

How To Network Boot (PXE) The Ubuntu LiveCD



With Ubuntu's latest release out the door, we thought we'd celebrate by showing you how to make it centrally available on your network by using network boot (PXE).

Overview

We already showed you how to setup a PXE server in the "[What Is Network Booting \(PXE\) and How Can You Use It?](#)" guide, in this guide we will show you how to add the Ubuntu LiveCD to the boot options.

If you are not already using Ubuntu as your number one "go to" for [troubleshooting, diagnostics and rescue procedures](#) tool... it will probably replace all of the tools you are currently using. Also, once the machine has booted into the Ubuntu live session, it is possible to perform the OS setup like you normally would. The immediate up shut of using Ubuntu over the network, is that if your already using the CD version, you will never again be looking for the CDs you forgot in the CD drives.

Image by [Dryope](#).

Prerequisites

- It is assumed that you have already setup the FOG server as explained in our "[What Is Network Booting \(PXE\) and How Can You Use It?](#)" guide.
- All the prerequisites for the FOG setup guide apply here as well.
- This procedure has been used to make Ubuntu 9.10 ([Karmic Koala](#)) up to and including 11.04 ([Natty Narwhal](#)) network bootable. It may work for other Ubuntu like distributions (like Linux Mint) but hasn't been tested.
- You will see me use VIM as the editor program, this is just because I'm used to it... you may use any other editor that you'd like.

How does it work?

In general the Ubuntu LiveCD boot process that we all know is like so:

- You put a CD into the cdrom drive the BIOS knows how to use the cdrom enough to get the boot program on the cdrom (isolinux).

- Isolinux is responsible for the menu options. Once you select a boot entry like “Start or install Ubuntu”, it calls the kernel + initrd (initial ram disk) files, copies them into memory and passes parameters to them.
- The now in RAM and in control kernel + initrd start the boot process, while using the parameters that were passed to them to determine things like: should the splash screen be shown? should the output be verbose?.
- When the initrd scripts have finished loading drivers and device information, they look for the Ubuntu liveCD files to continue the boot process. The normal behavior is to look in the local physical cdrom drive.

For network boot:

- Instead of a local media such as a CD, the client is booted using its network card (PXE) and is supplied with PXELinux over TFTP.
- Just like Isolinux, PXELinux is responsible for the menu options. Once you select a boot entry, it calls the Ubuntu kernel + initrd files, copies them into memory and passes parameters to them.
- The now in RAM and in control kernel + initrd start the boot process, with our additional information that they should not be looking for the boot files in the client's local physical cdrom drive, but rather in an NFS share on our FOG server.

This is possible because the Ubuntu creators have enabled networking by integrating, network cards drivers and protocols into the kernel + initrd files. For such an act, we can only say [thank you to the Ubuntu team.](#) 😊

Make the Ubuntu files available on the server

The first step is to make the Ubuntu files available on the server. You may opt to simply copy them from the CD drive, or extract them from the ISO, and that will work just fine. With that said, we will make the ISO auto-mounted. While not a must, doing this will enable you to use our “[How to Upgrade your Ubuntu ISO Without Re-downloading](#)” guide, to upgrade the Ubuntu version of your network boot without going through all the procedures from scratch or alternatively, replace a single file to update the entire entry.

With the above said, This author likes keeping a couple of past versions around, until the new one has been proven absolutely stable and issues free. That is why we will make a sub-directory and mount point according to version, but know that you could bypass that to have your single point of update.

1. Copy the ISO into the “/tftpboot/howtogeek/linux” directory
2. Create the mount point:

```
sudo mkdir -p /tftpboot/howtogeek/linux/ubuntu/<version-of-ubuntu>
```

3. Edit the “fstab” file to have the ISO auto-mounted at boot:

```
sudo vim /etc/fstab
```

4. Append to “fstab” the ISO mount entry:

```
/tftpboot/howtogeek/linux/ubuntu-11.04-desktop-amd64.iso /tftpboot
/howtogeek/linux/ubuntu/11.04 udf,iso9660 user,loop 0 0
```

Note: Despite representation, this is one unbroken line.

5. Test that the mount point works by issuing:

```
sudo mount -a
```

6. If all went well, you should be able to list the contents of the ISO by issuing:

```
ls -lash /tftpboot/howtogeek/linux/ubuntu/11.04/
```

Create an NFS share

While the boot procedure starts by using PXE, the actual heavy lifting is done by the NFS share on the server. As we are basing this guide on our FOG server, the NFS components and some configurations have already been done for us by the

FOG team, and all we have to do is add to them our Ubuntu share.

1. Edit the “exports” file to add the new share:

```
sudo vim /etc/exports
```

2. Append to it the pointer to our ISO mount point:

```
/tftpboot/howtogeek/linux/ubuntu/11.04/  
*(ro,sync,no_wdelay,insecure_locks,no_root_squash,insecure)
```

3. Restart the NFS service for the settings to take affect:

```
sudo /etc/init.d/nfs-kernel-server restart
```

PXE menu setup

Edit the “Linux stuff” menu:

```
sudo vim /tftpboot/howtogeek/menus/linux.cfg
```

Append to it the following:

```
LABEL Ubuntu Livecd 11.04  
MENU DEFAULT  
KERNEL howtogeek/linux/ubuntu/11.04/casper/vmlinuz  
APPEND root=/dev/nfs boot=casper netboot=nfs nfsroot=<YOUR-SERVER-  
IP>:/tftpboot/howtogeek/linux/ubuntu/11.04 initrd=howtogeek/linux  
/ubuntu/11.04/casper/initrd.lz quiet splash --
```

The above may look messy at first glance but all you have to do is replace *<YOUR-SERVER-IP> with the IP of your server NFS/PXE server.

For a clearer geek understanding, the text above will:

- Create a new PXE entry in the “Linux” sub-menu called “Ubuntu 11.04”.
- Because of the “MENU DEFAULT” parameter, this entry will be automatically selected when entering the “Linux” sub-menu.
- Point the client to take the kernel + initrd files using TFTP from the relative path in the “/tftpboot” directory of “howtogeek/linux/ubuntu...”
- Point the initrd scripts to mount the “root” filesystem from the NFS share on the absolute path of “<YOUR-SERVER-IP>:/tftpboot/howtogeek...”

Note: [I have tried \(and failed\)](#) to use a DNS name instead of an IP for the “<YOUR-SERVER-IP>”, I’m guessing that at that stage of the boot process there simply still isn’t support for DNS... success stories are welcomed.

Possible procedures

You should now be able to boot a client into Ubuntu from PXE (Usually F12).

At this stage we suggest you take the time to review some of the things you can do with this outstanding tool:

- [The 10 Cleverest Ways to Use Linux to Fix Your Windows PC](#)
- [Reinstall Ubuntu Grub Bootloader After Windows Wipes it Out](#)
- [How to Format a USB Drive in Ubuntu Using GParted – How-To Geek](#) & [Using GParted to Resize Your Windows 7 or](#)

[Vista Partition](#)

One last thing, If you create your Ubuntu ISO, using [this online builder](#), you will be able to slipstream all of the articles above into your PXE bootable Ubuntu.

[Ubuntu is everything, all that once was and all that will be, Ubuntu controls time and space, Love and Death, Ubuntu can see into your mind, Ubuntu can see into your SOUL!!](#)

JOIN THE DISCUSSION

Aviad Raviv, is an agile IT implementer that has reached the level of "Bankai" and is now working on incorporating the "hollow" side.	• Published 05/2/11

SHOW ARCHIVED READER COMMENTS (28)