Booting Ubuntu Live Disc

To be able to boot into the Live Disc of the Ubuntu distribution, there are several requirements:

- The Ubuntu iso
- An NFS daemon
- Typical network booting infrastructure, including but not limited to:
    - Physical layer
    - DHCP
    - DNS

Due to the similar file structures between Ubuntu variants, this guide should also apply to Edubuntu, Xubuntu, Kubuntu, and so on. Derivatives based on Ubuntu are not guaranteed.

Note: While this guide will be able to boot the live environment, installs from this environment will not work entirely as expected when done from the actual disc. In short, networking will not immediately work [https://lists.ubuntu.com/archives/ubuntu-installer/2013-April/001285.html] if an install from this environment is attempted.

Preparing the File Structure

First, download the latest version [http://releases.ubuntu.com/] of the Ubuntu variant of choice. Extract the iso file using your compression software of choice [http://alternativeto.net/tag/file-compression/]. All of the iso file's contents will be required, as this entire directory is going to be mounted to a client whenever it is booted to the Ubuntu environment.

Your directory structure may look like this:

```
# ls -alFh ubuntu-14.04.1-desktop-amd64
total 2.6M
drwxrwxrwx 13 root root 4.0K Jan 30 16:21 ./
drwxrwxrwx  6 root root 4.0K Jan 30 17:30 ../
-rwxrwxrwx  1 root root  134 Jul 22  2014 autorun.inf*
drwxrwxrwx  3 root root 4.0K Jan 30 16:20 boot/
drwxrwxrwx  2 root root 4.0K Jan 30 16:21 [BOOT]/
drwxrwxrwx  2 root root 4.0K Jan 30 16:21 casper/
drwxrwxrwx  2 root root 4.0K Jan 30 16:20 .disk/
drwxrwxrwx  3 root root 4.0K Jan 30 16:21 dists/
drwxrwxrwx  3 root root 4.0K Jan 30 16:20 EFI/
drwxrwxrwx  2 root root 4.0K Jan 30 16:21 install/
drwxrwxrwx  2 root root 4.0K Jan 30 16:21 isolinux/
-rwxrwxrwx  1 root root  21K Jul 22  2014 md5sum.txt*
drwxrwxrwx  2 root root 4.0K Jan 30 16:21 pics/
drwxrwxrwx  4 root root 4.0K Jan 30 16:21 pool/
drwxrwxrwx  2 root root 4.0K Jan 30 16:21 preseed/
-rwxrwxrwx  1 root root  231 Jul 22  2014 README.diskdefines*
-rwxrwxrwx  1 root root 2.5M Apr 14  2014 wubi.exe*
```

This structure will be copied into a NFS location later.

Preparing the NFS Share

Installing the Package

The NFS daemon package must be installed according to your distribution's instructions.

- Ubuntu [https://help.ubuntu.com/community/SettingUpNFSHowTo#Installation_and_Configuration]
- Debian [https://wiki.debian.org/NFSServerSetup]
- Fedora [http://fedoraproject.org/wiki/Administration_Guide_Draft/NFS]
- ArchLinux [https://wiki.archlinux.org/index.php/NFS]
- Follow your distribution's documentation

Configuring the Share

Edit the `/etc/exports` file using your editor of choice [http://alternativeto.net/tag/text-editor/]. Apply the following line, substituting in the path where you will host your Ubuntu file structure

```
/path/to/ubuntu/files     *(ro,sync,no_wdelay,insecure_locks,no_root_squash,insecure)
```

Note: The * prefix before those options specifies that this share is publicly accessible with access to the network resource. This can be changed [http://linux.die.net/man/5/exports] to fit any security requirements.

Restarting the Daemon

In most cases, after you have edited the /etc/exports file you will be required to restart the daemon for the change to take effect. Follow your distribution's documentation.

Putting Everything Together

Copy the Structure

Copy your extracted Ubuntu iso contents from earlier into the path specified in your server's /etc/exports file.

Note: This can be put into a subfolder. One export applies to all subfolders of the export.

iPXE Booting

For the boot to occur the following items must be supplied:

- Kernel
- Initial ramdisk
- NFS options
- Client IP information

Note: This configuration assumes the iPXE boot file you are using has the DOWNLOAD_PROTO_NFS option enabled. If it doesn't, you will have to change the kernel and initrd lines to boot over a protocol that is supported in your environment. Either this, or you could rebuild your iPXE boot file.

amd64

```
#!ipxe

set server_ip 192.168.100.1
set nfs_path /srv/nfs/ubuntu-14.04.1-desktop-amd64
kernel nfs://${server_ip}${nfs_path}/casper/vmlinuz.efi || read void
initrd nfs://${server_ip}${nfs_path}/casper/initrd.lz || read void
imgargs vmlinuz.efi root=/dev/nfs boot=casper netboot=nfs nfsroot=${server_ip}:${nfs_path} ip=dhcp splash quiet -- || read void
boot || read void
```

i386

```
#!ipxe

set server_ip 192.168.100.1
set nfs_path /srv/nfs/ubuntu-14.04.1-desktop-i386
kernel nfs://${server_ip}${nfs_path}/casper/vmlinuz || read void
initrd nfs://${server_ip}${nfs_path}/casper/initrd.lz || read void
imgargs vmlinuz root=/dev/nfs boot=casper netboot=nfs nfsroot=${server_ip}:${nfs_path} ip=dhcp splash quiet -- || read void
boot || read void
```

appnote/ubuntu_live.txt · Last modified: 2015/02/24 01:59 by jamesaepp