

Convert FreeBSD ISO image for PXE bootstrap

Usually ISO's of most Linux distros use isolinux as boot loader, so these may be served up easily for pxe bootstrap through pxelinux/memdisk.

For FreeBSD this does not hold true, so the vanilla ISO's must be converted before they can be bootstrapped.

The basic idea is to convert the FreeBSD ISO into a harddisk image to circumvent a limitation in the ISO boot loader.

#1 Preparation

We need some staging directory where we can safely play around with the files.

```
#mkdir /tmp/build
#mkdir /tmp/build/iso.mnt
#mkdir /tmp/build/hd.mnt
#mkdir /tmp/build/mfs.mnt
```

#2 Get the ISO image

I usually use the 'bootonly' image because of it's size (~25 MB). It would work with the full ISO's of course, however will not only take longer to download via tftp but also require more memory on the client host.

```
#wget ftp://ftp.freebsd.org/pub/FreeBSD/ISO-IMAGES-i386/6.2/6.2-RELEASE-i386-bootonly.iso
--output-document=/tmp/build/6.2-RELEASE-i386-bootonly.iso
```

Now we need to attach 'md' (memory device) to the ISO image and mount it.

```
#mdconfig -a -t vnode -f /tmp/build/6.2-RELEASE-i386-bootonly.iso
md0
#mount_cd9660 /dev/md0 /tmp/build/iso.mnt/
```

#3 Create a new harddisk image

Let's create a new harddisk image of 32 MB in size. It make it a bit bigger than the original ISO so I could include further scripts with it.

```
#dd if=/dev/zero of=/tmp/build/6.2-RELEASE-i386-bootonly.hd bs=1m count=32
32+0 records in
32+0 records out
33554432 bytes transferred in 2.078232 secs (16145664 bytes/sec)
```

Then we attach a 'md' device to this one, too. Afterwards partition table and bsdlable are initialized.

```
#mdconfig -a -t vnode -f /tmp/build/6.2-RELEASE-i386-bootonly.hd
md1
#fdisk -B -I /dev/md1
#bsdlable -B -w /dev/md1
```

Now we would need to edit the bsdlable to add the 'a' slice of type 4.2BSD covering the whole harddisk image.

```
#bsdlable -e /dev/md1
# /dev/md1:
8 partitions:
#   size  offset  fstype  [fsize bsize bps/cpg]
a: 65520   16   4.2BSD
c: 65536    0  unused    0   0    # "raw" part, don't edit
```

Finally, the slaved will be newfs'ed with UFS and mounted.

```
#newfs /dev/md1a
#mount /dev/md1a /tmp/build/hd.mnt
```

Now the 'boot' directory must be copied from the mounted ISO image onto the harddisk image.

```
#copy -r /tmp/build/iso.mnt/boot /tmp/build/hd.mnt/boot
```

#4 Optional: Prepare SysInstall for automatic installations

This step is completely optional, though will allow us to place an 'install.cfg' file to the image. SysInstall will use this file as configuration template for automatic installations. A sample of it is found within the source tree at /usr/src/usr.sbin/sysinstall/install.cfg. Optionally you can use mine as a starting point.

First we need to copy away mfsroot.gz into our build directory. It will then be extracted, attached to a vnode and mounted.

```
#cp /tmp/build/hd.mnt/boot/mfsroot.gz /tmp/build/hd.mnt
#gunzip /tmp/build/hd.mnt/mfsroot.gz
#mdconfig -a -t vnode -f /tmp/build/hd.mnt/mfsroot
md2
#mount /dev/md2 /tmp/build/mfs.mnt
```

Then add the install.cfg file to the mfsroot top level directory.

```
#cp /tmp/build/install.cfg /tmp/build/mfs.mnt
```

You are free to add further customizations, like self-built packages, to the mfsroot.

When finished unmount the mfsroot device, detach it from the 'md' device, compress and copy it back to the harddisk image.

```
#umount /tmp/build/mfs.mnt
#mdconfig -d -u 2
#gzip /tmp/build/mfsroot
#cp /tmp/build/mfsroot.gz /tmp/build/hd.mnt/boot
```

#5 Finalize

So let's clean up little.

```
#umount /tmp/build/hd.mnt
#mdconfig -d -u 1
#umount /tmp/build/iso.mnt
#mdconfig -d -u 0
```

Let's shift the harddisk image to the tftpd boot image directory, which is /var/tftpd/images for me.

```
#cp /tmp/build/6.2-RELEASE-i386-bootonly.hd /var/tftpd/images
```

Finally pxelinux must be instructed on how to boot this image.

The assistance of memdisk is required to achieve this by adding the lines below to pxelinux.cfg/default configuration file.

```
label fbsd62
kernel memdisk
append initrd=/images/6.2-RELEASE-i386-bootonly.hd harddisk
```

So from now on requesting 'fbsd62' at the pxelinux boot prompt should boot the image.

It happend to me occasionally that the FreeBSD boot loader crashed on some systems. So far this seems depending on the BIOS, because upgrading to the latest releases fixed it usually. I suspect that some BIOS versions mess up if a PXE bootstrapped image reports itself as beeing a harddisk.

It did not yet find a solution to this particular issue except using old-fashioned 1.44/2.88m floppy disk images for PXE bootstrap.

#6 Boot Environment and Configuration Files

This howto actually covers a topic which came up during development of a network installation environment.

The host acting as boot server is running FreeBSD with ISC dhcpd, stock tftpd and pxelinux serving as bootstrap loader.

I'm providing you with some basic configuration files to round up the picture.

My live-setup is in fact a bit more sophisticated than that and uses some additional scripting magic do achieve real automated setups not only for FreeBSD but also for various Linux distros. This however I am not allowed to publish in detail due to company policy.

[dhcpd.conf](#)

[inetd.conf](#)

[pxelinux.cfg](#)

[install.cfg](#)