

PXE BOOTING LINUX

A Talk By

Fish

(IT'S NOT THAT
SCARY, HONEST)

For boot options, press F1, or press F2 for the presentation!

boot: _

What Is PXE?

- o PXE is Pre-eXecution-Environment
- o Developed by Intel
- o Allows machines to boot from a network
- o BIOS needs to be able to talk to the network card, and so the network card, needs a particular ROM
- o Uses combination of DHCP and TFTP to work.
- o Fairly ubiquitous on modern hardware (BIOS and NIC ROMs)
however EtherBoot is available for non-supporting hardware

..... F3

boot: _

What use is it?

- o OS installation
 - o Machines with no optical drives
 - o Automated installation on many machines
- o Diagnostics and Repair
 - o Disk Partitioning
 - o Hardware testing (memtest86+ and friends)
 - o Remote diagnostics
 - o BIOS/firmware patch deployment
- o Thin clients
 - o Minimal Linux installation with XDMCP
- o Diskless clients
 - o Kernel booted via PXE, root filesystem on a NFS share!
 - o Just uhh, don't try to use filesystem swap files >_<
 - o Not-particularly-thin-but-not-really-fat-either clients

F2 ~~~~~ F4

boot: _

How does it work?

- o Client boots and loads the Intel Bootloader ROM on the network card
- o Makes a request for DHCP, and retrieves
 - o IP Address, Subnet Mask, Default Gateway
 - o TFTP server (or next-server)
 - o PXE Bootloader filename to retrieve from the TFTP server
- o The PXE firmware will download the file from the TFTP server and execute it
- o The bootloader is given access to the TFTP server and can load more data from it (e.g. help info, images, kernels and ramdisks)
- o The bootloader does not actually have network access (PXE ROM provides abstracted access to the TFTP server)
- o Anything loaded from the TFTP server is put into system memory whilst the bootloader is running!

F3 ~~~~~ F5

boot: _

Setting up the network

- o You will need:
 - o DHCP server that you can actually configure
 - o TFTP server (atftp works okay)
- 1. Configure the DHCP server
 - o Make a block for your PXE clients (e.g. recognise by MAC)
 - o Make sure it's allowing bootp and booting
 - o Send extra parameters to your PXE clients
 - next-server 192.0.0.1 (your TFTP server)
 - filename "pxelinux.0"
- 2. Set up TFTP server
 - o Make your root directory (e.g. /tftpboot/)
 - o Point the server at it (the last part of the tftp line in inetd.conf)
 - o Put your PXE bootable files in there

F4 ~~~~~ F6

boot: _

Configuring PXELINUX

- o Copy your kernel image (typically called linux or vmlinuz) into your tftpboot directory, as well as any initrd.gz file that came with it
- o Grab the SYSLINUX package however you feel like it
- o Configure, compile if required (the only dependency I know of is nasm !)
- o Copy the pxelinux.0 image to your /tftpboot directory
- o Make a directory called pxelinux.cfg/ containing a file called default
- o Make the config (as per SYSLINUX configuration, RTFM for that)

e.g.

```
DEFAULT linux
LABEL linux
  KERNEL linux
  APPEND initrd=initrd.gz
```

F5 ~~~~~ F7

boot: _

Generic Example

- o This example allows PXELINUX (or SYSLINUX) to boot a generic linux kernel

```
DEFAULT linux          # the default label to choose if you hit enter
PROMPT 1               # whether to give a "boot:" prompt
TIMEOUT 0              # how many seconds to wait before choosing
                       # the default option

LABEL linux
  KERNEL linux         # the location of your kernel image
                       # (relative to to /tftpboot
  APPEND initrd=initrd.gz # kernel options here (initrd is quite
                       # an important one ;) )
```

- o This format only works for linuxy sorts of kernels
- o initrd.gz can be as big as you like, within reason. Remeber it has to fit into RAM when uncompressed!

```
F6 ~~~~~ F8
boot: _
```

Bootable Floppies

- o Floppy images need a bit of help to get booted from PXELINUX
- o Enter MEMDISK (comes with SYSLINUX) - copy the image into /tftpboot with the floppy img and robert's your mothers brother.

```
DEFAULT freedos
PROMPT 1
TIMEOUT 0
```

```
LABEL freedos
  KERNEL memdisk          # just specify memdisk at the kernel image
  APPEND initrd=freedos.img # and pass the bootable floppy image as the
                           # initrd parameter
```

- o Very handy for network booting BIOS flash utilities or other tools that don't use linux kernels.

```
F7 ~~~~~ F9
```

```
boot: _
```

Debian/Ubuntu Installer

- o You can point your TFTP server at `/cdrom/install/netboot/` , but it's easy enough to intergrate with your own boot menu so lets do that

- o `cp -R /cdrom/install/netboot/debian-installer/i386/ \`
`/tftpboot/debian-installer/`

- o Modify your pxelinux config:

```
LABEL debian-installer
  KERNEL debian-installer/linux
  APPEND vga=normal initrd=debian-installer/initrd.gz --
```

- o Now, when you type `debian-installer` at the boot: prompt, the installer runs!

- o Exactly the same for ubuntu - replace the word "debian" with ubuntu above

F8 ~~~~~ F10

boot: _

LiveCDs, thin clients and clusters

- o LiveCDs make excellent diskless systems - except there's a bit of an issue. You can't download an entire ISO into RAM with TFTP.
- o Minimise your install, and pack all of the files into initrd.
- o Example: Damn Small Linux has a 50MB miniroot.gz file, which contains the entire OS and programs.
- o Alternatively: Make a diskless install and mount the root filesystem over an NFS share, using nfsroot methods. This can be read only OR read write (just make sure only one client is using it at a time!)
- o Clusters can also be made in this manner - ClusterKnoppix
- o Thin clients can also be created - build a basic initrd that runs X, and run XDMCP to a central server.

F9

boot: _

F1

Fun Facts!

- o This presentation was brought to you entirely by PXELINUX
- o No operating system was loaded to show the slides
- o This presentation will run on any machine capable of PXE boot, even over a serial terminal (without graphics obviously)
- o There are nowhere near enough F keys

```
#####          ###
# # # ## # # # #### ###
# # # # ## # # # ##
# ##### # # # # ####  ##
# # # ##### # # # # #
# # # # # ## # # # # ##
# # # # # # # # # # ##
```

- o Oh, and by the way - this is still a fully functional bootloader:

Available boot options:

```
dsl          - Damn Small Linux Live
freedos      - FreeDOS utility boot floppy
memtest      - Memtest86+
debian-installer - Debian Network Installer
```

boot: _