

PXE Server With Ubuntu 9.10

Contributed by Administrator
Sunday, 07 March 2010
Last Updated Monday, 08 March 2010

This tutorial shows how to set up a PXE (short for preboot execution environment) install server with Ubuntu 9.10 (Karmic Koala).

A PXE install server allows your client computers to boot and install a Linux distribution over the network, without the need of burning

Linux iso images onto a CD/DVD, boot floppy images, etc. This is handy if your client computers don't have CD or floppy drives, or if you

want to set up multiple computers at the same time (e.g. in a large enterprise), or simply because you want to save the money for the

CDs/DVDs. I want to say first that this is not the only way of setting up such a system. There are many ways of achieving this goal but this is the

way I take. I do not issue any guarantee that this will work for you!

1. Preliminary Note

It is important that you have a decent internet connection because your client computers will fetch all needed packages from the repositories in the Internet.

And the most important thing is that your client computers support booting over the network. You should check each computer's BIOS for this option.

I am going to be using a computer with two network connections, I will have two networks as well one for the internet and one for the client computers.

My server name will be pxe-server my user will be pxe and all passwords will be vadmin.

Most tasks in this guide will be carried out in terminal to access this follow: Applications --> Accessories --> Terminal 2. Installing Ubuntu 9.10

First thing we are going to need is a Ubuntu 9.10 CD/DVD. This can be obtained though many different distributors or downloaded from ubuntu.com

Insert your Ubuntu install CD into your system and boot from it. Select Install to the hard disk:

The installation starts, and first you have to choose your language:

Then select your location and time zone:

Then select your keyboard layout:

Next partitioner will start, and you will need to partition the hard drive, I have selected the whole disc for ubuntu:

Now enter your user details, my password is vadmin:

Next you will be asked to confirm options and install:

Ubuntu will now install onto your system.

Now reboot.

3. Initial Server Configuration Setup

After the computer restarts, log in with the user (pxe) you created during installation.

We now need to set the password for the root account, open up terminal and type: `sudo passwd`
and give root a password. (vadmin)

Now check you have the best update repository setup. System --> Administration --> Software Sources

Use the drop down list and select Other.

Then click Select Best Server.

It will then test which server is best for you to use.

Let update manager install updates or run in terminal: `sudo apt-get update`

Restart your system: `sudo reboot`

If you have more than one ethernet port you will need to edit `/etc/udev/rules.d/70-persistent-net.rules` to ensure our network interfaces match the labels on the system

First check if your network connections are as you expected: `ifconfig`

If they are not as expected then: `sudo gedit /etc/udev/rules.d/70-persistent-net.rules`

Switch `NAME="eth1"` to `eth0` and vice versa.

Restart your system: `sudo reboot`

4. Setting Up A Static IP Address

Newer versions of Ubuntu use an application called network-manager to manage network connections, which works great with desktops and laptops, but not so well for servers, so we are going to remove network-manager. `sudo apt-get remove network-manager`

Now let's set up the static IP addresses: `sudo gedit /etc/network/interfaces`

Enter the following information in the text file, `eth0` will be our Internet connection and `eth1` will be our client connection:

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
    address    [your.static.ip]
    netmask    [your.net.mask]
    network    [your.network]
```

```

        broadcast    [your.broadcast.addy]
        gateway      [your.gateway]
auto eth1
iface eth1 inet static
    address         [your.static.ip]
    netmask         [your.net.mask]
    network         [your.network]
    broadcast       [your.broadcast.addy]

```

Here is a example: auto lo

```

iface lo inet loopback
auto eth0
iface eth0 inet static
    address         192.168.1.4
    netmask         255.255.255.0
    network         192.168.1.0
    broadcast       192.168.1.255
    gateway         192.168.1.1
auto eth1
iface eth1 inet static
    address         192.168.2.1
    netmask         255.255.255.0
    network         192.168.2.0
    broadcast       192.168.2.255
#    gateway         192.168.2.1 leave out otherwise your internet connection wont work

```

Then save this file and restart your networking settings. `sudo /etc/init.d/networking restart`

You can confirm your ip address is set by typing the following after your networking has restarted. `ifconfig` 5. Installing Necessary Packages

We are going to install the packages required for the PXE server:

- `tftpd-hpa` - this is the file transfer server
- `dhcp3-server` - linux dhcp server, not needed if you allready have a dhcp server
- `openbsd-inetd` - new inetd server `sudo apt-get install tftpd-hpa dhcp3-server openbsd-inetd`

6. Setting Up Tftp Server

Edit `/etc/inet.conf` to and ensure the line is correct, last part is important: `sudo gedit /etc/inetd.conf tftp dgram udp wait root /usr/sbin/in.tftpd /usr/sbin/in.tftpd -s /tftpboot`

Ensure that it reads `/tftpboot` rather than `/var/lib/tftpboot`.

We need to check that the tftpd server is running. `netstat -lu`

Check for this line:

```

Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp      0      0 *:tftp                  *.*

```

If you don't see the tftp line, then go back and edit `/etc/inet.conf` again.

Then enable inetd at boot: `sudo update-inetd --enable BOOT`

Now restart your inetd server: `sudo /etc/init.d/openbsd-inetd restart`

Next we need to edit the `/etc/default/tftpd-hpa` file to change the boot directory: `sudo gedit /etc/default/tftpd-hpa`

Change the file to look like this: `RUN_DAEMON="yes"`
`OPTIONS="-l -s /tftpboot"`

Restart tftpd server: `sudo /etc/init.d/tftpd-hpa restart`

We now need to create a folder to store our boot stuff in. `sudo mkdir /tftpboot`

Now we need to change the user to nobody: `sudo chown nobody /tftpboot`

Then we need to let everyone read and write to /tftpboot: `sudo chmod 777 /tftpboot`

Now check if that worked: `ls -ld /tftpboot`

You should see a line similar to this:

```
drwxrwxrwx 2 nobody root 4096 2010-01-28 15:04 /tftpboot
```

7. Setting Up DHCP Server

On what network interfaces should the DHCP server (dhcpd) serve DHCP requests? Answer this question by editing the line in `/etc/default/dhcp3-server` file. Separate multiple interfaces with spaces, e.g. "eth0 eth1". As I only want the DHCP server running on eth1, I need to change this line. `sudo gedit /etc/default/dhcp3-server`

Change the INTERFACES line to: `INTERFACES="eth1"`

Now we need to back up the configuration files for dhcpd server: `sudo cp /etc/dhcp3/dhcpd.conf /etc/dhcp3/dhcpd.conf_orig`

Now we can configure dhcpd: `sudo gedit /etc/dhcp3/dhcpd.conf`

Here is my file, you will need to change the values to suit. `default-lease-time 86400;`
`max-lease-time 604800;`
`authoritative;`

```
subnet 192.168.2.0 netmask 255.255.255.0 {
  range 192.168.2.2 192.168.2.255;
  filename "pxelinux.0";
  option subnet-mask 255.255.255.0;
  option broadcast-address 192.168.2.255;
  option routers 192.168.2.1;
}
```

This will dynamically assign IP addresses from the range 192.168.2.2 to 192.168.2.255 to your client computers. The gateway is 192.168.2.1.

It is important that you have the line `filename "pxelinux.0";` in your configuration!

Then restart your DHCP server: `sudo /etc/init.d/dhcp3-server restart`

If you already have a DHCP server in your network, you must modify its configuration. Let's assume you have something

```
like subnet 192.168.2.0 netmask 255.255.255.0 {
  range 192.168.2.2 192.168.2.255;
  option subnet-mask 255.255.255.0;
  option broadcast-address 192.168.2.255;
  option routers 192.168.2.1;
}
```

in the configuration. You must add `filename "pxelinux.0";`
`next-server 192.168.2.100;`

to it (where 192.168.2.100 is the IP address of our Ubuntu PXE server) so that it looks like this: `subnet 192.168.2.0`
`netmask 255.255.255.0 {`
 `range 192.168.2.2 192.168.0.255;`
 `option subnet-mask 255.255.255.0;`
 `option broadcast-address 192.168.2.255;`
 `option routers 192.168.2.1;`
 `filename "pxelinux.0";`
 `next-server 192.168.2.100;`
`}`

Now verify that it DHCP is running (if not, you may have a problem with you dhcp config file). `ps ax | grep dhcpd`

8. Setting Up Boot Files

We now need to get the netboot files. `cd /tftpboot`

Download files with lftp: `lftp -c "open http://archive.ubuntu.com/ubuntu/dists/karmic/main/installer-i386/current/images/netboot/; mirror"`

Instead, if Ubuntu is in your cdrom drive: `mount /media/cdrom`
`cp -a /media/cdrom/install/netboot/* /tftpboot/`

Now we can add a splash screen to our configuration, first we need the versamenu.c32 file. `cp /tftpboot/ubuntu-installer/i386/boot-screens/vesamenu.c32 /tftpboot`

Second we need a splash image;

Here is one I created for HowtoForge, download. `lftp -c "get http://www.howtoforge.com/images/pxe_install_server_ubuntu_9.10/howtoforge_pxe.png;"`

We then need to edit `/tftpboot/pxelinux.cfg/default` to include our boot options and splash screen, the file is pretty self-explanatory. `sudo gedit /tftpboot/pxelinux.cfg/default` default vesamenu.c32
 Menu Background howtoforge_pxe.png
 Menu Title Boot Menu

```
label install
menu label ^Install
menu default
kernel ubuntu-installer/i386/linux
append vga=normal initrd=ubuntu-installer/i386/initrd.gz -- quiet
```

```
label expert
menu label ^Expert install
kernel ubuntu-installer/i386/linux
append priority=low vga=normal initrd=ubuntu-installer/i386/initrd.gz --
```

```
label cli-expert
menu label Command-^line expert install
kernel ubuntu-installer/i386/linux
append tasks=standard pkgset/language-pack-patterns= pkgset/install-language-support=false priority=low vga=normal
initrd=ubuntu-installer/i386/initrd.gz --
```

```
label rescue
menu label ^Rescue mode
kernel ubuntu-installer/i386/linux
append vga=normal initrd=ubuntu-installer/i386/initrd.gz rescue/enable=true -- quiet
```

```
label Local_drive
localboot 0
menu label ^Local Drive
```

```
prompt 0
timeout 0
```

Your `/tftpboot` directory should look like this:

```
/tftpboot/
/tftpboot/pxelinux.0
/tftpboot/mini.iso
/tftpboot/netboot.tar.gz
/tftpboot/howtoforge_pxe.png
/tftpboot/ubuntu-installer/i386
```

```
/tftpboot/ubuntu-installer/i386/initrd.gz
/tftpboot/ubuntu-installer/i386/linux
/tftpboot/ubuntu-installer/i386/pxelinux.0
/tftpboot/ubuntu-installer/i386/pxelinux.cfg
/tftpboot/ubuntu-installer/i386/pxelinux.cfg/default
/tftpboot/ubuntu-installer/i386/boot-screens
/tftpboot/ubuntu-installer/i386/boot-screens/adtext.cfg
/tftpboot/ubuntu-installer/i386/boot-screens/f1.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f2.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f3.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f4.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f5.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f6.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f7.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f8.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f9.txt
/tftpboot/ubuntu-installer/i386/boot-screens/f10.txt
/tftpboot/ubuntu-installer/i386/boot-screens/menu.cfg
/tftpboot/ubuntu-installer/i386/boot-screens/po4a.cfg
/tftpboot/ubuntu-installer/i386/boot-screens/prompt.cfg
/tftpboot/ubuntu-installer/i386/boot-screens/splash.png
/tftpboot/ubuntu-installer/i386/boot-screens/stdmenu.cfg
/tftpboot/ubuntu-installer/i386/boot-screens/text.cfg
/tftpboot/ubuntu-installer/i386/boot-screens/vesamenu.c32
/tftpboot/pxelinux.cfg
/tftpboot/pxelinux.cfg/default
```

9. Booting The Client (Please make sure that the computers that you don't want to reinstall have the network boot option disabled in their BIOS settings because otherwise it is possible that you or someone else accidentally installs Ubuntu over the existing operating system!)

Now you can boot up your first client computer. Make sure you specified in its BIOS settings that it should use the network as its first boot device. If everything goes well, you should see the splash screen, and you can choose from one of the installation options from the `/tftpboot/pxelinux.cfg/default` file.

Don't forget to change the order of the boot devices after the successful installation (e.g. disable booting over the network and make the HDD the first boot device) because otherwise you will start another installation!