Harnessing PXE Boot Services

in Linux Environments

The Preboot Execution Environment (PXE) is part of the Wired for Management (WfM) specification developed by Intel and Microsoft. This article describes the PXE process and explains how to set up a PXE environment on servers that run the Linux® operating system.

BY JOHN HULL, ROBERT HENTOSH, AND ROGELIO NORIEGA

Most computers sold today are compliant with the Wired for Management (WfM) specification. Initially created by Intel, WfM later became part of the Intel® and Microsoft® PC 98 specification, which defines minimum system requirements for PCs that run the Microsoft Windows® operating system (OS). The Preboot Execution Environment (PXE) is part of the PC 98 specification; it permits the system's BIOS and network interface card (NIC) to bootstrap a computer from the network. *Bootstrapping* is the process by which a system loads the OS into local memory so that it can be executed by the processor.

During a typical system startup, a boot program is loaded and executed before the much larger and more complex OS is loaded. The boot program is usually located on the first few blocks of the local hard disk. After the power-on self-test (POST), the system's BIOS routines load the boot program from the boot drive into memory. The boot program then reads the OS kernel from the hard disk into memory using BIOS calls.

PXE extends system startup functionality to allow the network to act as the medium from which the OS kernel can be loaded. This capability can simplify server deployment and maintenance for network administrators by allowing them to perform the following functions:

- Temporarily boot a server in a diagnostic environment, in which diagnostic tools can be downloaded from a remote server onto a local RAM disk
- Perform a network installation of the OS onto a server's local hard drive
- Boot a server into a network-mounted OS where no local hard disks are needed
- Boot a server into an OS downloaded from the network so that data can be recovered from a hard drive that fails to boot

This article explains the basic functionality of PXE as well as how PXE can be used in enterprise environments that run the Linux® OS.

Understanding how PXE works

When a system is powered on, the BIOS stored in the system's ROM goes through a POST. The POST checks

memory and processors, and sets up the controllers and bus bridges that enable the system to operate. The BIOS will then load the boot program, which in turn loads the OS kernel. Using PXE, instead of loading the boot program from the hard drive the BIOS uses Dynamic Host Configuration Protocol (DHCP) to obtain an IP address for the network interface and to locate the server that stores the network bootstrap program (NBP). Like the boot program (or boot loader) in the hard drive environment, the NBP is responsible for loading the OS kernel into memory so that the OS can be bootstrapped.

The PXE specification provides IT administrators with a powerful industry standard that can be used to load an OS onto a system across the network.

In a PXE environment, the first requirement is to obtain an IP address for the network card, which is accomplished using a DHCP discover message. The message is broadcast on the local subnet, a DHCP server on the local subnet responds with a DHCP offer, and the DHCP server then assigns the IP address to the interface. The client responds to the server to acknowledge that it has received the packet and is using the IP address.

This exchange also includes information from the DHCP server about the IP address and file name of the NBP; the client uses that information to download the NBP over the network using Trivial FTP (TFTP). The client then executes the NBP, which helps load the OS kernel. The NBP is equivalent to GRUB (GRand Unified Bootloader) or LILO (LInux LOader)—loaders typically used in local booting. The NBP does not need to reside on the same physical system as the DHCP service.

Configuring a PXE server for a Linux environment

The Red Hat® Enterprise Linux 3 operating system (versions AS and ES) provide the software required to configure and manage a PXE server. These capabilities are provided through four packages: dhcp, tftp-server, syslinux, and redhat-config-netboot. The dhcp package provides network configuration information to clients, including redirecting clients to boot servers as requested. The tftp-server package provides capabilities that allow a client to download the NBP from the boot server. The syslinux package provides the NBP (pxelinux.0) that clients download through the TFTP server. And finally, redhat-config-netboot provides a graphical user interface (GUI) as well as a command-line interface (CLI), both of which can be used to help simplify the configuration of installation options or diskless boot images.

Four primary steps are required to get a PXE server up and running. First, the TFTP server must be enabled and configured to allow downloading of the NBP. Second, the DHCP server must be configured to send clients the address of the TFTP server and the location of the NBP on that server. Third, images for network installation or diskless boot must be created on the PXE server. Finally, individual client systems must be enabled to boot using PXE.

Step one: Configuring the TFTP server

When configuring the TFTP server, first verify that the pxelinux.0 NBP is present in the /tftpboot/linux-install directory. If it is not, copy it to that directory from /usr/lib/syslinux. Next, verify that the pxelinux.0 file is readable by everyone by entering `ls -l /tftpboot/linux-install/pxelinux.0` at a command prompt and changing the permissions on the file if necessary. Finally, enable the tftp service by entering `chkconfig tftp on` at a command prompt.

Step two: Configuring the DHCP server

For a client system to download the NBP, the DHCP server must allow clients to boot from the network, and the DHCP must direct the clients to the NBP on the TFTP server. To do this, several options must be present in the /etc/dhcpd.conf file:

- `allow bootp`: Directs the DHCP server to respond to bootp queries
- `allow booting`: Directs the DHCP server to respond to queries from particular clients
- `next-server` *`TFTP server IP`*: Provides the IP address of the TFTP server that contains the NBP
- *`filename`* `/tftpboot/linux-install/pxelinux.0`: Points to the NBP on the TFTP server

Note that the DHCP server and the TFTP server do not have to reside on the same physical system. If they are on separate systems, the `next-server` option directs clients to the correct system. For a full description of how to configure a DHCP server, refer to "Chapter 25. Dynamic Host Configuration Protocol (DHCP)" in the *Red Hat Enterprise Linux 3: System Administration Guide* (https://www.redhat.com/docs/manuals/enterprise/RHEL-3-Manual/sysadmin-guide/ch-dhcp.html). For more information on PXE-specific DHCP options, refer to the SYSLINUX/PXELINUX home page at http://syslinux.zytor.com/pxe.php.

Step three: Creating images for network installation or diskless boot

The next step is to configure OS installation images or diskless boot images and to create configuration files on the PXE server. To configure OS installation images, Red Hat Enterprise Linux 3 provides two useful methods: the redhat-config-netboot GUI and the pxeos CLI. Both provide simple interfaces to set up client installation images quickly. For a thorough explanation of how to use these tools, refer to "Chapter 14. PXE Network Installations" in the *Red Hat Enterprise*

```
default 0
prompt 1
timeout 100
display msgs/boot.msg
F1 msgs/boot.msg
F2 msgs/general.msg
F3 msgs/expert.msg
F4 msgs/param.msg
F5 msgs/rescue.msg
F7 msgs/snake.msg


label 0
  localboot 1

label 1
  kernel /rhel3/vmlinuz
  append initrd=/rhel3/initrd.img ramdisk_size=10000
ks=nfs:192.168.0.1:/ks/ks.cfg
```

Figure 1. Sample pxelinux configuration file

Linux 3: System Administration Guide (https://www.redhat.com/docs/manuals/enterprise/RHEL-3-Manual/sysadmin-guide/ch-pxe.html).

Step four: Enabling clients to boot using PXE

The final step in preparing an environment for PXE installations is to enable the systems to boot using PXE. Current Dell™ PowerEdge™ servers, Dell Precision™ workstations, and Dell OptiPlex™ desktops have PXE capability built in to their integrated and add-in network interfaces. To enable the PXE booting capability, reboot the system and press the F2 key to enter the system setup menu. Select the Integrated Devices menu item and press Enter. Next, scroll down to the Network Interface Controller menu items and cycle though the available options until "Enabled with PXE" is selected. Save the changes and exit the menu.

To boot using PXE, administrators have several options. For PowerEdge servers, on the initial Dell splash screen during boot, administrators can use the Boot from PXE option by pressing the F12 key. On Dell Precision workstations and OptiPlex desktops, they can press F12 for the Boot Menu option, which presents a choice of boot devices including the integrated NIC. Alternatively, to boot from PXE at every reboot, administrators may alter the boot sequence in the Setup menu to move the "Integrated NIC" boot device earlier in the boot order.

Configuring the client for a PXE boot environment

When a client downloads the pxelinux.0 NBP from the TFTP server, the NBP searches the /tftpboot/linux-install/pxelinux.cfg directory for an appropriate configuration file that tells NBP what to do next. The search proceeds in the following sequence:

1. The NBP converts the client's IP address into an uppercase, hexadecimal string (for example, 10.9.0.1 would be 0A090001).
2. The NBP then looks for a file with that name in the pxelinux.cfg directory. If it does not find a file with that name, it will remove the final digit from the string and look again (using the previous example, it would search for 0A09000).
3. The NBP will continue this process until it finds the file (in the given example, searching for 0A0900, 0A090, 0A09, 0A0, 0A, and finally 0). If it does not find a matching file name, it will look for a file named "default."

This process can be helpful because it allows administrators to specify boot image options tailored to a specific client system or a group of systems using the IP address. The default file and the file named after the uppercase, hexadecimal string both contain the same types of information:

- OS images to boot
- Default OS image to load
- Menus that can be displayed to the client
- Options for displaying a prompt, including how long to display it

The configuration file: Customizing boot options for the client system

Figure 1 shows a sample configuration file that is similar to the grub.conf or lilo.conf file. Each section of the file that starts with `label` specifically points to an image that the client may boot. Choosing option `0` will cause the OS image on the local hard drive to boot. Choosing option `1` will load the kernel and the initrd file specified on the `kernel` and `append` lines, respectively. The paths to the kernel and the initrd file are relative to the location of the pxelinux.0 file (thus the path is actually /tftpboot/linux-install on the TFTP server). Note that as in the grub.conf and lilo.conf files, kernel parameters can be added to the `append` line. For example, to boot using a specific Red Hat kickstart file, administrators can use the `ks=` option. Other kernel parameters also can be used, such as `noapic`, `ide=nodma`, or any other desired kernel parameter.

The capability to load an OS from the network can simplify server deployment and maintenance for network administrators.

The `prompt 1` command directs the NBP to prompt the

user to choose a boot image, and to time out after the number of seconds specified by the timeout parameter. If the timeout is reached or the `prompt` parameter is not specified or set to 0, the boot image uses the default parameter.

The `display` parameter tells the NBP the location of a custom message to display to the user if the `prompt` parameter is specified. Once this message is displayed, the `F1` through `F7` parameters specify the messages that will be displayed subsequently if the user at the client system presses any of those keys. Note that, once again, the path to the message files is relative to the location of the pxelinux.0 file.

Current Dell PowerEdge servers, Dell Precision workstations, and Dell OptiPlex desktops have PXE capability built in to their integrated and add-in network interfaces.

Once the user receives the `boot` prompt at the client system, kernel parameters also can be specified at this prompt when choosing the desired label to boot. For example, the `rescue` parameter allows system administrators to enter rescue mode remotely to help speed system recovery. Other options, such as kickstart file location or specific kernel options, may be specified as well.

Enabling remote system administration across the network

The Preboot Execution Environment specification provides IT administrators with a powerful industry standard that can be used to load an OS onto a system across the network. This specification comes standard on NICs that are provided within current Dell servers, workstations, and desktops. Red Hat Enterprise Linux 3 provides software and configuration tools that help enable administrators to configure a PXE server. Combining Dell hardware with Red Hat Enterprise Linux 3 can help IT administrators to build a powerful enterprise data center that is designed to perform remote OS installations, OS booting on diskless systems, and remote recovery of failed systems.

John Hull (john_hull@dell.com) is a software engineer at Dell and is the lead Linux engineer for Dell Precision workstations. He has a B.S in Mechanical Engineering from the University of Pennsylvania and an M.S in Mechanical Engineering from the Massachusetts Institute of Technology (M.I.T.).

Robert Hentosh (robert_hentosh@dell.com) is a senior consultant on the Dell Linux Engineering team. Before joining Dell, he worked for 12 years in the software industry, with experience in device driver development, communications interfaces, and Web-based applications. Robert has a B.S. in Electrical Engineering and a B.A. in Physics, both from Bucknell University.

Rogelio Noriega (rogelio_noriega@dell.com) is a senior analyst on the Dell Linux Engineering team and a technician for Dell Precision workstations.

FOR MORE INFORMATION

Dell and Linux:
http://www.dell.com/linux