Flashing BIOS from Linux

From ArchWiki

This article aims on providing information on flashing your system BIOS under Linux. Most manufacturers provide a Windows executable or a BIOS executable that can only be run under Windows. However, there are a few utilities, that allow you to upgrade your system BIOS under Linux.

Warning: Flashing motherboard BIOS is a dangerous activity that can render your motherboard inoperable! While the author of this article has successfully run this procedure many times, your mileage may vary. Be careful! You may want to consider updating microcode instead if it is supported by your system.

Contents

BiosDisk

BiosDisk (http://linux.dell.com/git/biosdisk.git) BiosDisk

simplifies the process of flashing your system BIOS under Linux

Installation

Install the `biosdisk-git (https://aur.archlinux.org/packages/biosdisk-git/)`[AUR] package.

Usage

To use the biosdisk utility to create a BIOS flash image, first download the latest raw BIOS image for your system from your manufacturer's website. Make sure however, that you always get the BIOS executable and NOT the Windows executable. You then have one of several options: create a floppy, create a dd floppy image, create a user-installable distribution-specific package (e.g. RPM), or actually install the image for your bootloader.

- The mkfloppy action will create the biosdisk image and write it directly to a floppy disk. Usage is the following:

```
biosdisk mkfloppy [-o option] [-d device] [-k baseimage] /path/to/.exe
```

- The mkimage action will create a floppy image on the user's hard drive. Usage is the following:

```
biosdisk mkimage [-o option] [-i destination] [-k baseimage] /path/to/.exe
```

- The mkpkg action will create the floppy image, and use it to create a user-installable package specific to the distribution (example: RPM). When the package is installed, it will use the distribution's built-in tools to update the system's bootloader so that the user can boot to the image from the hard drive to flash the BIOS, without needing a floppy drive. Currently only Red Hat/Fedora RPM packages are supported. Usage is as follows:

```
biosdisk mkpkg [-o option] [--install] [--distro=] [--name=] [--version=] [--rele
```

- The install action will create the biosdisk image, copy the image file to /boot, and then update the bootloader with an entry for the image. Then all the user has to do is boot the system and select the image to flash the BIOS; this will load the biosdisk image directly from the hard drive and flash the BIOS.

```
biosdisk install [-o option] [--name=] /path/to/{.exe | .img}
```

Flashrom

Flashrom (http://www.flashrom.org/Flashrom)is a utility for identifying,

reading, writing, verifying and erasing flash chips. It is designed to flash BIOS/EFI/coreboot/firmware/optionROM images on mainboards, network/graphics/storage controller cards, and various programmer devices.

Warning: If you have a laptop/notebook/netbook, please do NOT try flashrom because interactions with the EC on these machines might crash your machine during flashing. flashrom tries to detect if a machine is a laptop, but not all laptops follow the standard, so this is not 100% reliable.[1] (https://www.flashrom.org/Board_Testing_HOWTO)

Installation

Install the `flashrom (https://www.archlinux.org/packages/?name=flashrom)` or `flashrom-svn (https://aur.archlinux.org/packages/flashrom-svn/)`[AUR] package.

Usage

Find out if your motherboard and chipset (internal) is supported by flashrom at this website. Supported Hardware (http://www.flashrom.org/Supported_hardware) You can also find out if your hardware is supported by issuing the following command

```
# flashrom --programmer internal
```

The above command will tell you your motherboard and chipset. You can then find out if yours is supported by issuing this command:

```
# flashrom --programmer internal -L | grep CHIPNAMEfrompreviouscommand
```

On modern mainboards you probably get more than one rom chip listed. You have to select the chipname you get from the upper command. Then you use the `-c` option to select which rom is affected by the command

```
# flashrom --programmer internal -c "CHIPNAME" -r backup_CHIPNAME.bin
```

Write and verify the new BIOS image (proprietary or Coreboot) on the ROM chip:

```
# flashrom --programmer internal internal -c "CHIPNAME" -w newbios.bin
```

If you want to flash other flash chips on your mainboard, you will find all options with

```
# flashrom
```

Note: With Linux kernel versions greater than 4.4, CONFIG_IO_STRICT_DEVMEM a new kernel security measure can make flashrom stop working, in that case you can try adding "iomem=relaxed" to your kernel parameters. [2] (https://www.flashrom.org/FAQ).

FreeDOS

FreeDOS (http://www.freedos.org/) a free DOS-compatible operating system, is up to the challenge, no need for proprietary DOS versions. So, all you need is a bootable floppy disk image with FreeDOS kernel on it.

Unetbootin

By far the easiest way to make a bootable FreeDOS USB Stick is using `unetbootin (https://www.archlinux.org/packages/?name=unetbootin)`, available in the Official repositories.

You should format a pendrive with FAT16 and flag it as "boot" (you may do this through a GUI with `gparted (https://www.archlinux.org/packages/?name=gparted)`, `qtparted (https://aur.archlinux.org/packages/qtparted/)`[AUR][broken link: archived in aur-mirror (https://github.com/felixonmars/aur3-mirror/tree/master/qtparted)] or `partitionmanager (https://www.archlinux.org/packages/?name=partitionmanager)`). Then, after mounting the flash drive, select under distribution **FreeDOS** and your mounted stick. The app will automatically download the image for you and copy it to the drive. Finally, you may copy everything you want to flash there (BIOS, firmwares, etc).

Warning: Unetbootin may not function properly on some Lenovo systems. It may be necessary to create the bootable stick on a different device. See here (http://reboot.pro/topic/9849-blinking-cursor-at-boot/).

Gentoo

Check out FreeDOS Flash Drive (https://wiki.gentoo.org/wiki/BIOS_Update#FreeDOS_environment) on the Gentoo Wiki if you want to create a bootable FreeDOS Flash drive.

Prebuilt images

Yet another simple solution: FreeDOS prebuilt bootable USB flash drive image by Christian Taube (http://chtaube.eu/computers/freedos

/bootable-usb/)

Images that are too large for a floppy

If your flash image is too large for a floppy, go to the FreeDos bootdisk website (http://www.fdos.org/bootdisks/), and download the 10Mb hard-disk image. This image is a full disk image, including partitions, so adding your flash utility will be a little trickier:

First find the first partition (at time of writing, the first partition starts at block 63; this means that the partitions starts at offset `512 * 63 = 32256`). You can either use:

```
# file -sk <image-file> | sed -r 's/.*startsector ([0-9]+).*/\1/'
63
```

Or:

```
# fdisk -l <image-file>
…
Units = sectors of 1 * 512 = 512 bytes
…
      Device  Boot  Start    End  Blocks  Id  System
              *        63  19151   9544+   1  FAT12
```

Now you can mount the image:

```
# mount -oloop,offset=$((63 * 512)) <image-file> /mnt
```

Then you can copy your flash utility onto the filesystem as normal. Once you're done:

```
# umount /mnt
```

The image can now be copied to a USB stick for booting, or booted as a memdisk as per normal instructions.

Usage

The OEM Bootdisk version is recommended, as it only includes `kernel` and `command.com` thus leaving more space for the flash utility and new BIOS image. Download and decompress the FreeDOS image:

```
$ wget http://www.fdos.org/bootdisks/autogen/FDOEM.144.gz
$ gunzip FDOEM.144.gz
```

Copy your BIOS flash utility and new BIOS image to the mounted floppy disk image. Load the necessary modules:

```
# modprobe vfat
```

```
# modprobe loop
```

`/proc/fileystems` shows if the needed file systems are supported. "loop mount" the floppy disk image to a temporary path:

```
$ mkdir /tmp/floppy
$ mount -t vfat -o loop FDOEM.144 /tmp/floppy
```

If the mount went without errors, copy the BIOS flash utility and new BIOS image to the mounted floppy disk image. You will probably have to unzip the archive you downloaded from your motherboard vendor site, to get to those two files. For example:

```
# unzip 775Dual-VSTA\(2.60\).zip
Archive: 775Dual-VSTA(2.60).zip
 inflating: 75DVSTA2.60
 inflating: ASRflash.exe
# cp 75DVSTA2.60 ASRflash.exe /tmp/floppy
```

Check that the two files were not too big for the floppy:

```
Filesystem           1K-blocks      Used Available Use% Mounted on
/tmp/FDOEM.144
                         1424       990       434  70% /tmp/floppy
```

Unmount the floppy disk image:

```
umount /tmp/floppy
```

The next step is to burn the floppy image to a CD/DVD-RW media, but in a way that it can be booted afterwards. First create a bootable CD image, and then burn it.

```
genisoimage -o bootcd.iso -b FDOEM.144 FDOEM.144
wodim -v bootcd.iso
```

You may alternatively add your image to the GRUB menu. Install syslinux and copy `memdisk` and your image to `/boot`:

```
cp /usr/lib/syslinux/memdisk /boot
cp FDOEM.144 /boot/flashbios.img
```

Now add an entry to `/boot/grub/menu.lst`:

```
title Flash BIOS
kernel /memdisk
initrd /flashbios.img
```

Or for GRUB2 in `/boot/grub/grub.cfg`:

```
menuentry "Flash BIOS" {
linux16 /boot/memdisk
initrd16 /boot/flashbios.img
}
```

Or for syslinux in `/boot/syslinux/syslinux.cfg` :

```
LABEL flashbios
        MENU LABEL Flash BIOS
        LINUX ../memdisk
        INITRD ../fdboot.img
```

Finally reboot your machine, making sure the CD drive is first in the boot sequence, and run the BIOS upgrade procedure when the CD boots. If using the GRUB method, choose the new entry on the list, and it should boot into FreeDOS.

Bootable optical disk emulation

The script Geteltorito.pl will extract the El Torito boot image. It has worked on Lenovo laptops like X220, X230, W540 and T450s. It may work for other vendors as well.

Installation

Install the `geteltorito (https://aur.archlinux.org/packages/geteltorito/)`[AUR] package.

Usage

Get the bios update iso from the vendor support site. Run the *geteltorito* image extraction:

```
$ geteltorito.pl -o <image>.img <image>.iso
```

Copy the image to the usb thumbdrive:

```
# dd if=<image>.img of=<destination> bs=512K
```

Reboot and boot from the USB drive, follow vendor directions.

Note: If you get the message "Secure Flash Authentication failed!", it means that some security check did not allow the flash to happen. It can help to go to the BIOS options page "Security" > "UEFI BIOS Update Option" and disable "Secure RollBack Prevention" and enable "Flash BIOS Updating by End-Users". You can set them to what you want after flashing.

fwupd

fwupd is a simple daemon to allow session software to update device firmware on your local machine. It's designed for desktops, but this project is also usable on phones, tablets and on headless servers. You can either use a GUI software manager like GNOME Software to view and apply updates, the command-line tool or the system D-Bus interface directly.

Large vendors including Dell will use this way to distribute firmware updates to Linux.

A list of hardware currently available in stable tree can be viewed [[3] (https://secure-lvfs.rhcloud.com/lvfs/devicelist%7Chere)].