

Zabawy z MBR - Instalacja Windows Server 2012 z Linuksa, na serwerze bez dostępu do BIOS'u, napędu CD ani USB

[Strona główna](#) > [@okokok](#) > Zabawy z MBR - Instalacja Windows Server 2012 z Linuksa, na serwerze bez dostępu do BIOS'u, napędu CD ani USB
13.12.2013 20:37

Kolega załapał się ostatnio na testy serwerów dedykowanych z nowym panele administracyjnym w homecloud.pl. Dzięki temu ja również miałem okazje zobaczyć ten panel i pobawić się serwerem :)

Są to serwery firmy Supermicro, ale panel jest wykonany przez home i wyszło im to świetnie! Home zawarło w nim kilka fajnych funkcji, takich jak uruchomienie serwera z obrazu ISO umieszczonego na zewnętrznym serwerze HTTP lub wybranego z listy preinstalowanych. Dostępny jest też bardzo fajny obraz rescue bazujący na Debianie, dzięki czemu, w razie awarii mamy pełny dostęp do dysku serwera i możemy naprawić znajdujący się na nim system.

Standardowy panel Supermicro też pozwala uruchomić serwer z obrazu ISO, jednak przekierowanie realizowane jest tylko i wyłącznie z komputera na którym uruchomiona jest konsola. W przypadku wolnego łącza, idzie to jak

kręć z nosa i powoduje (zbędne?) wkurzanie się na Jave w której napisana jest konsola.



Home rozwiązało to przy pomocy serwera PXE - serwer (klienta) podczas uruchamiania się odpytuje dostępne w sieci serwery DHCP o swoją konfigurację. Serwer DHCP w odpowiedzi przesyła adres serwera TFTP oraz nazwę pliku znajdującego się na tym serwerze. Dedyk (serwer klienta) pobiera z tego serwera (home) plik (w przypadku homecloud jest to gPXE) a następnie go uruchamia. gPXE to kolejny agent PXE, ale ma on więcej funkcji niż ten wgrany na stałe w ROM karty sieciowej - tutaj najważniejsza jest obsługa HTTP. Dalej już właśnie po HTTP pobierany jest bootloader który pozwala na start z ISO czyli GRUB, kolejno właściwy obraz ISO i dopiero uruchamiany jest system operacyjny.

Niestety z tym rozwiązaniem jest jeden problem - po zbootowaniu nie jest możliwy dostęp do CD-ROMU, a niektóre systemy tego wymagają. Problem nie występuje z obrazami typu netinstall, do korzystania z których namawia nas panel, jednak niektóre systemy - tutaj Windows Server nie posiadają obrazów tego typu i wysypują się zaraz po starcie :(

W tym wpisie przedstawię jak udało mi się na tym serwerze zainstalować Windows Server 2012 korzystając z Debiana 7 oraz GRUB'a. Dostęp do BIOS'u serwera był zablokowany na hasło. Podłączenie napędu CD, albo pendrive także nie wchodziło w grę bez interwencji kogoś w serwerowni.

1. Zainstalowałem na serwerze Debiana 7 korzystając z dostępnego w panelu instalatora. System został zainstalowany tylko na pierwszym dysku, bez konfiguracji RAID'u.



2. Po instalacji wyzerowałem MBR na drugim dysku:

```
dd if=/dev/zero of=/dev/sdb bs=512 count=1
```

3. Utworzyłem na drugim dysku 1 partycje o rozmiarze 1GB i następnie sformatowałem ją w systemie plików NTFS (pod Debianem konieczna jest do tego instalacja pakietu ntfs-3g).

```
apt-get update  
apt-get install ntfs-3g  
cfdisk /dev/sdb  
mkfs -t ntfs /dev/sdb1
```



4. Przymontowałem uprzednio pobrany obraz ISO Windows Server 2012 oraz partycje /dev/sdb1.

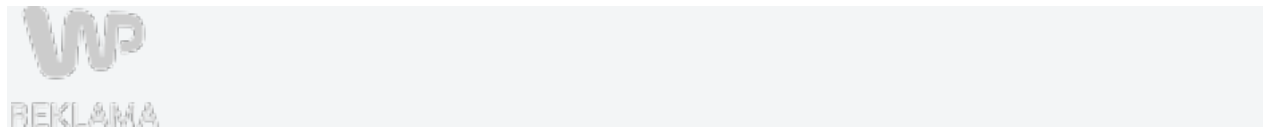
```
mkdir /mnt/cd /mnt/sdb1

mount -t udf /root/plyta.iso /mnt/cd

mount -r ntfs-3g /dev/sdb1 /mnt/sdb1
```

Płyty instalacyjne nowych Windowsów są sformatowane w nowszym formacie - UDF - ISO 13346, a nie typowym ISO 9960. Po przymontowaniu płyty w standardowy sposób (`mount -t iso9660`) zobaczymy tylko 1 plik z informacją nt. formatu płyty.

5. Przekopiowałem pliki z obrazu płyty na partycje sdb1:



```
cp -rv /mnt/cd/* /mnt/sdb1/
```

MBR to kod znajdujący się na samym początku dysku twardego, jeszcze przed partycjami, a dokładniej w pierwszym sektorze. Składa z 512 bajtów i określa rozmiar i pierwsze sektory 4 partycji podstawowych (tablica partycji znajdująca się na końcu) oraz zawiera kod rozruchowy (446 bajtów na samym początku). Kod rozruchowy wykonywany jest podczas startu komputera i to za jego pomocą ładowany jest system operacyjny.

VBR to kod znajdujący się na początku partycji. Również składa się z 512 bajtów. Pierwsza część NTFS VBR - 84 bajty, zawiera informacje specyficzne dla partycji - rozmiar sektora, ilość sektorów na klaster i ścieżkę, ilość głowic, sektorów, klastrów, identyfikator partycji itp. Po zmianie dokonanej w tej części NTFS VBR, partycja staje się niemożliwa do przywołania. W dalszej części znajduje się kod uruchamiający, podobny do tego z MBR.

Obecnie możliwe jest zawarcie kodu startowego w sektorze VBR partycji a następnie ustawienie dla tej partycji flagi boot (Windowsowe określenie to partycja aktywna), co spowoduje wykonywanie kodu z sektora VBR partycji podczas startu komputera. Wykorzystuje się to zamiast umieszczania kodu bezpośrednio w MBR.



Domyślnie po rekordzie VBR NTFS'a na partycji znajduje się jeszcze 14 wolnych sektorów. Właściwy system plików zaczyna się od 16 sektora. W tej wolnej części, jeśli partycja jest bootowalna, umieszcza się dalszą część kodu rozruchowego.

Tak właśnie jest z pendrive instalacyjnym Windowsa. Po skopiowaniu plików na pierwszą partycję pendrive z ustawioną flagą boot, uruchamia się program bootsect.exe znajdujący się na płycie, podając nazwę partycji na której ma być zapisany kod startowy. Na Linuksie nie jest to jednak możliwe, tak więc musiałem znaleźć inne wyjście.

Duuuuużo więcej informacji nt. MBR'u, VBR'u i Assemblera można znaleźć na tej stronie: <http://thestarman.pcministry.com/asm/mbr/index.html>. Tutaj znajduje się opis rekordu VBR dla partycji

NTFS: <http://thestarman.pcministry.com/asm/mbr/NTFSBR.htm>, a tutaj VBR wraz z kodem startowym dla Windows

Vista: <http://thestarman.pcministry.com/asm/mbr/VistaVBR.htm>.

6. Na wszelki wypadek wykonałem kopię zapasową rekordu MBR dla drugiego dysku oraz VBR wraz z 14 kolejnymi pustymi sektorami dla pierwszej partycji tego dysku do plików w katalogu domowym:

```
dd if=/dev/sdb of=/root/mbr.bin bs=512 count=1  
  
dd if=/dev/sdb1 of=/root/vbr.bin bs=512 count=15
```

7. Przy pomocy edytora heksadecymalnego dla Windowsa (HxD) wykonałem kopię sektora VBR oraz kolejnych sektorów w których znajdował się kod startowy z pendrive instalacyjnego Windowsa 8 do pliku i załadowałem go na serwer, a następnie nadpisałem tym plikiem sektor VBR i kolejne na partycji, ponownie nadpisałem konfigurację partycji znajdującą się w pierwszych 84 bajtach z kopii oryginalnego VBR (po poprzednim nadpisaniu została tam ta dla mojego pendrive) i sprawdziłem czy możliwe jest przymontowanie partycji:

```
dd if=/root/vbr_pendrive.bin of=/dev/sdb1 bs=512 count=15  
  
dd if=/root/vbr.bin of=/dev/sdb1 bs=84 count=15
```

```
mount -t ntfs-3g /dev/sdb1 /mnt/sdb1 -- zadziałało :)
```

8. Teraz gdyby serwer nie miał założonego hasła na BIOS i ekran wyboru urządzeń (nie wiem jak to się dokładnie nazywa, chodzi mi o opcje boot device w trakcie startu komputera), możliwy byłby start komputera z tego dysku i instalacja systemu.

Musiałem więc dopisać do pliku konfiguracyjnego GRUB'a - Linuksowego bootloadera, odpowiednią opcje, tak aby mógł on uruchomić system z drugiego dysku.

Konfig GRUB'a w Debianie 7 znajduje się w `/boot/grub/grub.cfg`, jednak twórcy dystrybucji nie zalecają jego edycji i proponują dodawać nowe opcje do jakiegoś folderu w `/etc` a następnie składać to wszystkiego w `grub.cfg` przy pomocy jakiegoś polecenia. That's the proper way, ale ja po prostu dopisałem poniższy fragment do pliku, w sekcji `40_custom`:

```
menuentry 'Instalator Windowsa' {  
    set root='(hd1,msdos1)'  
    chainloader +1  
}
```

Dyski (w drugiej linijce) numerowane są od 0, więc `hd1` to drugi dysk twardy, a `hd0` to ten pierwszy na którym postawiłem Debiana. Partycje numerowane są już od 1, więc chodzi tutaj o pierwszą partycje drugiego dysku.

Po aktualizacji kernela pod Debianem, przestanie to, co prawda działać, ale było to i tak chwilowe rozwiązanie.

Pomocna była jeszcze zmiana czasu po jakim automatycznie wybierana jest pierwsza pozycja w menu (czyli Debian) i ładowany jest ten system. Domyślnie to 5 sekund. Odpowiada za to opcja:

```
set timeout=50 -- tutaj zamieniłem na 50 sekund
```

9. Po restarcie, na liście systemów w GRUB'ie znajdował się już odpowiedni wpis, ale nie mogłem się do niego dostać strzałkami. Serwery Supermicro mają z tym jakiś problem. Wygooglałem rozwiązanie polegające na zmianie ustawień konsoli w panelu Supermicro do którego nie miałem dostępu. Pomogło wywołanie z menu konsoli jViewer (której niestety home nie udało się unicestwić bo jest za bardzo powiązana ze sprzętem) polecenia: Options -> Keyboard Mouse Hotplug.

10. Dalej było już prosto - Windows zainstalował się jak z płyty.

(tak wiem że jest po polsku a tłumaczenie jest brzydkie, ale kolega wrzucił na serwer właśnie ten obraz)

11. Właśnie wtedy dotarło do mnie że wystarczyło nadpisać cały 2 dysk obrazem ISO:

```
dd if=/root/plyta.iso of=/dev/sdb
```

Jednak robiąc partycje instalacyjną prawidłowym sposobem dowiedziałem się czegoś na temat MBR i VBR więc i tak wrzucam ten wpis.

Do doebugowania bardzo pomocne było polecenie `hd`, dostępne domyślnie w Debianie. `hd` czyli hexdump wyświetla pliki i strumienie binarne w postaci heksadecymalnej i ASCII, podobnie jak typowe windowsowe hexedytory (choć to pewnie jest odwrotnie i `hd` był pierwszy).

Aby wyświetlić rekord MBR zapisany w pliku, a właściwie to jakikolwiek plik binarny w programie `hd`, wpisujemy:

```
hd /root/mbr_sdb.bin
```

A, żeby odczytać MBR bezpośrednio z dysku:

```
dd if=/dev/sdb bs=512 count=1 | hd
```