- Partners
- Support
- Community
- Ubuntu.com

- Page History
- Login to edit

[Search]

Installation/LocalNet

Overview

This installation method utilizes DHCP/BOOTP, TFTP, and FTP/HTTP to obtain the installation sources. The installer itself is made up of a Linux kernel and an initrd. These files are transferred via the DHCP/TFTP boot process.

The remainder of the installation sources can be transferred via FTP or HTTP. If the target machine has internet access these sources could be pulled directly from an Ubuntu mirror. Alternatively, you could download one of the installation ISO's, loop-back mount them, and serve them with a local HTTP/FTP server.

Preparing for Network booting

The following describes how you can install Ubuntu onto a client machine, using just its network card to boot an Ubuntu installation image which will install Ubuntu. The way this works, in summary is: you set up and configure bootp, tftp, dhcp, and internet gateway services. Your client machine will

1. 'netboot' via its network adapter, using pxe/bootp protocol,
2. get passed a kernel and installer via tftp. It will boot into that kernel, which will run an installation of Ubuntu onto your client machine.

Most of the packages installed come from latest versions from the Ubuntu repositories on the internet.

Set up the servers

Ensure that you have a bootp server, a tftp (eg tftp-hpa) server, and access to your router/gateway.

1. Configure the router and its dhcp server. Get the mac address of the new machine you will be installing ubuntu onto (the 'client machine'). It is easiest if you decide on a fixed ip address for the client machine and set it into the router, as a fixed ip address, eg 192.168.1.42. Note also the gateway address of the router.
2. Configure a bootp service on a linux host. The same ip address as above needs to be assigned by the bootp service to the client machine.

    1. Install bootp.

    2. Edit /etc/bootptab. Here is an example.

    ```
    client:\
    ha="00:00:39:2B:54:B5":\
    ip=192.168.1.42:\
    gw=192.168.1.1:\
    sm=255.255.255.0:\
    td=/: hd=/: bf=pxelinux.0
    ```

        - ha = the mac address of the client machine (set this to its value)
        - ip = the ip address assigned in the router to the client machine

- gw = the address of the router
- sm = the subnet mask for your lan
- td = the directory on the tfptboot server where the pxe files reside, relative to the tftpboot server's root.
- hd = a further path which ends at either the pxe boot file or a link to it. (this is probably ok as is)
- bf = the pxe boot file (this is probably ok as is)

For further help: `man bootpd; man bootptab` and also found in the etc/bootptab file itself as comments.

If you don't know the hardware address, or which ip address the dhcp server of the router will assign, start the client machine. Make sure its bios is set to boot first from its network adapter. You should then be able to get the hardware mac address as it attempts to boot via the network adapter. Update `/etc/bootptab` on the bootp server with the client's mac address and the ip address for it as set in the dhcp server (eg in the router), then restart bootpd.

3. Start bootp: Here is a wrapper to start and stop bootpd from the command line.

```
vDaemon=bootpd
vCd=/var/lib/tftpboot

Start () {
    echo -n "Starting $vDaemon: default current directory is at $vCd ... :"
    /usr/sbin/$vDaemon -d 4 -c $vCd >/tmp/$vDaemon.log 2>/tmp/$vDaemon.err &
    sleep 1
    Status
}

Stop () {
        echo "Stopping $vDaemon ..."
    kill `pidof $vDaemon`
}

Reload () {
    if [ "`pidof $vDaemon`" ] ; then
        echo "Reloading config file for $vDaemon ..."
        kill -HUP "`pidof $vDaemon`"
    fi
    Status
}

Status () {
    vPid="`pidof $vDaemon`"
    if [ "$vPid" ] ; then
        echo "$vDaemon running, pid=$vPid"
    else
        echo "$vDaemon not running"
    fi
}

case "$1" in
    start)     Start ;;
    stop)      Stop ;;
    reload) Reload ;;
    restart) Stop ; sleep 2; Start ;;
    status) Status ;;
    ""|*) echo `basename $0` parameter: start stop status reload or restart ;;
esac
```

Preparing Files for TFTP Net Booting, Section "Setting up BOOTP server" has more info on bootp, including instructions for starting bootp using the inetd alternative.

4. Configure the tftp server:

1. Install tftpd-hpa. Its config in /etc/default/tftpd-hpa should be

```
RUN_DAEMON="yes"
OPTIONS="-l -s /var/lib/tftpboot"
```

The -s parameter is your tftp server file root.

2. Download the netboot installer tarfile from the Ubuntu Archives (karmic, jaunty), and extract its contents into the tftpboot server file root as per above.

(More background info is at Preparing Files for TFTP Net Booting).

Desktop Installation

From internet mirrors

Net-boot your client machine. The tftpboot server passes the client machine the install kernel and initrd, which presents a screen/menu with different installation options. Hit [Enter] to start a standard desktop installation.

When the installation asks to select a mirror, choose one and the installation will proceed as if there is a CD in the drive.

From a local server

If installing over the internet is not an option you can also install from a local server. One can loop-back mount the installation sources from the .ISO file. Based on the speed of your LAN, installing systems via this method can be considerably faster than using spinning media.

For a desktop installation, obtain one of the desktop ISO's. For example:

```
# wget http://old-releases.ubuntu.com/releases/9.10/ubuntu-9.10-desktop-amd64.iso
```

When the download has completed, loopback mount the ISO:

```
# mkdir /mnt/ubuntu
# mount -o loop ubuntu-9.10-desktop-amd64.iso /mnt/ubuntu
```

Expose your installation media via your HTTP server. If you are running Apache, the default configuration should be generous enough to allow you to use a symbolic link:

```
# cd /var/www/html
# sudo ln -s /mnt/ubuntu ubuntu
```

Test that your installation directory is accessible by pointing a browser at: For example, if Apache was running on the machine 192.168.1.50, then point Chrome to 'http://192.168.1.50/ubuntu'

Network boot the target machine, proceed to the mirrors dialog. When asked to select a mirror, go to the top of the list and select "Enter Manually".

Enter the IP address of the HTTP server which is sourcing your installation packages (you can use a hostname if your LAN's domain name resolution is configured properly).

At the next screen, it'll ask for a path. Enter /ubuntu.

From here on, the installation is the same as using the internet mirrors.

Alternate installation

If you're installing to a "headless" machine, or a machine that only has a serial connection, the "desktop" installation media will not work. You will, instead, need the "alternate" media for your installation.

Modify your pxelinux.cfg/default so that the kernel uses ttyS0 for the console:

```
LABEL alt
kernel ubuntu-alt/linux
append initrd=ubuntu-alt/initrd.gz console=ttyS0,115200n8 quiet --
```

After downloading the "alternate" media:

```
# wget http://old-releases.ubuntu.com/releases/9.10/ubuntu-9.10-alternate-amd64.iso
```

Follow the "local server" instructions substituting the "alternate" media for "desktop" media.

A variation: Booting the "Live CD" image

You can also employ a variation of the above to instead boot into the Desktop Edition Live CD image instead of the installer. To do this, here are the key steps:

1. Download the Desktop CD iso (eg /8.10/ubuntu-8.10-desktop-i386.iso from a mirror at http://www.ubuntu.com/getubuntu/downloadmirrors). Extract its contents including the hidden .disk directory, or mount the iso file, to the tftp root area. This needs to be reachable via nfs export; eg on the nfs host (192.168.1.10 as per the example below).

If you 'nfs-export' /, ie the nfs server's entire file system, you will not have a problem 😊 (Alternatively, you could also use an ftp or http server, as further described in the next section.)

2. Add a stanza of the following 3 lines into the ubuntu-installer/i386/pxelinux.cfg/default file

```
LABEL live
kernel ubuntu-desktop/casper/vmlinuz
append initrd=ubuntu-desktop/casper/initrd.gz boot=casper netboot=nfs nfsroot=192.168.0.10:/mnt/u01/
```

3. Add a line into the ubuntu-installer/i386/boot-screens/f3.txt menu file to record your selection, so you can remember the extra option *live*.

Your client machine can now boot the "Live CD" Ubuntu instead of the alternate CD installer, and you will get the Live Ubuntu desktop. You can then install Ubuntu via the "Install" icon on the desktop if you wish.

- With Gutsy or Hardy, you might find the bootup stops just after this line appears. `"squashfs: version 3.2-UBUNTU (2007/07/26) Phillip Lougher"` In this case, try hitting Alt-Enter, and the boot-up should continue, giving you the Ubuntu desktop.

- Up until late July 2008 for Hardy or earlier, if you fail to copy the hidden directory .disk, the install will stop at 82%, filling up the squash filesystem. After then it should abort cleanly. See this bug report on Launchpad for details.

Basic: Hands-On Interactive Network Server Edition Install

Another description of installing an Ubuntu distro from over your network, as if you had booted and installed it from the Ubuntu installation CD. This is almost a rewrite of the above, except its using a DHCP server on a Linux machine.

Services needed, on a server (or servers) :

1. dhcp or bootp: to provide netboot server support,
2. tftp: to feed the first boot image to the netboot client machine, when requested by the netboot server,
3. http, ftp or nfs: to supply the Ubuntu distro to the client machine during the installation process.
4. The client machine must be able to boot from its network card, else from a diskette that you built to 'netboot'.

This page focuses on the combination of dhcp, tftp, and http. In the examples below 'myserver' is the server for these 3 services. (Each of these can be run on a separate server if desired.)

1. Install and configure the dhcp server. Eg to install:

```
root@myserver:~ # apt-get install dhcp3-server
```

Configure the dhcp server to tell the clients what to boot. I added a default host name, you don't need it but it comes in handy for other things.

```
root@myserver:~ # cat /etc/dhcp3/dhcpd.conf
subnet 192.168.1.0  netmask 255.255.255.0 {
        range 192.168.1.10 192.168.1.254;
          }
# put your DNS IP's here:
option domain-name-servers 192.168.1.7, 68.87.66.196 ;
filename="ubuntu/feisty/alternate/install/netboot/pxelinux.0";

root@myserver:~ # /etc/init.d/dhcp3-server restart
Stopping DHCP server: dhcpd3.
Starting DHCP server: dhcpd3.
```

2. Install and configure the Trivial File Transfer Protocol server. The tftpd-hpa package is recommended. Install it. It is enabled via /etc/default/tftpd-hpa.

```
RUN_DAEMON="yes"
OPTIONS="-l -s /var/lib/tftpboot"
```

Also note its base directory. The default setting is shown above.

Mount the CD (media or image) under the tftpboot base directory found above,eg

```
root@myserver:/var/lib/tftpboot/ubuntu/feisty$ mount -o loop feisty-alternate-i386.iso alternate/
```

or, you can instead copy the contents of the CD into that location, instead of leaving the ISO mounted.

See Preparing Files for TFTP Net Booting for detailed information.

1. Install and configure Apache, eg:

   ```
   root@myserver:~ # apt-get install apache2
   ```

   Make a symlink from apache's doc Root to the CD

   ```
   root@myserver:/var/www # ln -s /var/lib/tftpboot/ubuntu/
   ```

   Or copy the /ubuntu directory from the CD into here.To use a pure ftp server like the vs-ftp server instead of Apache, install the ftp server, and configure it for anonymous ftp access. Then copy the /ubuntu directory from the CD into the anonymous ftp home. With vs-ftpd, this is set to /home/ftp. (Check /etc/passwd to confirm the anonymous ftp home.) Next test that anonymous ftp works and finds the installation stuff. eg via 'ftp localhost' on that machine. The install needs to be done in 'expert' mode, so you can select the ftp protocol instead of the default 'http'.
2. Boot up the client machine, set its bios to net boot and you should get the Ubuntu screen and Boot: prompt. Yippee!

Advanced: Hands-Off, Preseeded Network Server Install

The above is for an interactive install, just as if you booted from CD. The following addresses preseeding, which enables a Hands Off Install where all the questions have been answered and fed to the installer. In addition, the following does not pull anything from the Internet.

A handful of preseed options need to be passed via kernel options. The kernel only supports 255 chars, so things are cramped. To accommodate this, I dumped some of the paths and made symlinks. That gave me just enough space for now. Note that the path/name of the kernel gets appended, so even trimming that helped.

```
root@sahara:/var/lib/tftpboot # ln -s ubuntu-installer/i386/initrd.gz
root@sahara:/var/lib/tftpboot # ln -s ubuntu-installer/i386/linux
```

/var/lib/tftpboot/pxelinux.cfg/default

```
# pxelinux.cfg/default

# display ubuntu-installer/i386/boot-screens/syslinux.txt

default menu
prompt 1
timeout 150
ontimeout boothd

label menu
# makes a menu out of this file, allows editing the options on the client
        kernel menu.c32

label boothd
        # boot from the first HD
        # (this is what happens if nothing is pressed for 15 seconds)
        localboot 0

label ubuntu-feisty-normal
# ubuntu installer
        kernel ubuntu-feisty/linux
        append vga=normal initrd=ubuntu-feisty/initrd.gz ramdisk_size=14984 root=/dev/rd/0 rw  --

label ubuntu-feisty-hands-off
# ubuntu installer
# The 'kernel' and 'initrd' paths must identify files under tftpboot.
# For example, on my system I have /var/lib/tftpboot/ubuntu-feisty/linux and /var/lib/tftpboot/ubuntu-fei
        kernel ubuntu-feisty/linux
        append initrd=ubuntu-feisty/initrd.gz ramdisk_size=14984 root=/dev/rd/0 rw preseed/locale=en_US k

label ubuntu-gutsy-hands-off
# For gutsy the preseeds on the command line are slightly different.
        kernel ubuntu-gutsy/net-linux
        append initrd=ubuntu-gutsy/net-initrd.gz auto url=http://server/u710a/preseed/web-server.preseed
```

netcfg/get_hostname= Is blank so that it will pick up the host-name supplied by the DHCP server.

netcfg/choose_interface=eth0 - You may wish to specify eth1 or wlan0 for laptops if you want it to setup the wifi card, not the wired port.

Note for client-specific PXE configurations: The DHCP server can't pass it this can be done in PXE boot. Pretending that the MAC address of your wireless card is 12 : CD : 56 : AB : 78 : EF you can specify a custom pxelinux.cfg/01-12-dc-56-ab-78-ef (lowercase) file instead of default. If you are using static dhcp, you can also use the IP address in hex for a per-machine or per-subnet or per-network configuration. See http://syslinux.zytor.com/pxe.php#config for more information. Basically, say you have three wireless cards where the IP in hex translated to caf12d5e, caf12d6b, caf124ea, you could specify pxelinux.cfg/caf12d and pxelinux.cfg/caf12. The first two would match to the first config file and the third would match to the second config file.

See http://www.debian.org/releases/etch/example-preseed.txt for an example preseed file, and be sure to note the debconf-get-selections example commands. Here are the important changes for using the copy of the install CD on your server instead of the Internet repositories during the install:

```
# Choose software to install:
tasksel tasksel/first   multiselect     ubuntu-desktop
# Installation source
# Note the value of mirror/country "enter information manually" This is an actual choice in the mirror me
d-i     mirror/country          string enter information manually
# hostname This is whatever HTTP server you have set up.
d-i     mirror/http/hostname    string sahara
# This is the /ubuntu directory from the install CD copied (or linked) under the webroot of your HTTP ser
d-i     mirror/http/directory   string /ubuntu
# Name your ubuntu version here
d-i     mirror/suite            string feisty
# d-i   mirror/suite            string gutsy
# no proxy
d-i     mirror/http/proxy       string
```

If you want to use the first disk of your system "d-i partman-auto/disk string /dev/discs/disc0/disc" (because its a devfs-path, the installer use udev) wont work for you. As described in http://ubuntuforums.org/showthread.php?t=434405 "d-i partman-auto/disk string ?da" works well! More information about the partman-auto/expert_recipe seed that allows setting the partitions can be found at http://d-i.alioth.debian.org/svn/debian-installer/installer/doc/devel/partman-auto-recipe.txt

If you get questions popping up, you can find the corresponding seed name by looking in the /var/lib/cdebconf directory in the installer. Open vty2 and activate the shell. File template.dat contains all questions and the corresponding seed value. If you answered any questions manually, you can find the seed value in questions.dat

Here is my current dhcpd.conf

```
ping-check = 1;
log-facility local7;

option domain-name "sahara.net";
option time-servers  192.168.1.1;
option broadcast-address 192.168.1.255;

option domain-name-servers 63.240.76.4, 204.127.198.4;
option routers  192.168.1.1;

subnet 192.168.1.0  netmask 255.255.255.0 {
        range 192.168.1.10 192.168.1.254;
         }

# Hoary test boxes
group {
        filename="pxelinux.0" ;

        # You need a next-server option if TFTP and DHCPd aren't on the same system.
        next-server sahara;

        # 192.168.1.2 does not exist on my lan.
        # I am setting it like this to make sure the box doesn't find
        # a repo on the net to pull sources from.
        # It would be good I can figure out how to i
        #  limit these setting to the install phase
        #  (including the 2nd part after the reboot )

        option domain-name-servers 192.168.1.2 ;
        option routers 192.168.1.2 ;

        host dwl650p {
                hardware ethernet 00:05:5d:5a:81:f0 ;
                option host-name "dwl650p" ; }

        # tsp2 6100 - a=wired, b=wifi
        host tsp2a {
                hardware ethernet 00:00:39:fa:ff:f3 ;
```

```
                           option host-name "tsp2a" ; }

           host tsp2b {
                           hardware ethernet 00:02:2d:b0:c8:6c ;
                           option host-name "tsp2b" ; }

           host tsp1 {
                           hardware ethernet 00:00:39:88:31:a6 ;
                           option host-name "tsp1" ; }

           host e400 {

                           hardware ethernet 00:60:08:b0:62:0d ;
                           # fixed-address 192.168.1.181 ;
                           option host-name "e400" ; }

       }
```

If your box doesn't have the option to boot from lan (like one of mine) you can make a boot disk that will do that part.

- http://etherboot.sourceforge.net/

- http://rom-o-matic.net/

I am pleased to say that I was able to read, download, build, run, make the floppy in under 30 min.

"Etherboot does not (yet) offer support for PCMCIA cards." so the older laptops will have to wait.

Advanced: Network install using apt-mirror

apt-mirror is a Perl script for maintaining a mirror of the Debian or Ubuntu installation sets. It is often set up as a cron job to download updates to the mirror automatically. Updates are downloaded incrementally using parallel threads. Contents of the mirror are typically served via a web server to the local network.

apt-mirror can be installed on most Unix machine from the tarball. On Debian or Ubuntu, the apt-mirror package can usually be installed by apt-get with the appropriate addition to sources.list. See the apt-mirror project home page for details.

For basic instructions pertaining to Ubuntu, see How To Create A Local Debian/Ubuntu Mirror With apt-mirror. Some modifications are required to the mirror configuration to work in conjunction with netboot. Netboot requires debian-installer. The following snippet from an example configuration is reported to work:

```
deb http://mirrors.kernel.org/ubuntu intrepid main main/debian-installer restricted restricted/debian-ins
deb http://mirrors.kernel.org/ubuntu intrepid-updates main restricted
deb http://mirrors.kernel.org/ubuntu intrepid-security main restricted

clean http://mirrors.kernel.org/ubuntu
```

You will need to modify your apt-mirror configuration file accordingly. Often this file is located at /etc/apt/mirrors.list. Note that not everyone chooses to mirror the security files unless the mirror updates regularly.

If your mirror is configured to support netboot, the following directories should be present:

```
 /ubuntu/dists/intrepid/restricted/debian-installer/
 /ubuntu/dists/intrepid/main/debian-installer/
```

If these portions of the file tree are not present on your mirror, the Intrepid netboot installer displays a generic error screen beginning with the text "The installer failed to download a file from the mirror."

Surprisingly, my netboot installation of the Feisty server release set did not install openssh-server by default. After all this work setting up netboot, it was back to the console again for a couple of minutes.

Related Links

Pile of PXE related links

- http://freshmeat.net/projects/syslinux/

- http://syslinux.zytor.com/memdisk.php

- http://support.3com.com/infodeli/tools/nic/mba.htm New Universal NDIS Driver for DOS

- http://www.qualystem.com/en/dualboot.html

- http://unattended.sourceforge.net/step-by-step.php

- http://unattended.sourceforge.net/advanced.php#pxe

- http://syslinux.zytor.com/archives/2003-June/002185.html

- http://www.intel.com/design/network/drivers/int21143.htm

- http://www.tux.org/pub/distributions/tinylinux/tomsrtbt/

- http://www.winimage.com/winimage.htm

- http://www.weird-solutions.com/docs/pxe_booting.pdf

- http://www.weird-solutions.com/bin/util/tftp_root.zip

- http://etherboot.sourceforge.net

- http://marc.herbert.free.fr/linux/win2linstall.html Install GNU/Linux without any CD, floppy, USB-key, nor any other removable media.

- http://osdev.berlios.de/netboot.html - Network-booting Your Operating System - the part I like: "...loads the GRUB, the second-stage loader, off the server."

PPC Mac related links

- http://www.macgeekery.com/hacks/how_to_install_debian_via_network_boot_from_a_mac

- http://davespicks.com/writing/programming/mackeys.html#boot

- SmartBootManager

CategoryInstallation

Installation/LocalNet (zuletzt geändert am 2011-10-03 20:08:53 durch guntbert @ chello084115174066.graz.surfer.at[84.115.174.66]:guntbert)