

Chaining FreeBSD's pxeboot with pxelinux

Recently I invested some development time on my company's PXE-based network boot system. While pxelinux serves as a general purpose network boot loader at our site, current demands required further extension beyond it's capabilities. The main reason for this was the inability of pxelinux to be used for certain bootstrap scenarios. As an example to this we may note FreeBSD. While it can be booted from floppy images or an [hd-converted ISO-image](#) via pxelinux's memdisk loader, this actually has some serious limitations.

- You're always limited to the size of the floppy image
- Converting ISO's to hd-like images tends to crash on some buggy BIOS versions

So I decided to conquer FreeBSD to boot directly from pxe. Because I had to retain compatibility with pxelinux as our primary PXE bootloader, FreeBSD's own loader had to be "chained" to pxelinux. Now this is the easy part as you will only add your pxelinux **default** file like this:

```
label fbsdpxe
    KERNEL pxeboot
```

You see the error!? Well, I didn't either at first and stumbled accross an ever lasting error message while trying to boot.

```
-----
NetInstall :: Main Menu
-----
```

```
F1 :: F2 :: F3 :: F4 :: F5 :: F6 :: F9 :: exit
-----
```

local boot is default after 10 seconds.

Press F1 - F10 to cycle menu pages, enter "exit" or option name to

boot: fbsdpxe

Loading

Invalid or corrupt kernel image

boot:

At first I though my pxeboot image got corrupted so I fetched another copy still receiving the same error.

However I saw the tftp download request in the logs so the config by itself could not be in error.

```
Jul 19 20:21:39 setup tftpd[22265]: 192.168.2.239: read request for /pxelinux.cfg/default: success
```

```
Jul 19 20:21:39 setup tftpd[22267]: 192.168.2.239: read request for /pxelinux.cfg/screens/f1.txt: success
```

```
Jul 19 20:21:48 setup tftpd[22272]: 192.168.2.239: read request for /pxeboot: success
```

So what had happened?

When copying the FreeBSD pxeboot loader over to the tftpd boot directory I must have forgotten to type the filename correctly, so **pxeboot.0** suddenly became **pxeboot**.

Without noticing this I added it like above to the pxelinux config file and ended up with the exact error message as previously shown.

After changing the filename to read **pxeboot.0** it actually chain-booted via pxelinux.

During further examination I made the same mistake once before when I iniatlly configured pxelinux years ago, omitting the **.0** by

the end of the filename.

Interestingly enough pxelinux was served without any problems and could be booted successfully despite it's possibly wrong filename.

Even more amazing was the fact that also the FreeBSD pxeboot loader would work if served up as primary loader as specified from dhcpd.conf using the "wrong" filename.

Now I'm not sure if there's a real reason or a naming convention which actually defines the boot loader's filename to end in **.0**. The [official specs](#) mention **REMOTE.0** to be used as NBP (Network Boot Program) and an optional **REMOTE.1** to be fetched additionally in case the NBP exceeds 32k in size and needs to be splitted.

However there seems to be no real, fixed naming convention so to my understanding the file could actually be called anything.

So the mystical **.0** at the end of the filename might have been chosen initially to comply with the original specs and show that the file in question is the first part of the NBP (even if a second part would not exist).

Finally remains the question if pxelinux's behaviour is by error or by design.

The true answer to this is: by design.

H. Peter Anvin explains this on the [common problems](#) page as follows:

[..]It is unfortunate that there isn't a standard extension used for Linux kernels, and that none of the commonly loaded data formats (except perhaps COM32) have reliable magic numbers.[..]

To conclude from this and the information on reserved filename extensions pxelinux loader routines will decide what to do with a given kernel only upon its filename extension.