# 4.1.3 Linux Alternative Boot Options Facts

In addition to booting the Linux kernel from the computer's hard disk drive, you can boot Linux in a number of ways. These boot alternatives offer a number of benefits, including ease of administration and the ability to troubleshoot and repair a computer you cannot boot. Boot alternatives include:

- PXE
- NFS
- Boot from ISO
- Boot from HTTP

## PXE

The Preboot eXecution Environment (PXE) is the process whereby a computer initially boots from firmware installed on the computer's network card. When the client computer boots, it sends out an initial request for a PXE server. If a server that hears this request uses PXE, it sends the client a list of boot servers that contain the bootloaders available.

Using this method, a computer can download and run Linux without the need to install the operating system to its local hard drive. This method is often used on thin clients where no hard drive exists. This method can also be useful for troubleshooting a system where the bootloader has become corrupt and can no longer boot.

When using PXE, be aware of the following:

- The NIC in the client computer must support PXE boot.
- The client BIOS/UEFI must be configured to use PXE.
- The PXE client uses DHCP to obtain an IP address.
- A Trivial File Transfer Protocol (TFTP) server sends the necessary files to the client. The TFTP server can reside on the DHCP server.
- Linux servers can be configured as PXE servers.

## NFS

Network File System (NFS) is a file system that allows the storage and retrieval of data over a network. You can use the NFS file system to boot Linux on a client in a diskless/thin client environment. You can also boot Linux from the NFS file system to reduce administrative overhead. Booting from an NFS server requires a PXE environment. To boot Linux using NFS, you must specify the following for configuration:

- NFS client support as built-in.
- The server and the name of the directory to mount as root, (e.g., root=/dev/nfs). This is not the path to the actual server, but a synonym to indicate that NFS is being used.
- nfsroot using **nfsroot=[<server-ip>:]<root-dir>[,<nfs-options>]**. NFS options for nfsroot include:
    - Port from the server portmap daemon.
    - Rsize and wsize to specify the buffer size for read and write requests.
    - timeo to specify the timeout threshold in tenths of a second.
- Configuration for IP addresses or use the **ip=autoconf** to indicate autoconfiguration.
- nfsrootdebug to enable debugging messages to display during kernel boot.

## ISO

An ISO image is a file that is an identical copy of the original media. ISO images are a convenient and common way to load a Linux distribution. An ISO image containing the Linux distribution can be used from a variety of media including a CD/DVD, a USB stick, and a hard disk drive (HDD). The media containing the .iso image can be configured to be bootable, allowing the Linux distribution to be loaded from the media. A common system recovery tool is a bootable CD/DVD or USB drive with a Linux iso file.

ISO files are:

- Uncompressed and can be in any format
- A sector-by-sector copy of data
- Stored inside a binary file
- Identified by .iso extension

## HTTP

HTTP is another option for booting the Linux kernel and is a function in the UEFI specifications. HTTP boot is client-server communication-based application. It combines the Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), and Hypertext Transfer Protocol (HTTP) to provide the capability for system deployment and configuration over the network. Because HTTP boot provides higher performance (larger files and greater distances than TFTP), it is used to replace the PXE boot method of network deployment.

The basic process for booting using HTTP is as follows:

- The client initiates the DHCP process by broadcasting a DHCP request containing the HTTP Boot identifier. A DHCP server that supports the HTTP Boot extension provides a

boot resource location in Uniform Resource Identifier (URI) format to the client.
- The URI points to the Network Boot Program that is appropriate for the client's request. Then the client uses the HTTP protocol to download the NBP from the HTTP server to its memory.
- The client executes the downloaded NBP image. This program may then use other UEFI interfaces for further system setup based on the NBP design.

A Linux server can be configured as the HTTP boot server. The HTTP boot client must support and be configured (using the UEFI setup) to support HTTP boot. HTTP can be accessed from an existing Linux system by:

- Editing the grub.cfg file to load the kernel and initrd through the HTTP protocol.
- Using persistent memory firmware support in the UEFI that will pull an ISO file from the HTTP server.

---