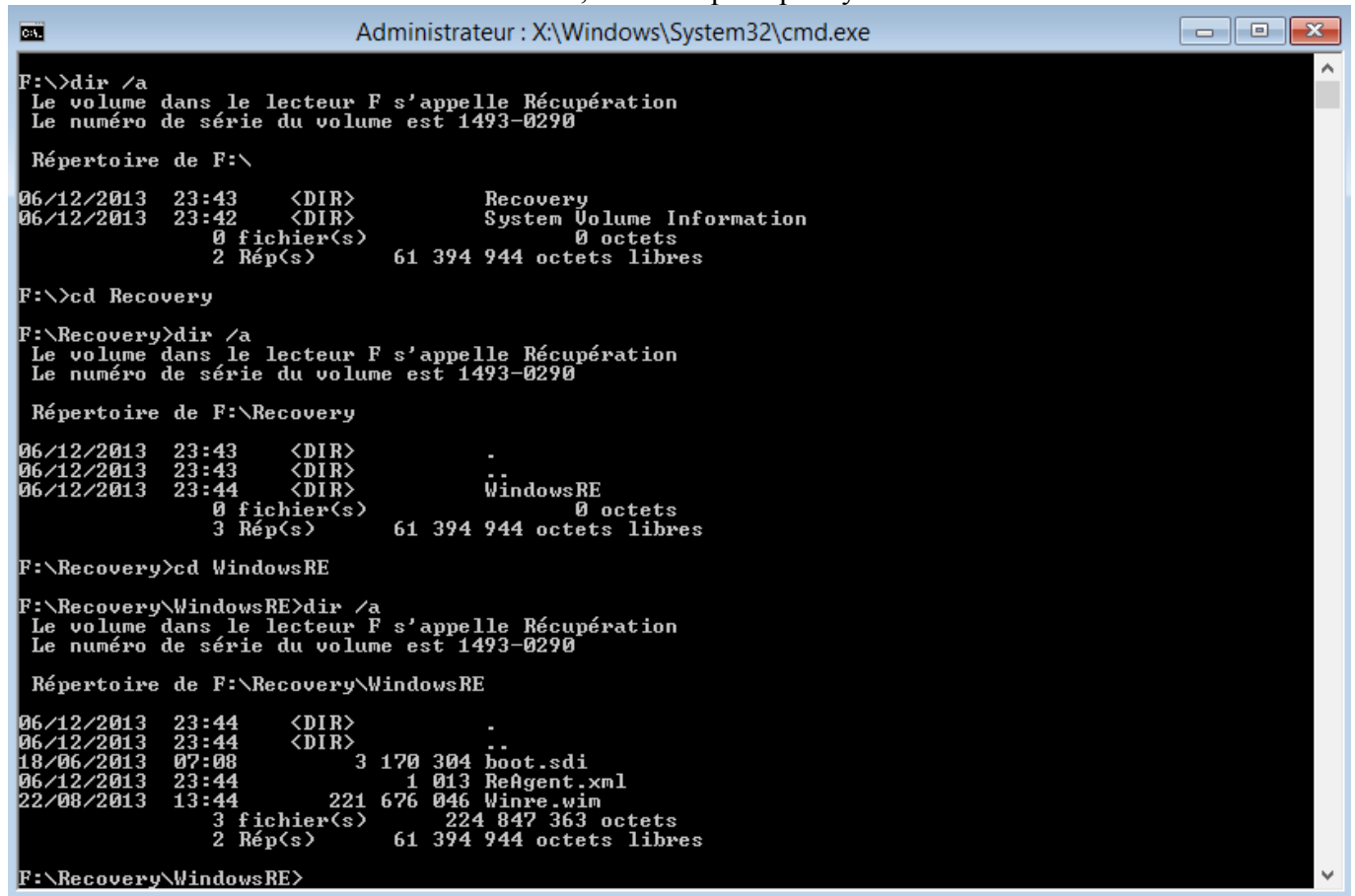
N° volume	Ltr	Nom	Fs	Type	Taille	Statut	Info
Volume 0	D	OUTILS_MAXI	CDFS	DUD-ROM	774 M	Sain	
Volume 1	E	LIVE_WINDOW	UDF	DUD-ROM	623 M	Sain	
Volume 2	C		NTFS	Partition	59 G	Sain	
* Volume 3	F	Récupératio	NTFS	Partition	300 M	Sain	Masqué
Volume 4			FAT32	Partition	100 M	Sain	Masqué

```
DISKPART>
```

On voit alors la lettre attribuée, ici "F".

Si on souhaite choisir la lettre, ce serait la commande : **assign letter=X** (où X est la lettre choisie)

En listant les fichiers et dossiers de ce lecteur, voici ce qu'on peut y trouver :



```
F:\>dir /a
Le volume dans le lecteur F s'appelle Récupération
Le numéro de série du volume est 1493-0290

Répertoire de F:\

06/12/2013 23:43 <DIR> Recovery
06/12/2013 23:42 <DIR> System Volume Information
0 fichier(s) 0 octets
2 Rép(s) 61 394 944 octets libres

F:\>cd Recovery
F:\Recovery>dir /a
Le volume dans le lecteur F s'appelle Récupération
Le numéro de série du volume est 1493-0290

Répertoire de F:\Recovery

06/12/2013 23:43 <DIR> .
06/12/2013 23:43 <DIR> ..
06/12/2013 23:44 <DIR> WindowsRE
0 fichier(s) 0 octets
3 Rép(s) 61 394 944 octets libres

F:\Recovery>cd WindowsRE
F:\Recovery\WindowsRE>dir /a
Le volume dans le lecteur F s'appelle Récupération
Le numéro de série du volume est 1493-0290

Répertoire de F:\Recovery\WindowsRE

06/12/2013 23:44 <DIR> .
06/12/2013 23:44 <DIR> ..
18/06/2013 07:08 3 170 304 boot.sdi
06/12/2013 23:44 1 013 ReAgent.xml
22/08/2013 13:44 221 676 046 Winre.wim
3 fichier(s) 224 847 363 octets
2 Rép(s) 61 394 944 octets libres

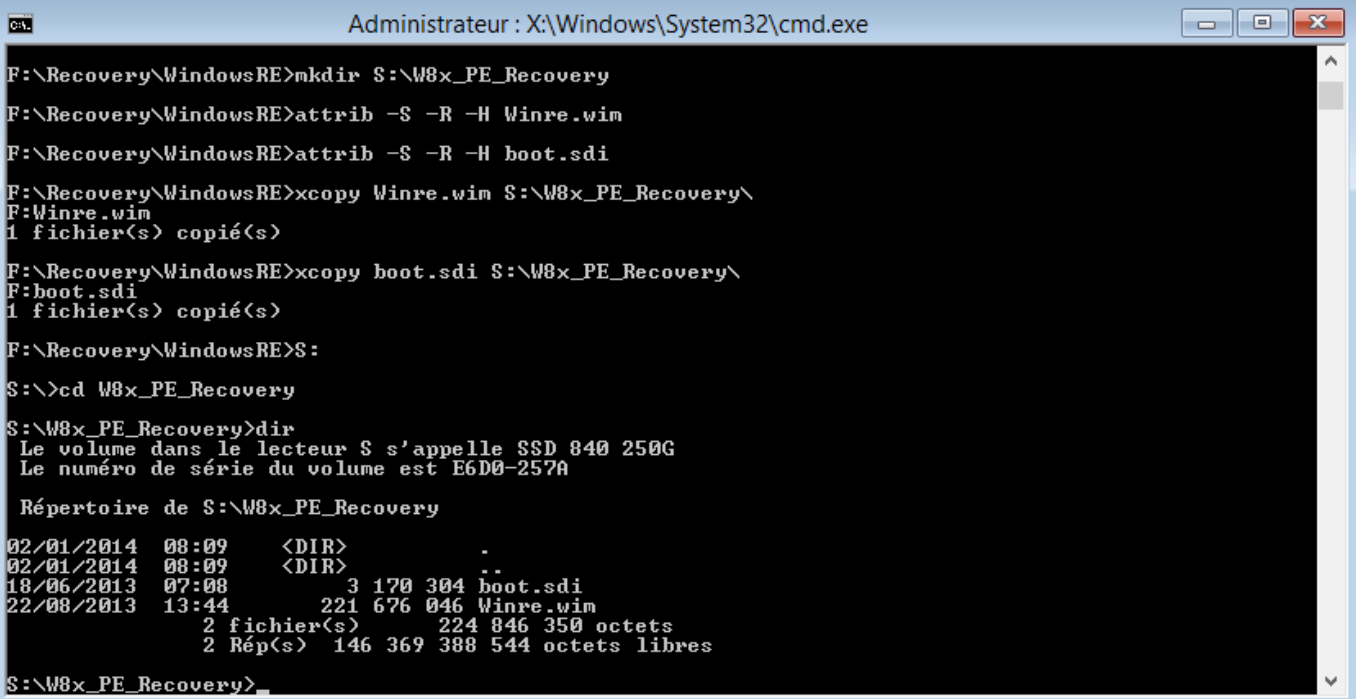
F:\Recovery\WindowsRE>
```

Seuls les fichiers **Winre.wim** et **boot.sdi** sont à garder.

Je vais utiliser la lettre **S** comme lecteur réseau ou disque dur / SSD qui va stocker les images WIM entre autres.

Adaptez bien à votre lettre et votre cas.

Voici alors de quoi copier les fichiers voulus vers ce lecteur **S** :



```
Administrateur : X:\Windows\System32\cmd.exe
F:\Recovery\WindowsRE>mkdir S:\W8x_PE_Recovery
F:\Recovery\WindowsRE>attrib -S -R -H Winre.wim
F:\Recovery\WindowsRE>attrib -S -R -H boot.sdi
F:\Recovery\WindowsRE>xcopy Winre.wim S:\W8x_PE_Recovery\
F:Winre.wim
1 fichier(s) copié(s)
F:\Recovery\WindowsRE>xcopy boot.sdi S:\W8x_PE_Recovery\
F:boot.sdi
1 fichier(s) copié(s)
F:\Recovery\WindowsRE>S:
S:\>cd W8x_PE_Recovery
S:\W8x_PE_Recovery>dir
Le volume dans le lecteur S s'appelle SSD 840 250G
Le numéro de série du volume est E6D0-257A

Répertoire de S:\W8x_PE_Recovery
02/01/2014  08:09    <DIR>          .
02/01/2014  08:09    <DIR>          ..
18/06/2013  07:08             3 170 304 boot.sdi
22/08/2013  13:44            221 676 046 Winre.wim
                2 fichier(s)          224 846 350 octets
                2 Rép(s)    146 369 388 544 octets libres

S:\W8x_PE_Recovery>
```

J'ai créé un dossier (inventé) pour les copier : **S : \W8x_PE_Recovery**

Les fichiers **Winre.wim** et **boot.sdi** étant cachés et système par défaut, j'ai dû utiliser la commande **attrib** pour changer les attributs pour qu'ensuite **xcopy** veuille bien les copier.

Création des images WIM

Comme indiqué précédemment, je vais utiliser la lettre **S** comme lecteur réseau ou disque dur / SSD qui va stocker les images WIM et des fichiers temporaires.

Adaptez bien à votre lettre et votre cas.

Le lecteur de destination n'a pas besoin d'être vide, il y aura simplement un ou plusieurs fichier(s) .WIM créés.

Le logiciel de capture (DISM) utilisant des fichiers temporaires, il est préférable de modifier les variables d'environnement **TEMP** et **TMP** afin que les fichiers temporaires ne soient pas créés dans le mini lecteur **X**: qui est en RAM et limité en taille (la capture peut planter).

Au plus simple, on peut créer un dossier "Temp" dans S:

La commande est "**mkdir nom-du-dossier**".

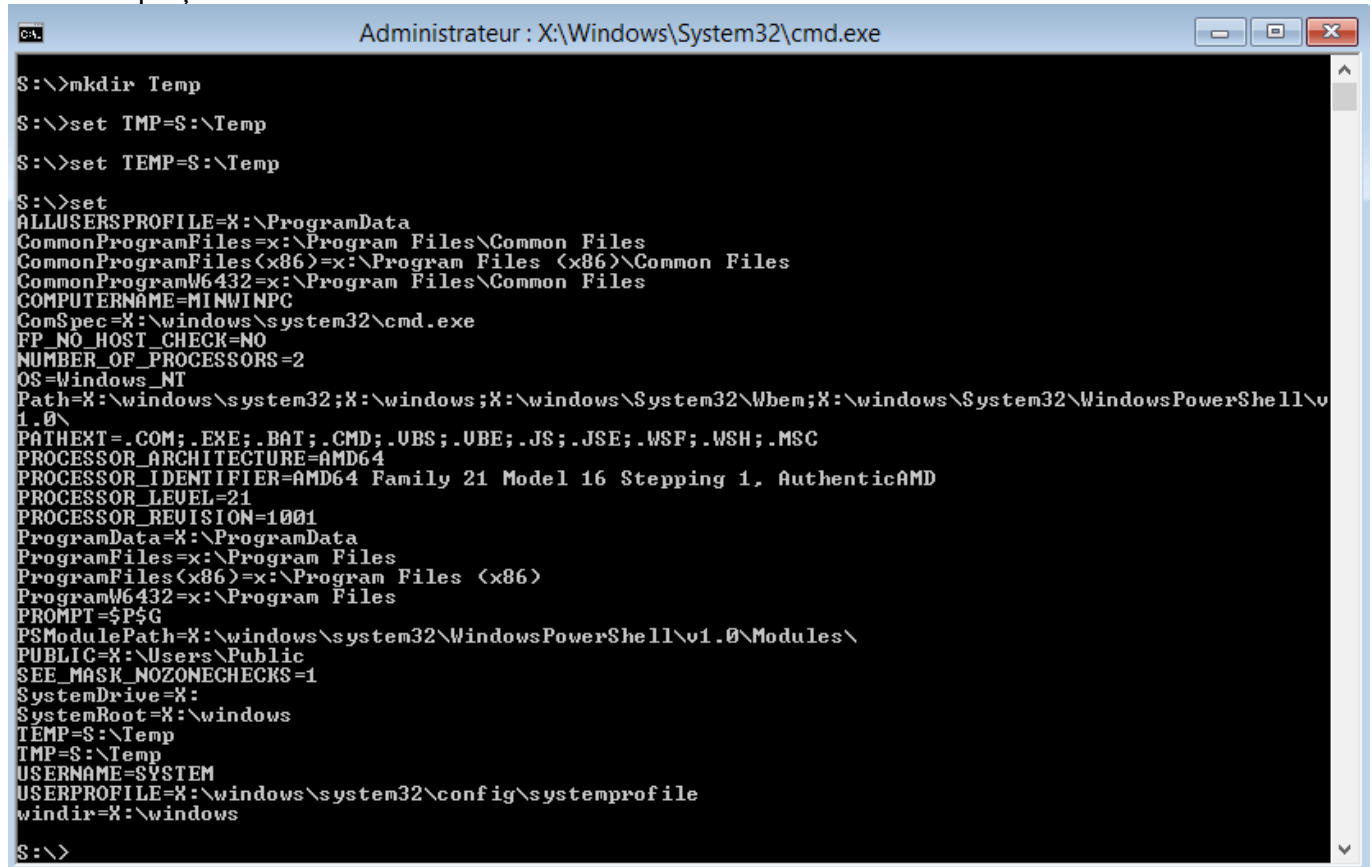
Les commandes pour changer les variables sont alors :

```
set TMP=S:\Temp
```

```
set TEMP=S:\Temp
```

On peut même vérifier que ça fonctionne en listant toutes les variables d'environnement, via la commande "set".

Voici un aperçu :



```
Administrateur : X:\Windows\System32\cmd.exe
S:\>mkdir Temp
S:\>set TMP=S:\Temp
S:\>set TEMP=S:\Temp
S:\>set
ALLUSERSPROFILE=X:\ProgramData
CommonProgramFiles=x:\Program Files\Common Files
CommonProgramFiles(x86)=x:\Program Files (x86)\Common Files
CommonProgramW6432=x:\Program Files\Common Files
COMPUTERNAME=MINWINPC
ComSpec=X:\windows\system32\cmd.exe
FP_NO_HOST_CHECK=NO
NUMBER_OF_PROCESSORS=2
OS=Windows_NT
Path=X:\windows\system32;X:\windows;X:\windows\System32\Wbem;X:\windows\System32\WindowsPowerShell\v1.0\
PATHEXT=.COM;.EXE;.BAT;.CMD;.UBS;.UBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE=AMD64
PROCESSOR_IDENTIFIER=AMD64 Family 21 Model 16 Stepping 1, AuthenticAMD
PROCESSOR_LEVEL=21
PROCESSOR_REVISION=1001
ProgramData=X:\ProgramData
ProgramFiles=x:\Program Files
ProgramFiles(x86)=x:\Program Files (x86)
ProgramW6432=x:\Program Files
PROMPT=$P$G
PSModulePath=X:\windows\system32\WindowsPowerShell\v1.0\Modules\
PUBLIC=X:\Users\Public
SEE_MASK_NOZONECHECKS=1
SystemDrive=X:
SystemRoot=X:\windows
TEMP=S:\Temp
TMP=S:\Temp
USERNAME=SYSTEM
USERPROFILE=X:\windows\system32\config\systemprofile
windir=X:\windows
S:\>
```

Et maintenant capture de la partition système...

La syntaxe est :

```
disk /Capture-Image /ImageFile:S:\nom_image.wim
/CaptureDir:Lettre:\ /Name:nom_image /Compress:fast
```

"Lettre:" est à remplacer par le lecteur à cloner, et "nom_image" par le nom de votre image, sans espace.

Pour l'option "**Compress**" on peut utiliser 3 options possibles :

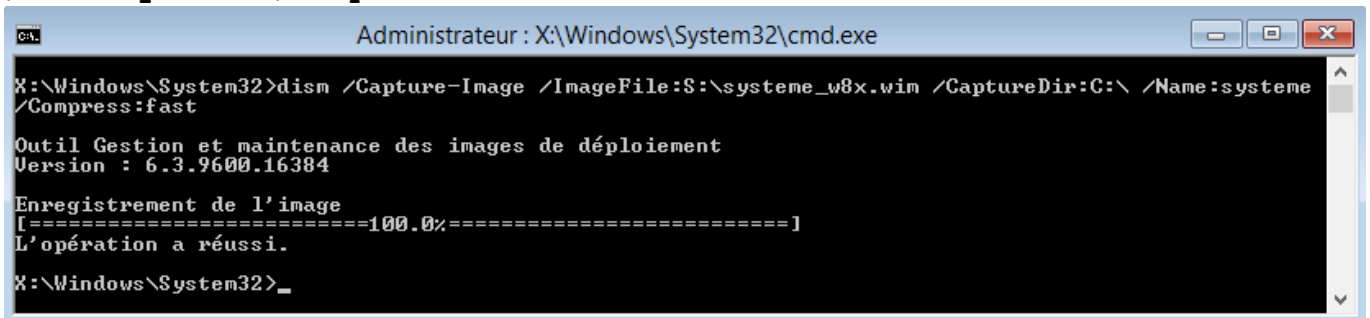
- **max** (Le fichier prend moins de place, et le transfert est plus rapide si le processeur arrive à suivre la compression à la volée)
- **fast** (Le fichier prend un peu plus de place qu'en mode maximum, cela consomme moins de processeur mais utilise plus votre réseau ou stockage qui va recevoir les images WIM)
- **none** (Aucune compression, peut prendre beaucoup de place, utilise peu le processeur, mais utilise beaucoup le stockage de destination et éventuellement le réseau si vous vous en servez)

La plupart du temps le mode "**fast**" suffira, sauf si le réseau ou disque dur (ou SSD) de destination plutôt lent, dans ce cas ce sera le mode **max** (seulement pour processeur assez puissant).

Je déconseille le mode "**none**"...

Exemple de clonage avec 1 commande (1 commande par volume à copier) :

```
dism /Capture-Image /ImageFile:S:\systeme_w8x.wim /CaptureDir:C:\
/Name:systeme /Compress:fast
```



```
Administrateur : X:\Windows\System32\cmd.exe
X:\Windows\System32>dism /Capture-Image /ImageFile:S:\systeme_w8x.wim /CaptureDir:C:\ /Name:systeme
/Compress:fast
Outil Gestion et maintenance des images de déploiement
Version : 6.3.9600.16384
Enregistrement de l'image
[=====100.0%=====]
L'opération a réussi.
X:\Windows\System32>_
```

Ici, le "**systeme_w8x.wim**" a donné 3,22 Go

Cette valeur peut changer selon les logiciels installés et les fichiers personnels présents dans l'ordinateur.

Il ne reste plus qu'à éteindre cet ordinateur, via la commande :

wpeutil shutdown

Restauration des images WIM

Sur le nouveau disque dur ou SSD, ou sur le nouvel ordinateur (**matériel identique**), démarrer sur Windows PE 5.

A moins que vous ayez besoin de restaurer / cloner seulement une des partitions, et dans ce cas il faudra adapter à votre usage, nous allons utiliser **diskpart** pour supprimer toutes les partitions éventuelles présentes.

Volontairement, j'ai mis un "disque" plus petit dans ce nouvel ordinateur, pour montrer que c'est tout à fait possible, tant que la quantité réelle de données rentre à l'intérieur.

Dans diskpart, lister les "disques" :

list disk

Choisir le bon (pour moi il n'y en a qu'un, j'utilise un lecteur réseau pour les images WIM) via :
select disk numero_disque

Ici ce sera :

select disk 0

Effacer toute partition, et retour en mode MBR par défaut (si on crée des partitions via diskpart) :

clean


```
Administrateur : X:\Windows\System32\cmd.exe - diskpart
X:\windows\system32>diskpart
Microsoft DiskPart version 6.3.9600
Copyright (C) 1999-2013 Microsoft Corporation.
Sur l'ordinateur : MININT-HTA8864
DISKPART> list disk

  N° disque  Statut      Taille  Libre  Dyn  GPT
-----
Disque 0    En ligne    50 G octets  2048 K octets  *

DISKPART> select disk 0
Le disque 0 est maintenant le disque sélectionné.
DISKPART> clean
DiskPart a réussi à nettoyer le disque.
DISKPART> _
```

Maintenant nous allons exécuter des commandes dans diskpart qui vont permettre de :

- Convertir le "disque" en GPT (GUID Partition Table)
- Créer la partition EFI contenant le chargeur de démarrage (100 Mo), la formater en **FAT32** et lui assigner une lettre (ici **Y**)
- Créer la partition MSR de 128 Mo (non formatée ensuite)
- Créer la partition du système (tout l'espace restant), la formater en **NTFS** et lui assigner une lettre (ici **Z**)

convert gpt

```
create partition efi size=100
format FS=FAT32 QUICK
assign letter=Y
```

```
create partition msr size=128
```

```
create partition primary
format FS=NTFS QUICK
assign letter=Z
```

On obtient alors :

```
Administrateur : X:\Windows\System32\cmd.exe - diskpart
DISKPART> convert gpt
DiskPart a correctement converti le disque sélectionné au format GPT.
DISKPART> create partition efi size=100
DiskPart a réussi à créer la partition spécifiée.
DISKPART> format FS=FAT32 QUICK
    100 pour cent effectués
DiskPart a formaté le volume.
DISKPART> assign letter=Y
DiskPart a correctement assigné la lettre de lecteur ou le point de montage.
DISKPART> create partition msr size=128
DiskPart a réussi à créer la partition spécifiée.
DISKPART> create partition primary
DiskPart a réussi à créer la partition spécifiée.
DISKPART> format FS=NTFS QUICK
    100 pour cent effectués
DiskPart a formaté le volume.
DISKPART> assign letter=Z
DiskPart a correctement assigné la lettre de lecteur ou le point de montage.
DISKPART>
```

Et voici la liste des volumes :

```
Administrateur : X:\Windows\System32\cmd.exe - diskpart
DISKPART> list volume

  N° volume  Ltr  Nom                Fs      Type           Taille  Statut  Info
-----
  Volume 0   E    OUTILS_MAXI        CDFS    DUD-ROM        774 M   Sain
  Volume 1   F    LIVE_WINDOW        UDF     DUD-ROM        623 M   Sain
  * Volume 2  Z                      NTFS    Partition      49 G    Sain
  Volume 3   Y                      FAT32   Partition      100 M   Sain   Masqué

DISKPART>
```

Dans mon cas le volume 3 est la partition EFI de démarrage (100 Mo) formatée en **FAT32**, et le volume 2 est la partition Principale du système formatée en **NTFS**.

Aussi, nous avons assigné une lettre directement, pour cet exemple j'ai pris les lettres "Y" et "Z" car je savais qu'elles n'étaient pas utilisées.

Il y a donc la lettre **Y** pour la partition EFI FAT32 de 100 Mo.

Aussi, il y a la lettre **Z** pour la partition Principale NTFS de 49 Go, prévue pour le système.

On peut quitter diskpart, et passer à l'application des images sur chaque partition, en considérant que le lecteur **S:** correspond toujours à l'endroit où se trouvent les images WIM.

Attention à bien définir cette fois aussi les variables d'environnement.

Volontairement, je vais placer le dossier temporaire sur le volume de destination si j'utilise un lecteur réseau, car autrement l'erreur "**Le client ne dispose pas d'un privilège nécessaire.**" peut apparaître, ou alors vérifier que l'utilisateur pour le lecteur réseau **S:** fait partie du groupe Administrateurs du PC ayant

créé ce partage réseau, et bien vérifier aussi que les droits sont au maximum.

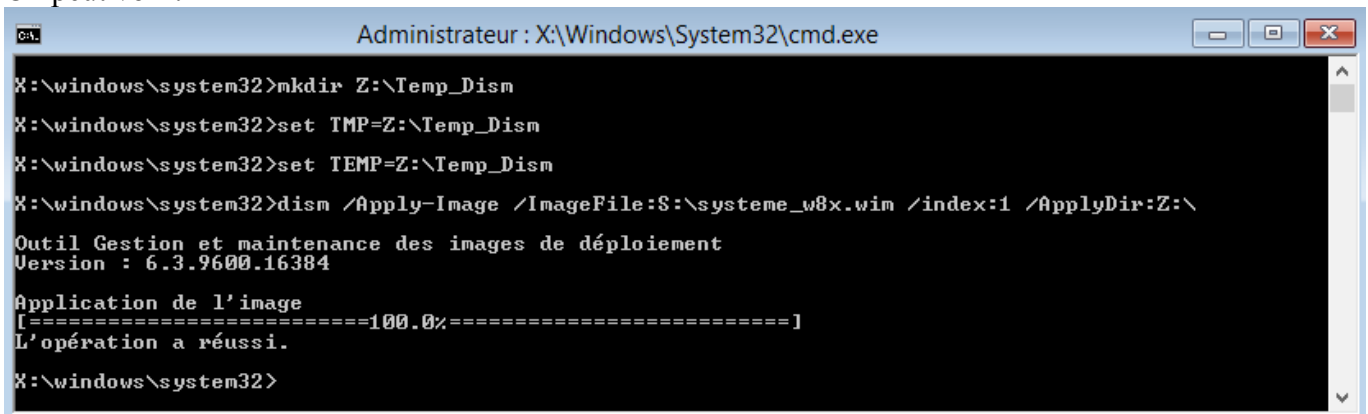
Au plus simple : placer le dossier "Temp" directement sur le lecteur de destination du système, dans mes tests cela a bien fonctionné, via les commandes (pour le lecteur **Z:**) :

```
mkdir Z:\Temp_Dism
set TMP=Z:\Temp_Dism
set TEMP=Z:\Temp_Dism
```

Et l'application (restauration) de l'image WIM :

```
dism /Apply-Image /ImageFile:S:\systeme_w8x.wim /index:1 /ApplyDir:Z:\
```

On peut voir :



```
Administrateur : X:\Windows\System32\cmd.exe
X:\windows\system32>mkdir Z:\Temp_Dism
X:\windows\system32>set TMP=Z:\Temp_Dism
X:\windows\system32>set TEMP=Z:\Temp_Dism
X:\windows\system32>dism /Apply-Image /ImageFile:S:\systeme_w8x.wim /index:1 /ApplyDir:Z:\
Outil Gestion et maintenance des images de déploiement
Version : 6.3.9600.16384
Application de l'image
[=====100.0%=====]
L'opération a réussi.
X:\windows\system32>
```

(Penser à supprimer le dossier **Z:\Temp_Dism** à la fin)

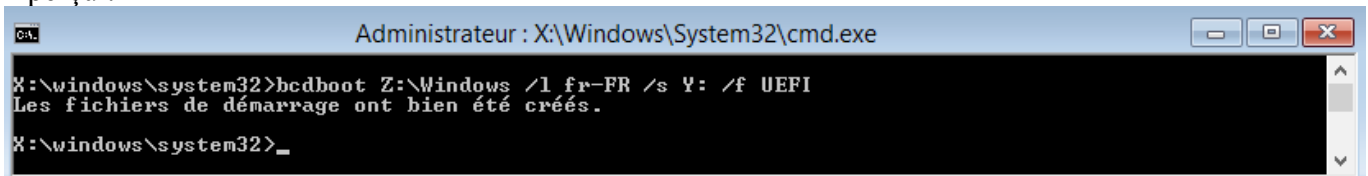
Il reste à générer un nouveau BCD (catalogue contenant les entrées de démarrage de Windows 8.x) afin que le système démarre, car nous n'avons volontairement pas gardé le BCD du système.

Solution de réparation du BCD la plus simple :

- Générer un nouveau BCD, via :

```
bcdboot Z:\Windows /l fr-FR /s Y: /f UEFI
```

Aperçu :



```
Administrateur : X:\Windows\System32\cmd.exe
X:\windows\system32>bcdboot Z:\Windows /l fr-FR /s Y: /f UEFI
Les fichiers de démarrage ont bien été créés.
X:\windows\system32>_
```

Pour plus de détails sur cette commandes, vous pouvez utiliser : **bcdboot /?**

Cette méthode est la plus simple et rapide, même si ce n'est pas la seule possible (on aurait pu réparer les entrées existantes via **bcdedit** et en modifiant le catalogue BCD se trouvant dans Y:\Boot).

En allant dans le dossier **Y:\EFI\Microsoft\Boot**, on obtient alors via "**bcdedit /store BCD**" :

```
Administrateur : X:\Windows\System32\cmd.exe
X:\windows\system32>Y:
Y:\>cd EFI\Microsoft\Boot
Y:\EFI\Microsoft\Boot>bcdedit /store BCD

Gestionnaire de démarrage Windows
-----
identificateur      <bootmgr>
device              partition=Y:
path                \EFI\Microsoft\Boot\bootmgfw.efi
description         Windows Boot Manager
locale              fr-FR
inherit              <globalsettings>
default              <default>
resumeobject        <690e56b5-7427-11e3-bb2b-000c299b4128>
displayorder        <default>
toolsdisplayorder   <memdiag>
timeout             30

Chargeur de démarrage Windows
-----
identificateur      <default>
device              partition=Z:
path                \Windows\system32\winload.efi
description         Windows 8.1
locale              fr-FR
inherit              <bootloadersettings>
isolatedcontext     Yes
allowedinmemorysettings 0x15000075
osdevice            partition=Z:
systemroot          \Windows
resumeobject        <690e56b5-7427-11e3-bb2b-000c299b4128>
nx                  OptIn
bootmenupolicy      Standard
detecthal           Yes

Y:\EFI\Microsoft\Boot>
```

Si vous le souhaitez, il est possible de réactiver le menu F8 du démarrage, désactivé par défaut dans Windows 8.x

Pour cela, utiliser la commande :

```
bcdedit /store BCD /set {default} bootmenupolicy Legacy
```

```
Administrateur : X:\Windows\System32\cmd.exe
Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set {default} bootmenupolicy Legacy
L'opération a réussi.
Y:\EFI\Microsoft\Boot>
```

Il reste une étape facultative, qui va permettre de restaurer aussi le Windows PE de réparation et dépannage de Windows, qui peut apparaître aussi via le menu "Réparer" du menu F8 au démarrage de Windows (si on a réactivé le menu F8 de Windows 8.x).

Si vous ne faites pas cette étape facultative, il suffit de redémarrer, via le commande : **wpeutil reboot**

Le Windows 8 x64 (mode EFI) cloné démarre bien.

Si vous souhaitez restaurer aussi ce menu "Réparer", alors continuez ce tutoriel.

Etape facultative : Restauration du Windows PE de dépannage fourni par Microsoft

Comme indiqué dans une précédente étape facultative, par défaut en installant Windows 8.x, dans un volume nommé "Récupération" il y avait un dossier caché et système "Recovery" qui contenait une

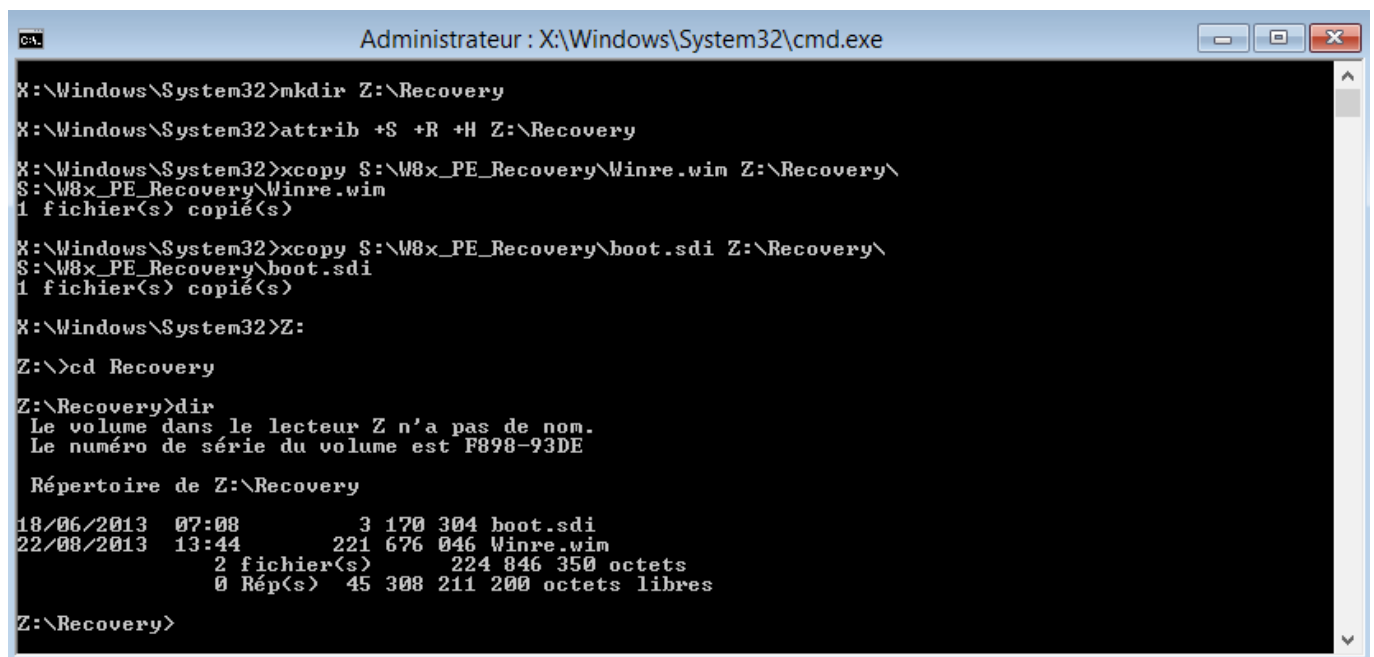
image WIM d'un Windows PE permettant de réparer Windows 8.x au moins de manière basique.

Si vous aviez bien copié les fichiers **Winre.wim** et **boot.sdi** qui étaient indiqués, alors vous pouvez continuer.

Pour rappel, j'ai placé ces fichiers dans le dossier suivant : **S:\W8x_PE_Recovery**

Dans un 1^{er} temps, le but est de créer un dossier "**Recovery**" à la racine du lecteur système et copier ces 2 fichiers **Winre.wim** et **boot.sdi** pour dans un 2^{ème} temps les intégrer au catalogue BCD.

Voici une possibilité pour y arriver :



```
Administrateur : X:\Windows\System32\cmd.exe
X:\Windows\System32>mkdir Z:\Recovery
X:\Windows\System32>attrib +S +R +H Z:\Recovery
X:\Windows\System32>xcopy S:\W8x_PE_Recovery\Winre.wim Z:\Recovery\
S:\W8x_PE_Recovery\Winre.wim
1 fichier(s) copié(s)
X:\Windows\System32>xcopy S:\W8x_PE_Recovery\boot.sdi Z:\Recovery\
S:\W8x_PE_Recovery\boot.sdi
1 fichier(s) copié(s)
X:\Windows\System32>Z:
Z:\>cd Recovery
Z:\Recovery>dir
Le volume dans le lecteur Z n'a pas de nom.
Le numéro de série du volume est F898-93DE

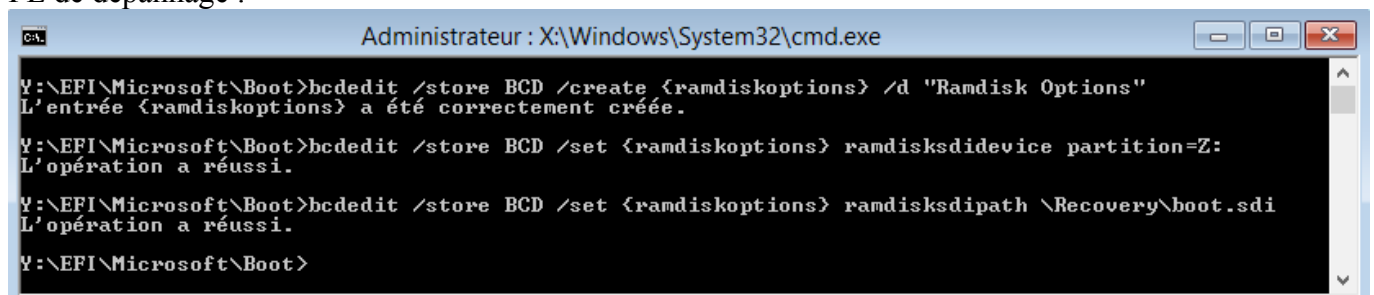
Répertoire de Z:\Recovery
18/06/2013  07:08          3 170 304 boot.sdi
22/08/2013  13:44        221 676 046 Winre.wim
            2 fichier(s)          224 846 350 octets
            0 Rép(s)          45 308 211 200 octets libres

Z:\Recovery>
```

Il y a donc eu une création du dossier **Z:\Recovery** puis via attrib je l'ai masqué car c'est un dossier système, et enfin j'ai copié les fichiers **Winre.wim** et **boot.sdi** dedans.

Maintenant, se placer dans le dossier **Y:\EFI\Microsoft\Boot** car c'est là que se trouve le catalogue **BCD** que nous allons modifier pour ajouter l'entrée "Réparer" dans le menu F8 du démarrage.

Tout d'abord, il y a la création d'une entrée qui permettra qu'un RAM Disk soit utilisé pour le Windows PE de dépannage :



```
Administrateur : X:\Windows\System32\cmd.exe
Y:\EFI\Microsoft\Boot>bcdedit /store BCD /create {ramdiskoptions} /d "Ramdisk Options"
L'entrée {ramdiskoptions} a été correctement créée.
Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set {ramdiskoptions} ramdisksdidevice partition=Z:
L'opération a réussi.
Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set {ramdiskoptions} ramdisksdipath \Recovery\boot.sdi
L'opération a réussi.
Y:\EFI\Microsoft\Boot>
```

Comme indiqué sur la capture précédente, voici les commandes bcdedit qui ont été utilisées :

```
bcdedit /store BCD /create {ramdiskoptions} /d "Ramdisk Options"  
bcdedit /store BCD /set {ramdiskoptions} ramdiskdevice partition=Z:  
bcdedit /store BCD /set {ramdiskoptions} ramdiskpath \Recovery\boot.sdi
```

Ensuite, il y a la création d'une entrée pour un Windows PE dédié à la réparation de Windows, via la commande suivante (à écrire en une seule fois) :

```
bcdedit /store BCD /create /d "Windows Recovery Environment"  
/application osloader
```

On obtient alors un ID entre accolades {} que l'on va réutiliser.

Exemple : {123abcd-98765-fedcba-0123456789abcd}

Nous avons alors ensuite (chaque commande bcdedit est à écrire en une seule fois, c'est juste que ça continue sur une 2ème ligne avec les commandes un peu longues) :

```
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd} device  
ramdisk=[Z:]\Recovery\Winre.wim,{ramdiskoptions}  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd} osdevice  
ramdisk=[Z:]\Recovery\Winre.wim,{ramdiskoptions}  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd} path  
\windows\system32\winload.efi  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd} inherit  
{bootloadersettings}  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd}  
displaymessage Recovery  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd}  
displaymessageoverride Recovery  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd}  
systemroot \windows  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd} nx OptIn  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd}  
bootmenupolicy Standard  
  
bcdedit /store BCD /set {123abcd-98765-fedcba-0123456789abcd} winpe Yes
```

Cela donne alors :

```
Administrateur : X:\Windows\System32\cmd.exe

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /create /d "Windows Recovery Environment" /application osloader
L'entrée <4f4a1040-742c-11e3-bb2b-000c299b4128> a été correctement créée.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> device ramdisk=[Z:]\Recovery\Winre.wim,<ramdiskoptions>
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> osdevice ramdisk=[Z:]\Recovery\Winre.wim,<ramdiskoptions>
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> path \windows\system32\winload.efi
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> inherit <bootloadersettings>
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> displaymessage Recovery
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> displaymessageoverride Recovery
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> systemroot \windows
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> nx OptIn
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> bootmenupolicy Standard
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <4f4a1040-742c-11e3-bb2b-000c299b4128> winpe Yes
L'opération a réussi.

Y:\EFI\Microsoft\Boot>
```

Il reste à définir que cette entrée nouvellement créée devienne celle à charger quand on demande le mode de réparation de Windows, via les commandes suivantes (réutiliser l'ID entre accolades, à nouveau) :

```
bcdedit /store BCD /set {default} recoverysequence {123abcd-98765-fedcba-0123456789abcd}
```

```
bcdedit /store BCD /set {default} recoveryenabled Yes
```

Aperçu en image :

```
Administrateur : X:\Windows\System32\cmd.exe

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <default> recoverysequence <4f4a1040-742c-11e3-bb2b-000c299b4128>
L'opération a réussi.

Y:\EFI\Microsoft\Boot>bcdedit /store BCD /set <default> recoveryenabled Yes
L'opération a réussi.

Y:\EFI\Microsoft\Boot>
```

Si on souhaite vérifier que tout est bien présent, on peut utiliser la commande :

```
bcdedit /store BCD /enum all
```

Il y a une partie au début qui nous correspond :

```
Administrateur : X:\Windows\System32\cmd.exe
Y:\EFI\Microsoft\Boot>bcdedit /store BCD /enum all
Gestionnaire de démarrage Windows
-----
identificateur      <bootmgr>
device              partition=Y:
path                \EFI\Microsoft\Boot\bootmgfw.efi
description         Windows Boot Manager
locale              fr-FR
inherit             <globalsettings>
default             <default>
resumeobject        <690e56b5-7427-11e3-bb2b-000c299b4128>
displayorder        <default>
toolsdisplayorder  <memdiag>
timeout             30

Chargeur de démarrage Windows
-----
identificateur      <4f4a1040-742c-11e3-bb2b-000c299b4128>
device              ramdisk=[Z: ]\Recovery\Winre.wim,<ramdiskoptions>
path                \windows\system32\winload.efi
description         Windows Recovery Environment
inherit             <bootloadersettings>
displaymessage      Recovery
displaymessageoverride
osdevice            ramdisk=[Z: ]\Recovery\Winre.wim,<ramdiskoptions>
systemroot          \windows
nx                  OptIn
bootmenupolicy      Standard
winpe               Yes

Chargeur de démarrage Windows
-----
identificateur      <default>
device              partition=Z:
path                \Windows\system32\winload.efi
description         Windows 8.1
locale              fr-FR
inherit             <bootloadersettings>
recoverysequence    <4f4a1040-742c-11e3-bb2b-000c299b4128>
recoveryenabled     Yes
isolatedcontext      Yes
allowedinmemorysettings
osdevice            partition=Z:
systemroot          \Windows
resumeobject        <690e56b5-7427-11e3-bb2b-000c299b4128>
nx                  OptIn
bootmenupolicy      Legacy
detecthal           Yes
```

Puis une autre partie à la fin :

```
Administrateur : X:\Windows\System32\cmd.exe
Options Ramdisk du programme d'installation
-----
identificateur      <ramdiskoptions>
description         Ramdisk Options
ramdiskdevice       partition=Z:
ramdiskpath         \Recovery\boot.sdi
Y:\EFI\Microsoft\Boot>
```

Tout est bien présent.

En redémarrant (et en quittant Windows PE donc), en demandant le menu de démarrage via la touche **F8** on doit avoir plusieurs options dont "**Réparer**" :

Options de démarrage avancées

Choisissez les options avancées pour : Windows 8.1
(Utilisez les touches fléchées pour mettre votre choix en surbrillance.)

Réparer l'ordinateur

Mode sans échec

Mode sans échec avec prise en charge réseau

Invite de commandes en mode sans échec

Inscrire les événements de démarrage dans le journal

Activer la vidéo basse résolution

Mode débogage

Désactiver le redémarrage automatique en cas d'échec du système

Désactiver le contrôle obligatoire des signatures de pilotes

Désactiver le pilote de protection contre les programmes malveillants

Démarrer Windows normalement

Description : Affiche la liste des outils de récupération du système qui vous permettent de réparer les problèmes de démarrage, exécuter des diagnostics ou restaurer votre système.

Entrée=Choisir

Échap=Annuler

Et cela permet d'obtenir cette interface :

