

## Usage:

```
MbrFix /drive <num> <command> { /yes } { /byte }
```

## Requirements:

The program is made for Windows NT, Windows 2000, Windows XP, Windows Server 2003 and Windows PE.

You need administrative privileges to make it work!

## Commands:

MbrFix /drive <num> driveinfo	Display drive information
MbrFix /drive <num> listpartitions	Display partition information
MbrFix /drive <num> savembr <file>	Save MBR and partitions to file
MbrFix /drive <num> restorembr <file>	Restore MBR and partitions from file
MbrFix /drive <num> fixmbr	Update MBR code to W2K/XP/2003
MbrFix /drive <num> clean	Delete partitions in MBR
MbrFix /drive <num> readsignature {/byte}	Read disk signature from MBR
MbrFix /drive <num> generatesignature	Generate disk signature in MBR
MbrFix /drive <num> readstate	Read state from byte 0x1b0 in MBR
MbrFix /drive <num> writestate <state>	Write state to byte 0x1b0 in MBR

Drive numbering <num> starts on 0.

Commands restorembr, fixmbr, generatesignature, writestate and clean will ask for confirmation unless /yes is included.

If the /byte option is given for the readsignature command, the signature is returned as a byte array instead of as a DWORD.

## Return codes:

The program reports an errorlevel as 0 for OK and other values for errors.

The exception to this is the ReadState command which returns the state as a return code as well as printing to stdout. Any errors for this command are returned as negative values because of this.

## Why do I need this utility?

Well, I was testing Windows Embedded XP using a virtual PC, and was booting Windows PE from a CD-ROM to partition a brand new harddisk (quite cheap, since it was virtual !). Using the diskpart.exe utility (found in Windows XP and Windows Server 2003, available as a download for Windows 2000), I created the partition, made it Active, assigned it a drive letter. I then used the format.exe utility to format it to NTFS (format C: /fs:ntfs).

```
DISKPART> detail part
```

```
Partition 1
```

```
Type: 07
```

```
Hidden: No
```

```
Active: Yes
```

Volume ###	Letter	Lable	Fs	Type	Size	Status	Info.
* Volume 1	C	WinXP	NTFS	Partition	149 MB	OK	System

I then mounted a network drive to my physical computer and copied the boot.ini, ntldr, ntdetect.com, windows folder, program files folder, etc. from the Windows Embedded XP image directory. Everything to make it boot.

Well everything should be OK, so I rebooted the (virtual) computer and expected it to boot up Windows Embedded XP.

But not so! It just sat there with a nice black screen. No error message at all.

Some investigation later on the Internet, I found this very informative page by Daniel B. Sedory: [http://www.geocities.com/thestarman3/asm/mbr/MBR\\_in\\_detail.htm](http://www.geocities.com/thestarman3/asm/mbr/MBR_in_detail.htm), containing a wealth of MBR info. The info on one of the pages states that the MBR code is installed on the drive when the Windows setup program runs and no MBR code exists already. Hmm.., seemed to be right on the mark.

I searched further on the Internet, but the only utilities I found which could install MBR code was for other operating systems, like FDISK.exe /MBR for DOS (but I wouldn't boot to DOS...) or the fixmbr command in the Windows 2000/XP/2003 Recovery Console. Didn't want that either. Wanted something to work on a Windows 32 bit platform, to be able to fix the problem in other cases when using Win PE and just plain copying in the operating system files. So, then I was left to make the utility myself!

Problem solved: After booting up Windows PE from CD again, the command **MBRFix /drive 0 fixmbr /yes** solved it all. Embedded XP booted like a charm.

## Samples:

```
C:\> MbrFix /drive 0 listpartitions
```

```
# Boot Size (MB) Type
```

1	Yes	151001	6	DOS 3.31+ 16-bit FAT (over 32M)
2		1623	12	WIN95 OSR2 32-bit FAT, LBA-mapped
3		0	0	None

4                    0    0   None

```
C:\> MbrFix /drive 0 savembr Backup_MBR_0.bin
```

```
C:\> MbrFix /drive 0 restorembr Backup_MBR_0.bin
```

```
You are about to Restore MBR,  
are you sure (Y/N)? Y
```

```
C:\> MbrFix /drive 0 fixmbr /yes
```

## Readstate and Writestate:

The commands *readstate* and *writestate* use a byte at offset 0x1b0 to store state. The commands are only useful in a scenario where you use Windows PE to roll out other MS operating systems, and you *have no other place* to store information about where you are in the installation process. This location in the MBR is unused as far as known.

## Credits:

Daniel B. Sedory, for his compilation of information on [http://www.geocities.com/thestarman3/asm/mbr/MBR\\_in\\_detail.htm](http://www.geocities.com/thestarman3/asm/mbr/MBR_in_detail.htm) and for his feedback on the program.

## User License Agreement:

The program is delivered AS-IS, that is without any form of guarantee. You may freely use the program for any legal purpose, and you may freely distribute it as long as this file is distributed along the program file.

Incorrect use of this program **may** cause **loss of all data** on your disk-drives. All use of this program is at the risk of the user. Systemintegrasjon AS does not accept any liability for any damage the program might cause.

If you do not accept these terms, you must delete the program, **MbrFix.exe**, at once!

*Developed by K 鎔 e Smith, 29. november 2004*