

Linux Project Notes Red Hat 8.0/9.0

The following is a collection of notes that I have created to help my sanity as I have been exploring RedHat Linux versions 8.0 and 9.0. I've tried to group my thoughts in a relatively logical manner so that they may be of some use to others.

All information contained within this document is as-is and without warantee. It is accurate to the best of my knowledge, but some of it makes certain assumptions that may be specific only to my situation, so take them with a grain of salt.

If you find corrections or would like to contribute to this document, email them to zclobes@swbell.net.

-Zack

Important System Files

/etc/fstab – A listing of file mounts that are mounted during the boot up process.

/etc/rc.d/rc.local – This file executes at boot up time. It is meant to kick off configuration/initialization scripts for the host.

/etc/crontab – A listing of jobs to be kicked off at specific types by the cron daemon.

/etc/X11/XF86Config – Contains the settings for the X-Windows configuration. Includes video settings, mouse settings, etc..

Important Networking Files

/etc/hosts – Contains IP to host name mappings for non-DNS enabled clients (or for static mappings not found in DNS)

/etc/sysconfig/network-scripts/ifcfg-ethX – Contains the basic configuration information that corresponds to a Ethernet adapter. Normally, ifcfg-eth0 is always defined. Some of the parameters that can be defined are:

DEVICE – The physical device being used.

BOOTPROTO – Static vs. DHCP

BROADCAST – The TCP/IP broadcast address for the subnet.

IPADDR – The IP address to use for a static configuration.

NETMASK – The subnet mask.

NETWORK – The basic network address of the network (192.168.0.0)

ONBOOT – Whether or not the adapter is enabled at bootup.

/etc/sysconfig/network – Enables networking and sets the hostname. Parameters include:

NETWORKING – Normally set to YES.

HOSTNAME – The fully qualified hostname of the machine.

GATEWAYDEV – The default gateway adapter???

/etc/modules.conf – Configures the physical Ethernet adapters as aliases to eth0, etc. For example, to configure a Linksys 10/100 card (tulip driver) to use eth0, add the line 'alias eth0 tulip' to the file. For some drivers, especially ISA cards, you may also have configuration options in this file such as 'options eepr io=0x300, irq=5'.

Configuring IPTABLES as a Firewall

Various scripts are available on the web for simple firewall setups using IPTABLES as the filtering mechanism. The firewall script is placed in `/etc/rc.d/rc.firewall` and made to be executable by using the `'chmod a+x rc.firewall'` command.

You also need to add the `rc.firewall` script to the `rc.local` file so that the IPTABLES configuration actually loads at boot up time. To do this, just add the following lines to the end of `rc.local`:

```
echo "Loading the rc.firewall ruleset..."
/etc/rc.d/rc.firewall
```

Configuring the GRUB Bootloader

Generally the GRUB bootloader comes pre-configured and ready to go, however sometimes guesses incorrectly or the partition changes. In this case, you can press 'e' to edit the OS boot commands and manually modify the command line. There is no way to actually save the command, however. In order to permanently change the command, boot into Linux and modify the `grub.conf` in the `/boot` partition.

Booting into Single User Mode – In case of login problems, you can boot the Linux OS into a single-user mode which will leave you at a `#` (root).

To enter Single User Mode, boot the PC to the GRUB boot loader and press 'e' to edit the Linux boot commands. Once you're in edit mode, find the line that begins with 'kernel' and add the word 'single' to the end of it (leave a space between the end of the line and the word 'single'). Then press Enter to store the changes (this boot only) and press 'b' to boot into the Single User Mode. When done fixing the problem, type 'exit' and it will continue booting into a normal Multi-User Mode.

Daemons – dhcpd

The main configuration file is in `/etc/dhcpd.conf`. A sample configuration is as follows:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
    option routers      192.168.0.1;
    option subnet-mask 255.255.255.0;
    option domain-name-servers 192.168.0.2;
    option domain-name  "acme.com";
    range      192.168.0.10 192.168.0.50;
}
```

To start the service, type `service dhcpd start`

Daemons – httpd

The primary configuration file is located `/etc/httpd/conf/httpd.conf` and contains all sorts of configuration variables for the web server. Some of the more important parameters are:

- `documentroot` – indicates where the HTML/script files are stored
- `scriptalias` – Sets the cgi-bin directory to a location on the disk.

There's really not a whole lot to configure on the server. Just be careful – many times a parameter must be changes multiple times in the document. The changes are usually lump together though.

To install PHP or Perl, the easiest to to let the Red Hat RPM's take care of it in the beginning. It is very complex to try to add them in manually after-the-fact.

After changing the configuration file, you will need to restart the service by typing `'service httpd restart'`.

Daemons – nfs

Test for current status by typing 'service nfs status' and 'rpcinfo -p' at the server's console to determine the current status of the nfs daemon.

The main configuration file for nfs shares is /etc/exports. The file format of the configuration file is the directory name, then the host that is allowed access to that directory, followed by the (rw) permissions for that user. BE SURE NOT TO PUT SPACES BETWEEN THE USERNAME AND THE PERMISSIONS. For example, to share a directory named /home/router to the host router.acme.com with read and write access, add the following line to the /etc/exports file:

```
/home/router router.acme.com(rw)
```

You will then need to restart the nfs service with the 'service nfs restart' command. You should then be able to mount the share on the remote host (router.acme.com in this case) by typing something like 'mount -t nfs zeus:/home/router /mnt/zeus'.

Daemons – smb

Configuration

The main configuration file is /etc/samba/smb.conf. The following are the main parameters to look for in customizing a installation.

- Workgroup – NT domain or workgroup
- hosts allow – Security permissions based on IP address.
- Printcap name
- load printers – These setup Samba to load printers during the startup.
- Password server – Specifies an NT domain controller.
- Encrypt passwords
- smb passwd file – These indicate how Samba treats passwords

Setting up Users and Passwords

Begin by creating a new Samba password file by using the mk smbpasswd.sh shell script.

```
cat /etc/passwd | mk smbpasswd.sh > /  
etc/samba/smbpasswd  
(All on one line)
```

The change the permissions on the /etc/samba/smbpasswd file by typing 'chmod 600 /etc/samba/smbpasswd'

You then have to add each user's account information to the smbpasswd file by using the smbpasswd program. For example, to add a user JonD with password abc123, type 'smbpasswd JonD' and then type/retype the password. You can also do that in one string by using 'smbpasswd JonD abc123'.

You may need to restart the smb service by typing 'service smb restart' at this point.

Creating File Shares

At the end of the smb.conf file, you will find a variety of examples of various share types. To do a basic Users directory share, add the following to the end of the smb.conf file.

```
[users]  
comment = User Files  
path = /home  
public = yes  
writable = yes  
printable = no  
write list = @staff
```

I really don't know if the printable parameter applies to a file share. I also have no idea what the write list parameter is.

Creating Printer Shares

Printers are configured the same way as the file shares. At the end of the smb.conf file, add

the following:

```
[printers]
comment = All Printers
path = /var/spool/samba
browsable = no
guest ok = yes
writable = no
printable = yes
```

Again, I'm not quite sure if the writable applies since this is configuration for printers. Once this is configured, the next step is to create the spools to print to the appropriate printer ports and drivers.

Somehow we have to create the printer spools but I can't remember how I did it.

Daemons– sshd

There is very little configuration for the ssh daemon. Try starting it with `service sshd start` and see what happens. You can use a Windows piece of software called OpenSSH or Putty to connect to the remote Linux box via ssh.

Installing RedHat 9.0

CD-ROM

The simplest method for installing RedHat 9.0 is to either buy the boxed disks, or download the ISO disk images and burn them to a blank CD-ROM. You can then insert the first disk into the CD-ROM drive and boot the system. You will be taken directly into the Anaconda installer.

NFS

If you already have a Linux system on the local network, and are needing to install Linux on a second box, you can use an NFS install. This is particularly helpful if you don't have a CD-ROM drive in the second box.

First you must download the ISO images from RedHat's website and put them on the existing Linux server's hard drive. That directory must be shared using NFS by properly configuring the `/etc/exports` directory and starting the NFS daemons.

Next you will need a set of boot floppies that will load the NIC drivers and the Anaconda installer. These can be found on the first CD of the installation disks.

Finally, boot the target PC with the boot disk and when prompted, insert the NIC drivers disk to get the network up and running. Eventually, you will wind up at a menu that gives you the option of, among others, loading the distribution from an NFS share. Select that option and provide the installer with the name and mount point of the NFS server on the network.

Installation will proceed as normal from here on out.

Installing Software

RPMS

The rpm utility is very useful to install pre-compiled binaries in RedHat Linux. Generally, you simply download the .rpm file from the Internet, and type `rpm -ivh package_name.rpm` and it will install the RPM into the system.

I had the rpm utility lock up on me at one point where every time I tried to query the RPM database, the command would lock up on me and I had to reboot the system. Per an article on www.experts-exchange.com, I deleted some temp files in `/var/lib/rpm`. The temp files begin with two underscores.

apt-get

apt-get is a utility that makes RPM packages even easier. The apt-get utility will automatically download the requested RPM package, determine if there are any dependencies that need to be installed with it, and will install them automatically.

Compiling from Source

Compiling from source is usually the most difficult and error-prone. Usually, you must download the tar-ball, and then begin reading any documentation you can find. It usually takes multiple steps to configure, compile, and install the software.

Perl Modules

Most add-on modules for Perl scripts can be obtained from the CPAN repository. To install a module, type `perl -MCPAN -e 'install Chocolate::Belgian'` where `Chocolate::Belgian` is the name of the module to be installed.

Mounting File Systems

Floppy Drives

This is done by using the mount command with a set of parameters. To mount the 3.5" floppy drive which is normally known as `fd0`, type `'mount /dev/fd0 /mnt/floppy'`. This assumes that the floppy device is in fact `fd0` and that the directories `/mnt/floppy` exist. Sometimes you will have to specify the file system that is contained on the floppy disk before it will let you mount it.

CD-Rom Drives

Mounting the CD-Rom drive is about the same as the floppy. Red Hat provides us with an alias to the CD-Rom device called `/dev/cdrom`. To mount it, type `'mount /dev/cdrom /mnt/cdrom'`. Again, sometimes it does require you to specify the file system on the CD-Rom.

Compact Flash Drives

Mounting a compact flash drive is identical to mounting a partition on a hard disk – you just have to figure out which drive represents the CF card. The PCMCIA card reader on my Dell Inspiron 4100 laptop initializes itself as `hde1`, so to mount it, type `'mount /mnt/cf /dev/hde1'`

NFS File Systems

Once the NFS server is setup (see Daemons – `nfsd`), you mount the remote file system on the client by typing `'mount -t nfs servername:/full/path/to/folder /mnt/theservermount type nfs'`. For example, to mount the server `zeus:/home/router` on a remote client to the folder `/mnt/zeus`, type `'mount -t nfs zeus:/home/router /mnt/zeus'`.

Windows (SAMBA) Shares

Mounting a Samba share on a Windows system is similar to the NFS shares, except I think you need to have the `smb` daemon running and you must specify the servers a little bit different. For example, to connect to a Windows server `WinServ` with a file share named `Users`, a logon name `JonD`, and a password of `abc123`, you would type `'mount -t smbfs -o username=JonD,password=abc123 //WinServ/Users /mnt/Users'`. This would put the mount in `/mnt/Users` for the Linux host to then access.

Notes on Disk Drives

Drive naming scheme

`hda-hdh` = IDE disks with partitions 1-9 on each
`sda-sdg` = SCSI disks with partitions 1-8 on each
`scd` = SCSI CD-ROM with partitions 0-7

Extra names can be added to the list by typing `'makedev /dev/sdx'` where `x` is the new drive letter.

Note: SCSI drives may rename themselves as new devices are added to the chain. For this reason, it is always best to install and add SCSI drives in numeric order on the SCSI bus with no skips in the sequence.

Adding a new disk

1. Physically install the disk drive.
2. Install the driver for the disk/controller. If this is a fairly standard device (IDE/SCSI), the driver may be automatically installed.
3. Define one or more partitions using the fdisk or Disk Druid utilities.
4. Make the file system using the mkfs command.
5. Add the new partition to the /etc/fstab file so that the new drive automatically mounts a boot-up.
6. Mount the file system.

Drive Partitions

/ - The root partition is where all systems are mounted.

Swap partition – Virtual RAM disk. Should be at least as big as the physical RAM installed in the system so that a memory dump could be done to it.

/bin – Compiled binary files that are available to average (non-root) users.

/boot – Files to get the system booted including the system kernel

/dev – Everything in Linux is a “file” and this is a directory full of devices such as disk drives, I/O ports, etc.

/etc – Pronounced “et-see” and stands for Extended Tool Chest. This directory contains most of the system/daemon configuration files.

/home – Location of the user's home directories (except the root user)

/lib – Shared libraries of system functions.

/lost+found – Contains any orphaned files from the journaling file system.

/mnt – A directory where you normally mount foreign file systems (CD-ROM's, NFS shares, etc).

/proc – Running processes.

/root – Location of the home directory for the root user.

/sbin – System binaries normally accessible to administrators such as root.

/tmp – Temporary file space (not to be confused with the swap space).

/usr – Location of most software on the system.

/usr/local – Location of programs to be installed separate from the RedHat software.

/usr/src – Location of the source code for the various applications..

/var - “Variable” files that tend to change such as logs, webserver files, cache's, etc.

Misc. Commands

adduser – Creates a new user.

addgroup – Creates a new group.

cat – Can concatenate two strings together. More commonly used to view the contents of a file from a terminal session (e.g. 'cat /etc/hosts')

cd – Change directory. Make sure you always use a space (it's 'cd .' and not 'cd..')

chgrp – Changes the group on a file.

chmod – Changes the permissions on a file. The easiest method is to use chmod {u|g|a} {+|-|=} {rwx}. For example, to set execute permissions for all users on a file named myscript, type 'chmod a+x myscript'.

chown – Changes the owner of a file.

cp – Copy a file.

dd – Installs a disk image (*.img) file to a floppy disk. For example, `dd if=boot.img of=/dev/fd0 bs=1440k` will write the image `boot.img` to the floppy disk drive. The command is similar to the MS-DOS utility `rawrite`.

df – Displays the system disk usage. Often used with the `df -h` flag to readout in human-readable (kB, MB, GB) terms.

du – Displays the disk usage for a specified directory. Use with the `-sh` flags for a human-readable summary. For example, to check the usage of the `/var` directory, type `du -sh /var`.

fdisk – Partitions a disk. Typing `fdisk /dev/hda` will allow you to partition the first IDE disk drive so that you may use it. Once in the `fdisk` program, type `'m'` to get a listing of available commands.

find – Used to search for files throughout the file system. To search for a file anywhere on the system, `cd` to the root (`/`). Then do a `find -name a_lost_filename.abc` and sit back and relax.

halt – Initiates the shutdown process.

ifconfig – Supplies configuration information about the Ethernet adapters such as their MAC address, DHCP lease address, DNS configuration, etc.

ifup and **ifdown** – Brings an interface up and down. `'ifup eth0'`.

ls – Lists directories on the disk. `'ls -l'` will list the directories and include file sizes, permissions, etc. The `ls -a` flag will include hidden files and directories.

ll – A RedHat (and possibly others) shortcut for the command `ls -l` to see file permissions, sizes, etc.

mkdir – Make a directory.

mkfs – Creates the actual file system on a partition. This is the Linux equivalent to formatting a disk. Be sure to use `fdisk` on the drive first to create the partitions. For example, `'mkfs -t ext2 /dev/sda1'` will format the first partition on the SCSI disk A with an `ext2` type filesystem.

moduser – Modifies a user.

mount – Mounts a device or file system into a folder. See [Mounting File Systems](#) elsewhere in the document. Just typing `mount` without any parameters will show the current mountings for the file system. See also `umount`.

pwd – Prints the Present Working Directory to the screen.

rm – Remove directory.

rpcinfo – Typing `rpcinfo -p` will provide a list of currently running RPC ports open on the local computer. You can also run `'rpcinfo -p remotehost.domain'` to view the RPC's on a remote host.

service – Can be used to start, stop, restart, and query various services (daemons). To start the `sshd` daemon, type `'service sshd start'`.

startx – Launches X-Windows from a terminal session.

top – A command line utility for monitoring the CPU/memory utilization of the machine. For CPU monitoring, the three numbers at the top-right of the screen indicate how many processes are being queued up at a given time. For acceptable performance, those should be well below 1.00 most of the time.

umount – Unmounts a file system. Just type umount and the name of the mount.

vi – A simple terminal-based text editor that is available on virtually *nix distributions. You should become familiar with at least the basics of how to get around in this editor – it will make your system administration much easier.

whereis – Does a query across the filesystem for a given executable file/folder. Note, this does not search non-executable files and sometimes doesn't seem very predictable with its results. I suspect that it only queries against executables and directories that lie in the system's path statement or in the immediate vicinity of the current working directory.

Miscellaneous Linux Commands That Emulate MS-DOS Disk Utilities

mformat – Formats a floppy disk in MS-DOS format.

mlabel – Labels the floppy disk.

mcd – Changes directories.

mdir – The dir command.

mtype – Displays the contents of a file (type command in DOS).

mcopy – Copies a file between floppies and Linux OS.

mdel – Deletes a file on the floppy.

mren – Renames a file on the floppy disk.

mmd – Makes a new directory on the floppy.

mrdd – Removes a directory on the floppy.

mattrib – Changes the MS-DOS file attributes.

Common Programs/Utilities

chkconfig – Lists and configures services to startup at boot time. See also ntsysv and serviceconf.

pico – A small text editor that comes with the Pine email client

nano – A small text editor similar to pico, but comes as an independent installation. I found this off the web somewhere.

ntsysv – Configures services to start at boot up time. See also chkconfig and serviceconf.

serviceconf – Configures services to start at boot up time. See also chkconfig and ntsysv.