IBM
Advanced Technical Skills

Linux For System z
Installation Workshop - RedHat Enterprise Linux

2011
Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix D, "Special notices".

First Edition (June 2011)

This edition applies to RedHat Enterprise Linux 6.1 for System z

This document created or updated on June 2011.

Comments may be addressed to:
IBM Corporation, Attention: Richard Lewis
Building 183, Room 2N79
800 N. Frederick Ave.
Gaithersburg, MD 20879

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2000. All rights reserved.
Note to U.S Government Users – Documentation related to restricted rights – Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.
Contents

Chapter 1. Introduction ......................................................... 5
  1.1 Lab Overview ............................................................. 5
  1.2 Pace ................................................................. 7

Chapter 2. Installing Linux For System z ............................ 9
  2.1 Prepare Userid ......................................................... 9
  2.2 IPL Linux For System z ............................................... 9
    2.2.1 Punch files to VM reader .................................... 9
    2.2.2 Modify REDHAT CONF File .................................. 10
    2.2.3 IPL VM reader .................................................. 11
  2.3 Use loader To Install System ...................................... 12
    2.3.1 SSH To Your Guest ........................................... 12
    2.3.2 Installer Setup ............................................... 15
    2.3.3 Start VNC On Your Workstation ............................. 18
  2.4 Checkpoint 1 ............................................................ 56

Chapter 3. Post Install Tasks .............................................. 57
  3.1 SSH To Your Guest .................................................... 57
  3.2 Disable Firewall and SELinux ....................................... 60
  3.3 Build Package .......................................................... 66

Chapter 4. Add Additional FCP Disks for Multipath LVM ............. 69
  4.1 Add Additional FCP attached SCSI LUNs .......................... 69
  4.2 Harden Dynamic Changes ............................................. 72
  4.3 Obtain UUIDs of New FCP LUNs ..................................... 74
  4.4 Modify /etc/lvm.conf ............................................... 78
  4.5 Create Physical Volumes ............................................. 79
  4.6 Create lvm Volume Group ............................................ 79
  4.7 Create lvm Logical Volume .......................................... 80
  4.8 Create File System on Logical Volume ............................ 80
  4.9 Configure Necessary Boot Scripts .................................. 81
  4.10 Optional - Test Multipath and LVM I/O Distribution .......... 83
  4.11 Checkpoint ............................................................ 89

Chapter 5. Basic Linux For System z Administration .................. 91
  5.1 Setup z/VM Virtual Disk Swap Device ............................. 91
  5.2 Networking FYI ....................................................... 91
  5.3 Add A New User ......................................................... 93
    5.3.1 Add a New User Using GUI Interface ....................... 94
    5.3.2 Add a New User Using Command Line ..................... 101
  5.4 Update Message Of The Day File ................................... 102
5.5 Monitoring Disk Usage ............................................. 102
5.6 Additional Commands ............................................ 104
5.7 If You Have Extra Time ......................................... 104

Chapter 6. Configuring IPTABLES ................................. 105
6.1 Introduction ....................................................... 105
6.2 Verify IPTABLES Version ....................................... 105
6.3 Configuring iptables ............................................. 107
   6.3.1 Check Netfilter Operation ................................ 108
6.4 Checkpoint 2 ..................................................... 108
6.5 If You Have Extra Time ......................................... 108

Chapter 7. Configure BIND-9 ....................................... 111
7.1 Introduction ....................................................... 111
7.2 Customize Configuration Files ................................. 111
7.3 Start NAMED And Test ......................................... 117
7.4 Configure NAMED To Start Automatically ................... 117
7.5 Test your change ................................................ 123
7.6 If You Have Extra Time ......................................... 127

Chapter 8. Configuring Samba ..................................... 129
8.1 Introduction ....................................................... 129
8.2 Customize Samba ................................................ 129
8.3 Test Use Of Samba From Linux ............................... 130
8.4 Test Use Of Samba From Workstation ......................... 131
8.5 Additional Configuration Programs ......................... 132
   8.5.1 Test Tools Share ......................................... 132
8.6 Setup Samba To Automatically Start At Boot ................ 132
8.7 Checkpoint 3 ..................................................... 137

Chapter 9. Customizing Apache ................................. 139
9.1 Introduction ....................................................... 139
9.2 Customize Apache Configuration File ....................... 139
9.3 Run Apache ...................................................... 140
9.4 Change Content ................................................ 141
9.5 Integrate NFS Mount Into Linux Startup .................... 144
9.6 Test System Changes .......................................... 145
9.7 Apache Extra Credit ............................................ 149

Chapter 10. Configuring KDE with VNC ......................... 151
10.1 Introduction ..................................................... 151
10.2 Configure KDM ................................................ 151
<table>
<thead>
<tr>
<th>Appendix A. VIM Reference</th>
<th>157</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix B. ED Reference</td>
<td>159</td>
</tr>
<tr>
<td>Appendix C. FCP SCSI LUN ID Table</td>
<td>161</td>
</tr>
<tr>
<td>Appendix D. Special notices</td>
<td>163</td>
</tr>
</tbody>
</table>
Workshop
Chapter 1. Introduction

1.1 Lab Overview

This lab is intended to assist you in developing familiarity with and skill in the installation and customizing of Linux for System z. It is a self paced, self guided lab. However, please do not hesitate to ask questions when you do not understand the instructions, or would like to know why you are doing something. Upon completion of this lab, you should be able to return to your shop, obtain the Linux for System z components from RedHat, and install Linux for System z on your system.

Every effort has been made to make this lab such that you can duplicate the results in your own shop. The Linux software, and file system objects you will use, are the same ones that you can obtain from RedHat. However, this is still a workshop environment, so there have been some slight deviations. For example, you will not need to format devices. This is to eliminate the wait while 30 teams simultaneously format devices.

Since our processor does not have sufficient resources for 30 lpars, this entire lab will be done using z/VM. However, it must be emphasized that the same process can be followed (with some exceptions) to install Linux in an LPAR. One change is that in an LPAR you will not be able to use a VM reader for starting the installation process. The LPAR alternative is to use the Hardware Management Console (HMC) to load the Linux kernel, parm file and Ram Disk from the distribution CD-ROM. However, once Linux is IPLed, much of the later procedures in this book can be followed identically when installing Linux in an LPAR.

Each team will have a virtual machine to use. The userids are LINLABnn (where nn corresponds to your team number). The passwords for the virtual machines are the same as the userid. Your windows workstation should already have a PCOM session defined for 3270 access to the VM system. At this point, go ahead and start that session, and logon to your LINLABnn userid. Each virtual machine will have a virtual network interface card connected to a z/VM QDIO Guest Lan. The IP addresses you should use are 9.82.56.(91-120). The IP address of your gateway router will be 9.82.56.1. So for example, if you are team 1, your IP address will be 9.82.56.91. If you are team 10, your IP address will be 9.82.56.100. To summarize, use the following formula to determine your Linux IP address.

9.82.56.xxx where xxx = team_number + 90
The z/VM system, and services on that system will be represented by the IP address:

9.82.56.1

One thing you need to be aware of as you proceed through the lab is that CASE is VERY important. When you are following the instructions and entering commands, please notice and follow the case of the printed instructions. In addition, there are many figures included in the text to illustrate what sort of response you should expect to receive from tasks and commands. Please do not use the figures as the source of information you should type. Rather, use the text as your source of input. Text in bold print will typically be what you should type at the console, or an ssh session.

Each Linux for System z virtual machine has the following identical directory entry:

```
USER LINLAB01 LINLAB01 512M 512M G
MACHINE ESA
CPU 00 BASE CPUID 111101
LOADDEV PORT 50050763031BC08D
LOADDEV LUN 003B000000000000
IPL CMS
IUCV ANY PRIORITY MSGLIMIT 2000
IUCV ALLOW
OPTION QUICKDSP
CONSOLE 009 3215
SPOOL 00C 2540 READER *
SPOOL 00D 2540 PUNCH A
SPOOL 00E 1403 A
DED A00 A00
DED B00 B00
LINK MAINT 0190 0190 RR
LINK MAINT 019D 019D RR
LINK MAINT 019E 019E RR
LINK LINLABPL 68E A628 RR
LINK LINLABPL A532 A532 RR
LINK LINLABPL A507 A507 RR
LINK TCPMAINT 592 592 RR
MDISK 191 3390 02551 00050 CLAS01 MR ALL ALL
MDISK 200 3390 00001 01500 CLAS02 MR ALL ALL
MDISK 202 FB-512 V-DISK 20000 MR ALL ALL
MDISK 300 3390 02601 00020 CLAS01 MR ALL ALL
MDISK 291 3390 02621 00002 CLAS01 MR ALL ALL
```

Before you begin focusing on the trees, it will be useful to take a look at the forest. In this lab you will be completing the following tasks:
- Install RedHat Enterprise Linux 6.1 for System z
- Add additional SCSI LUNs to the system and configure as multipath LVM
- Perform some system administration tasks
- Add new user
- Execute Linux commands to monitor disk usage
- Configure IPTables (Linux firewall)
- Configure BIND-9, creating a caching only name server
- Configure Samba for file sharing
- Configure Apache
- Use new cmsfs-fuse support to access VM minidisk, and make system changes to automate mounting of the minidisk at system boot
- Configure KDM for use with VNC to provide the KDE desktop within a vncviewer session.

When you are all done, you will have a virtual machine running Linux for System z, Samba, Bind, Apache, the Linux firewall code, and with a vnc accessible KDE desktop, that can be autologged as a standalone service machine.

1.2 Pace

It is very likely that you will not be able to complete all of the chapters in this workbook during the 3 hour period available for the workshop. Rather than speed through the chapters you are encouraged to select the chapters that are of most interest to you, and perform the work at a pace that allows you to consider what you are doing. Following chapter 3, each of the remaining chapters can be done independently of the other chapters. You can proceed directly to the chapters you most want experience with, and then go back to other chapters if time allows.

Within the workshop notes, some of the commands to be entered do not fit entirely on a single printed line. In this case, the book software split the command where there was a space between words. Thus when you see a command split across two lines, you can safely assume that a space should be entered between words on the first line, and words continued on the second line.

If there are no questions at this time, you are ready to begin!
Chapter 2. Installing Linux For System z

2.1 Prepare Userid

Logon to LINLABnn where nn corresponds to your team number 01-30

- In an ordinary installation you would format your devices. In the interest of time, this has already been done for you.
  - Your A-disk (191) has been formatted and a PROFILE EXEC supplied
  - Your 202 vdisk (Linux swap area) will be formatted from Linux

- Normally you would insert the RedHat DVD into a workstation DVD drive and ftp the following files to your VM userid, with a record format of F, an LRECL of 80 and a transfer mode of binary. This task has already been done for you. You will find the following files on your A-disk (191)
  - KERNEL IMG (the Linux kernel from RedHat built for IPL from a VM Reader)
  - INITRD IMG (the initial ram disk image)
  - Note: These two files will be punched to the VM Reader which explains why they are blocked F 80. If you were uploading these files to VM from your workstation, you would want to specify 'quote site fixrecfm 80' after specifying binary.
  - GENERIC PRM (the generic parm file from RedHat for use with the ram disk system)
  - REDHAT CONF (extended parm file created to provide installation parameters)

- You will also see the following support files on your A-disk:
  - REDHAT EXEC (A simple exec to spool your punch to your rdr, punch the kernel, parmfile and ramdisk, then IPL your rdr)
  - PROFILE EXEC (Set basic pf keys, and disable terminal chardel and linend functions)

2.2 IPL Linux For System z

2.2.1 Punch files to VM reader

Enter: SP PUN * CLOSE
2.2.2 Modify REDHAT CONF File

The REDHAT CONF file needs to be customized for your particular guest by specifying the particular host name of your guest, the specific IP address of your guest, and the Logical Unit Number (LUN) of the FCP attached SCSI disk you will use for your root file system.

Use the CMS editor XEDIT to make these changes.

Enter: XEDIT REDHAT CONF A

First, change the line HOSTNAME= from linlabnn.endicott.ibm.com to your particular hostname (i.e. change nn to your team number). Move your cursor over the nn on the HOSTNAME line and change the nn to your team number.

Type: linlabnn.endicott.ibm.com (where nn is your team number)

Next, change the LUN number in the FCP_1 = line from 0x00nn000000000000 replacing the nn with the last two digits of the value in the "Install LUN" column corresponding to your Userid in Appendix C.

Type: 0x00nn000000000000 (where nn is the Install LUN for your guest from Appendix C)

Next, change the IPADDR= from 9.82.56.xxx to 9.82.56.(team number + 90).

Type: 9.82.56.xxx (where xxx is your team number + 90)

At this point, your file should look similar to the following figure:
**Figure 1.**

Move your cursor to the command line at the top of the file and:

Enter:  **FILE**

The system will respond by closing the XEDIT session and displaying the CMS Ready prompt.

### 2.2.3 IPL VM reader

Enter:  **IPL 00C CLEAR**

Since you used an extended parameter file (REDHAT CONF), you will not need to answer any of the network configuration questions, or questions related to the installation media that you would normally need to enter at this point during the installation. Instead you should see a message directing you to use telnet or ssh to connect to your guest, similar to the following figure.
2.3 Use loader To Install System

2.3.1 SSH To Your Guest

Each workstation has the freeware windows telnet/ssh client PuTTY installed. Click on the PuTTY icon to begin a session. The system will respond with a window similar to the following.
Figure 3.

Click to highlight: **LINLAB**

Click: **Load**

The system will respond with a window similar to the following:
Click your mouse pointer in the text box named "Host Name (or IP address)". Type: 9.82.56.(91-120)

Click: Open

The RedHat ramdisk file system is configured so that no password is required for the install userid. When prompted to enter a userid:

Enter: install
The system will respond by establishing a new session for you, and automatically starting the installation program in the shell session. You will receive a screen similar to the following figure.

Note: to navigate the text based full screen menu items, use the keyboard arrow keys to move through the list of choices, then use the tab key to move to a particular text image button. Pressing the enter key will cause the selected button to be activated.

2.3.2 Installer Setup

![Figure 5.](image)

Select: **English**

Tab to: **OK**

Press: **ENTER**

The system will respond with a panel similar to the following:
Figure 6.

Select **NFS directory**

Tab to: **OK**

Press: **ENTER**

After pressing ENTER, the system will respond with a panel similar to the following.
Figure 7.

Select: **NFS server name:**
Type: 9.82.56.242
Tab to: **Red Hat directory:**
Type: /isovol/rhel61
Tab to: **OK**
Press: **ENTER**

The system will respond by ending the full screen text application, and printing messages that indicate Anaconda (the Red Hat installer) is starting. You will finally be prompted to connect to your system using VNC as in the following figure:
2.3.3 Start VNC On Your Workstation

VNC stands for Virtual Network Computing. It is, in essence, a remote display system which allows you to view a computing 'desktop' environment not only on the machine where it is running, but from anywhere on the Internet and from a wide variety of machine architectures.

VNCVIEWER is a client for VNC. It is used to display the X environment running on your Linux virtual machine. VNCVIEWER is different from having an X Server on your workstation in that no state information is stored on the workstation, and it is platform independent. VNCVIEWER is available in binary format for download from the Internet. We have already installed the product on your Windows workstation and created an icon for it on the desktop. Locate the vncviewer icon on your desktop and double click the icon. You will receive a window similar to the following:
At this point a VNC session will be started with the server running within your Linux virtual machine. Rather than seeing a generic desktop, you will be placed immediately into the Anaconda application. You should receive a panel similar to the following.

Figure 9.

Type: 9.82.56.(91-120):1

Click: ENTER
The system will respond with a pop-up window requesting information about the type of devices that will be used during the installation, similar to the following:
Since we will be installing on FCP attached SCSI LUNs leave the Specialized Storage Devices radio button selected, and:

Click: Next

The system will respond with a window allowing you to select your devices similar to the following:
Figure 12.

Click: **box next to IBM S390 DASD drive**

Click: **Other SAN Devices (on tabs at top)**

The system will respond with a window showing the FCP SCSI devices that are available (from the REDHAT CONF file on your A-disk) similar to the following figure:
Figure 13.

Select the single device available by clicking in the small box next to the device name string and:

Click: **Next**

The system will respond with a pop-up warning window indicating that the DASD device needs to be reinitialized which may cause data loss. This warning is issued because the DASD selected is a z/VM virtual disk that has not been previously formatted or used.
Click: **Yes, discard any data**

The system will respond with a panel prompting for the network host name to use for your new Linux system, similar to the following figure.
Since the hostname was entered in the REDHAT CONF file earlier, the value from that file is already entered.

Click: **Next**

The system will respond with a panel requesting the time zone to be used, similar to the following.
Click: **Next**

The System will respond with a window requesting a password to assign to the user root after the installation completes, similar to the following figure:
Enter: linlabnn (in the Root Password box)
Enter: linlabnn (in the Confirm box)
Click: Next

The system will respond with a pop-up window indicating that the password supplied is considered week, similar to the following figure.
The system will respond with a panel requesting a partitioning scheme for the installation disk, similar to the following figure:

Click: **Use Anyway**

The system will respond with a panel requesting a partitioning scheme for the installation disk, similar to the following figure:
Figure 19.

Click: **Create Custom Layout (radio button)**

Click: **Next**

The system will respond by displaying a window where you can identify the device to install the root file system onto, similar to the figure below.
Figure 20.

Click: **to Highlight the FCP LUN**

Click: **on the right pointing arrow**

The system will respond by redisplaying the window with the FCP LUN moved to the Install Target Devices portion of the window, similar to the following figure.
Below are the storage devices you've selected to be a part of this installation. Please indicate using the arrows below which devices you'd like to use as data drives (these will not be formatted, only mounted) and which devices you'd like to use as system drives (these may be formatted). Please also indicate which system drive will have the bootloader installed.

Tip: Your filesystems on install target devices will not be wiped unless you choose to do so during customization.

Figure 21.

Click: **Next**

The system will respond with a disk partitioning window similar to the following figure:
Figure 22.

Click: **On the sda1 line to highlight this line**

Click: **Delete**

We are deleting the existing disk partition in order to allow the installer to create a fresh new partition. If you are installing on disks never before formatted, you will skip this step. The system will respond with a pop-up window confirming the partition delete similar to the following figure:
Figure 23.

Click: **Delete**

The system will respond by returning to the disk selection window as shown in the following figure:
Figure 24.

Click: **On the sda Free line to highlight this line**

Click: **Create**

The system will respond with a pop-up window requesting storage creation information, similar to the following figure:
Figure 25.

The Standard Partition radio button is already selected, which is the type of partition that we want to create. Leave this radio button selected and:

Click: Create

The system will respond with a pop-up window requesting partition creation information similar to the following figure:
Select: / (from the Mount Point: list box)

Click: to uncheck dasdb

Click: Fill to maximum allowable size

Click: OK

Leave the file system Type as ext4 (the default). The system will return you to the device selection window, similar to the following figure:
Click: **Next**

The system will respond with a pop-up window warning that a swap partition was not created, similar to the following figure:
Since we will define and use a swap device later, simply answer yes.

Click: Yes

The system will respond with another pop-up window confirming that the z/VM virtual disk will be formatted causing potential data loss, similar to the following figure.

Figure 28.
The system will respond with a panel, similar to the following figure, confirming that the previously selected operations will be written to the respective devices.

Figure 29.

Click: **Format**

The system will respond with a panel, similar to the following figure, confirming that the previously selected operations will be written to the respective devices.
Figure 30.

Click: Write changes to disk

The system will respond with a panel, similar to the following figure, prompting for packages to install.
In order to have the packages necessary for the optional workshop exercises we will need to select additional packages over the default installation.

Click: Customize now (radio button)
Click: Next

The system will respond with a package selection panel, similar to the following figure.
The package selection panel is divided into categories (such as Base System, Servers, Web Services), and then within each category package groups. It is possible to select or deselect a package group by clicking on the check box adjacent to the package group name. Package groups consist of base packages, and optional packages. Once a package group has been selected it is possible to add or subtract optional packages by clicking on the Optional Packages button when the package group is highlighted.

With Base highlighted:

Highlight: **Console internet tools**
Click: *Optional packages*

The system will respond with a pop-up window displaying the optional packages that can be selected in this group, similar to the figure below:

![Packages in Console internet tools](image)

*Figure 33.*

Click: `ftp-0.17-51.1.el6.s390x`

Click: *Close*

The system will respond by closing the pop-up window and returning you to the package selection window, similar to the following figure:
Figure 34.

With Base System still highlighted, scroll down on the right side until you see Legacy Unix Compatibility.

Click: **Legacy Unix Compatibility**

Continue scrolling until you see Storage Availability Tools

Click: **Storage Availability Tools**

Click: **Servers (on left side of window to highlight this group)**

With Servers highlighted:
Click: CIFS file server
Click: FTP server (then scroll down with scroll bar)
Click: Network Infrastructure Server
Click: Optional packages

The system will respond with a pop-up window displaying the optional packages that can be selected in this group, similar to the figure below:

![Packages in Network Infrastructure Server](image)

*Figure 35.*

Click: bind-9.7.3-2.el6.s390x
Click: bind-chroot-9.7.3.2.el6.s390x
Click: Close
The system will respond by closing the pop-up window and returning you to the package selection window, similar to the following figure:

![Optional packages selection window](image)

**Figure 36.**

Click: **System administration tools**
Click: **Optional packages**

The system will respond with a pop-up window displaying the optional packages that can be selected in this group, similar to the figure below:
The system will respond by closing the pop-up window and returning you to the package selection window, similar to the following figure:

Click: \texttt{lsscsi-0.23.2.el6.s390x}

Click: \texttt{Close}

The system will respond by closing the pop-up window and returning you to the package selection window, similar to the following figure:
Highlight Web Services:
    Click: **Web Server** *(to select)*

Highlight Desktops:
    Click: **Desktop** *(to select)*
    Click: **Optional Packages**
The system will respond with a pop-up window displaying the optional packages that can be selected in this group, similar to the figure below.

![Packages in Desktop](image)

*Figure 39.*

Deselect all checked optional packages, then:

- Click: `tigervnc-server-1.0.90-0... (to select)`
- Click: `Close`

The system will respond by closing the pop-up window and returning you to the package selection window, similar to the following figure.
Click: **Graphical Administration tools (to select)**
Click: **KDE Desktop (to select)**

Highlight Development:

Click: **Development tools (to select)**
Click: **Next**

The system will respond with a pop-up window indicating that package dependencies are being checked, similar to the following figure.
After the dependency checking is complete the system will begin the installation and respond with a package installation progress window similar to the following figure:
This panel will show which package is being installed, and overall progress completing the system install. This is a good point to take a break and stretch. When all of the packages have been installed, the system will respond with a panel similar to the following figure.

Figure 42.
Congratulations, your Red Hat Enterprise Linux installation is complete.

Please reboot to use the installed system. Note that updates may be available to ensure the proper functioning of your system and installation of these updates is recommended after the reboot.

Figure 43.

Click: Reboot

At this point, the system installation is complete. When you exit from this panel, Anaconda will end, and shutdown the ramdisk based system currently running in your virtual machine. Since the Linux operating system (and it’s TCP/IP stack) will end, your PuTTY Telnet session will also terminate. At this point, you should return to your 3270 emulator session (which represents
your Linux virtual machine console). You will see that your system is being rebooted as shown in the following figure.

Figure 44.

Continue clearing the screen (Pause/Break key) until you see a login prompt, similar to the following figure.
Enter: root (for login userid)

Enter: linlabnn (where nn is your team number, for password)

At this point, you are logged in as root, and your installation of RedHat Enterprise Linux Server Release 6.1 is complete. Congratulations, you have reached a milestone in the lab!
2.4 Checkpoint 1

You have now completed your installation of RedHat Enterprise Linux Server Release 6.1 for System z. You are now ready to proceed to some post install tasks.

It will be easiest to perform these tasks using a PuTTY SSH session.
Chapter 3. Post Install Tasks

Unlike RedHat Enterprise Linux 4, it is no longer possible to specify that you do not want the firewall activated, or SELinux enabled from the Installation GUI. Since we will work with IPTABLES (the Linux Firewall interface) in a later chapter to setup a very simple firewall rather than make use of the prebuilt RedHat firewall, we need to disable it now. In addition, to avoid additional complication in the workshop environment, we will also disable SELinux.

3.1 SSH To Your Guest

Return to your workstation desktop, and double click on the PuTTY icon to begin a session. The system will respond with a window similar to the following.
Figure 46.

Click to highlight: **LINLAB**

Click: **Load**

The system will respond with a window similar to the following:
Figure 47.

Click your mouse pointer in the text box named “Host Name (or IP address)”.  

Type: 9.82.56.(91-120)  

Click: Open  

When prompted to enter a userid:  

Enter: root  

When prompted to enter a password:  

Enter: linlabnn (where nn is your team number)
The system will respond by displaying a post install setup tool window similar to the figure below.

### 3.2 Disable Firewall and SELinux

![Choose a Tool](image)

*Figure 48.*

Select **Firewall configuration** (using the keyboard arrow keys)

Enter: Run Tool (by using the tab key to highlight that button and pressing Enter)

The system will respond with a screen similar to the following figure:
Chapter 3. Post Install Tasks

Figure 49.

Tab to: **Enabled** *(and press the space bar to deselect)*
Tab to: **OK** *(and press Enter key)*

The system will respond with another window confirming your action, similar to the following figure:
Figure 50.

Tab to: **Yes** *(and press Enter key)*

The system will return to the Choose a Tool menu.

Tab to: **Quit** *(and press Enter key)*

The system will return to the shell prompt in your PuTTY session. RedHat Enterprise Linux 5.5 allowed you to disable SELinux from the setup tool we just used. This is no longer possible with RedHat Enterprise Linux 6.1. In order to disable SELinux you will need to update a configuration file, and reboot the system.

Enter: `cd /etc/sysconfig`
Enter: `cp selinux selinux.original`
Enter: `ed selinux`
Enter: `7`Enter: `.c`
Enter: `SELINUX=disabled`
Enter: `.`Enter: `1,$p`
Enter: `1,$w`
Enter: `q`

To reboot your Linux virtual machine it is best to exit from your PuTTY session, and return to your 3270 console session. From that session:
Enter: `shutdown -r now`

Continue clearing the screen as you receive the more... status until you see the login prompt.

Enter: `root (as login id)`
Enter: `linlabnn (for password)`

Before proceeding to the next section, you will want to establish a PuTTY session again.

Return to your workstation desktop, and double click on the PuTTY icon to begin a session. The system will respond with a window similar to the following.
Figure 51.

Click to highlight: **LINLAB**

Click: **Load**

The system will respond with a window similar to the following:
Click your mouse pointer in the text box named “Host Name (or IP address)

Type: 9.82.56.(91-120)

Click: Open

When prompted to enter a userid:

Enter: root

When prompted to enter a password:

Enter: linlabnn (where nn is your team number)
The system will respond by establishing a new SSH session for you as user root, and placing you in root's home directory. You should receive a screen similar to the following figure.

![Screenshot of SSH session](image)

### 3.3 Build Package

After we add some additional LUNs and configure a multipath logical volume, it will be good to run a disk exerciser to make sure that we are using all of the paths we define. The RedHat installation does not include the Bonnie disk exerciser, so we will need to build it from a source tar file.

Enter: `cd /usr/src`
Enter: `ftp linftp2` (when prompted for a userid enter anonymous)
Enter: `cd /pub`
Enter: `binary`
Enter: `get bonnie.tar.gz`
Enter: `quit`

We are now ready to begin building the bonnie binary.
Enter: `mkdir bonnie`
Enter: `cd bonnie`
Enter: `tar -xzvf ..;/bonnie.tar.gz`
Enter: `make` (this will compile the binary, you will see some warning messages which you may ignore)
Enter: `cp Bonnie /sbin`

This completes the post installation tasks. We are now ready to proceed with adding some additional SCSI LUNs to our system.
Chapter 4. Add Additional FCP Disks for Multipath LVM

Now that Linux is installed on a single FCP attached LUN, we will turn our attention to preparing additional FCP attached LUNs for use with applications and application data. To avoid complication in the lab environment we will leave the root device used during installation as it is. In the following chapter we will add two additional devices to the current configuration, create a multipath.conf file, and activate the drivers and daemons needed to support a multipath device.

4.1 Add Additional FCP attached SCSI LUNs

RedHat does not provide a GUI tool for defining zFCP devices. While that might be viewed as a negative, it does provide the opportunity to see the low level process that is typically hidden by GUI interfaces.

In the System z environment, the FCP subchannel dedicated to a guest is equivalent to a Host Bus Adapter (HBA). For this workshop, your guest has two FCP subchannels dedicated, one on each of two physical FCP chpids managed by the z/VM system. You have already used the first HBA at address 0A00. You also have an HBA at address 0B00.

The first thing that needs to be done then is to bring that adapter online so that paths to the new LUNs can be defined using it.

The following commands should be entered from the PuTTY session you were using in the previous chapter

Enter: `chccwdev -e 0.0.0b00`  

Next we need to define the target Logical Unit Numbers (LUNs) to the two HBAs. In the instructions below you should replace `nn` with the last two digits of the `Addtl LUN1` entry that corresponds to your userid in Appendix C. Also, replace `yy` with the last two digits of the `Addtl LUN2` entry that corresponds to your userid in Appendix C.

Enter: `cd /sys/bus/ccw/devices/0.0.0a00`  
Enter: `cd 0x500507630b08406e`  
Enter: `echo 0x00nn000000000000 > unit_add`  
Enter: `echo 0x00yy000000000000 > unit_add`  
Enter: `cd ..`  
Enter: `cd 0x500507630b03406e`  
Enter: `echo 0x00nn000000000000 > unit_add`  
Enter: `echo 0x00yy000000000000 > unit_add`
Enter: `cd ../../0.0.0b00`
Enter: `cd 0x500507630b08406e`
Enter: `echo 0x00nn000000000000 > unit_add`
Enter: `echo 0x00yy000000000000 > unit_add`
Enter: `cd ../0x500507630b03406e`
Enter: `echo 0x00nn000000000000 > unit_add`
Enter: `echo 0x00yy000000000000 > unit_add`

This completes the process of dynamically entering 4 paths for the two new LUNs being used by your guest. In order to see the block special device names associated with the new paths use the `lsscsi` command.

Enter: `lsscsi`

The system will respond with a display similar to the following figure:

![Display of lsscsi output]

Figure 53.

This display shows the path for a SCSI device, information about the type of device (i.e. disk, and IBM 2107), and the Linux block special name currently assigned to the device.
The path column looks a bit strange [0:0:0:0]

The information within the brackets consists of 4 fields with the following meaning:

a. SCSI_host
b. Channel
c. Target_id
d. LUN

In the Linux for System z context, the Channel is always 0, and taken together with the SCSI_host field represents the FCP subchannel that the device is associated with. In the context of this lab, seeing the values 0:0 for fields a and b indicate a LUN on subchannel 0.0.0a00. Values of 1:0 indicate a LUN on subchannel 0.0.0b00.

Field c, the Target_id represents a particular target WWPN used to access the LUN. Thus a value of zero will represent the first target WWPN defined (in our case that is 0x500507630b08406e), and a value of 1 represents the second target WWPN defined (again in our case that would be 0x500507630b03406e).

Field d, the LUN number is the LUN number converted from hex to decimal. So [0:0:0:18] represents LUN 12 defined on FCP subchannel 0.0.0a00 with target WWPN 0x500507630b08406e. Similarly [0:0:0:19] represents the LUN 13 defined on FCP subchannel 0.0.0a00 with target WWPN 0x500507630b08406e, and [0:0:1:18] represents LUN 12 defined on FCP subchannel 0.0.0a00 with target WWPN of 0x500507630b03406e.

You can use the output of this display to see at any given time which devices actually represent different paths to the same target LUN. In the context of this workshop, the second FCP subchannel only contains definitions for the additional LUNs across both target WWPNs. Thus we can see that /dev/sdb, /dev/sdd, /dev/sdf, and /dev/sdh all represent the same LUN accessed via different paths (i.e. FCP subchannel and target WWPN). Similarly, /dev/sdc, /dev/sde, /dev/sdg, and /dev/sdi all represent the second additional LUN available to our guest.

It is possible to verify which target WWPN, and device LUN number are associated with any given row in the lsccsi output from the /sys file system. In the case of the third row [0:0:0:73] you would do the following:

Enter: cd /sys/class/scsi_device/0:0:0:73/device
Enter: `cat fcp_lun` (to see the lun id, e.g. 0x0049000000000000)
Enter: `cat wwpn` (to see the target WWPN, e.g. 0x500507630b08406e)
Enter: `cat hba_id` (to see the FCP subchannel e.g. 0.0.0a00)

### 4.2 Harden Dynamic Changes

All of the changes made in the last section were dynamic. If we reboot the Linux system at this point, all of the changes will be lost. We can harden these changes by updating the `/etc/zfcp.conf` file. If the new FCP disks were to be used with the root file system, then we would update zipl.conf instead since the devices would need to be available very early in the boot process. In the context of this workshop, these new disks are considered data disks, so updating zfcp.conf is not necessary. In the instructions below you should replace `nn` with the last two digits of the `Addtl LUN1` entry that corresponds to your userid in Appendix C. Also, replace `yy` with the last two digits of the `Addtl LUN2` entry that corresponds to your userid in Appendix C.

Enter: `cd /etc`
Enter: `cp zfcp.conf zfcp.conf.orig`
Enter: `ed zfcp.conf`
Enter: `$`
Enter: `:a`
Enter: `0.0.0a00 0x500507630b08406e 0x00xx000000000000`
Enter: `0.0.0a00 0x500507630b08406e 0x00yy000000000000`
Enter: `0.0.0a00 0x500507630b03406e 0x00nn000000000000`
Enter: `0.0.0a00 0x500507630b03406e 0x00yy000000000000`
Enter: `0.0.0b00 0x500507630b08406e 0x00nn000000000000`
Enter: `0.0.0b00 0x500507630b08406e 0x00yy000000000000`
Enter: `0.0.0b00 0x500507630b03406e 0x00nn000000000000`
Enter: `0.0.0b00 0x500507630b03406e 0x00yy000000000000`
Enter: `.
Enter: `1,$w`
Enter: q

At this point the devices have been added to the appropriate configuration file and made available to your guest.

At this point it is a good idea to reboot your Linux guest and make sure that the changes just made were properly hardened.

Exit from your PuTTY session and return to your 3270 console session.

Enter: `shutdown -r now`

Keep clearing the messages through the Linux shutdown and subsequent boot, until you see the login prompt.

Enter: `root (as the login userid)`

Enter: `linlabnn (where nn is your team number) for the password prompt)`

From the shell command line:

Enter: `lsscsi`

The system will respond with a display similar to the following figure:
4.3 Obtain UUIDs of New FCP LUNs

When the device mapper portion of the Linux kernel works with an FCP attached SCSI LUN, it assigns the new logical device created a name consisting of the Universal Unique Identifier (UUID) returned from the device. These names are very difficult to work with since they consist of a large string of numbers. Within the multipath.conf file, it is possible to assign an alias to a particular UUID when naming the device.
Before updating the multipath.conf file, we need to find the UUIDs of the two new LUNs added to the guest.

Enter:
```
scsi_id --page=0x83 --whitelisted --device=/dev/sdb
```

Enter:
```
scsi_id --page=0x83 --whitelisted --device=/dev/sdc
```

The system will respond to each of these commands with a string of numbers similar to the figure below. Jot down the strings for use in creating the multipath.conf file.

![Screenshot of command output]

Figure 55.
It you would rather use vi (vim) to create the multipath.conf file, then you will need to establish a PuTTY SSH session, and create the file from that session. In fact, it is recommended that you establish a PuTTY SSH session even if you are going to use ed as described below since entering brackets and curly braces many times does not work well from a 3270 emulator. The instructions below use ed to create the new file.

    Enter: cd /etc
    Enter: ed multipath.conf
    Enter: .a
    Enter: multipaths {
    Enter: multipath {
    Enter: wwid <the first UUID from sdb>
    Enter: alias mymp1
    Enter: }
    Enter: multipath {
    Enter: wwid <the second UUID from sdc>
    Enter: alias mymp2
    Enter: }
    Enter: blacklist {
    Enter: devnode "^(dasd) [0-9]*"
    Enter: }
    Enter: .
    Enter: 1, $p
    Enter: 1, $w
    Enter: q

Now that the multipath.conf file has been created, we are ready to run the multipath binary to create the multipath devices. This command will cause the device mapper tables to be built according to the criteria specified in the /etc/multipath.conf file. Before running the command though we will need to load the device mapper driver, and the device mapper multipath target.

    Enter: modprobe dm_mod
    Enter: modprobe dm_multipath
    Enter: /sbin/multipath

The system will respond with a display showing the new multipath devices and the lower level scsi devices that have been grouped to create the multipath devices.
The figure below is a logical diagram of what you have just defined in terms of physical devices and multipath devices.
4.4 Modify /etc/lvm.conf

We are almost ready to begin using the new multipath devices in a logical volume configuration. However, by default, the lvm.conf file allows lvm to recognize too many devices. To avoid conflicts that might arise when metadata is discovered on physical devices as well as logical multipath devices we will modify the lvm.conf file to limit lvm to just device-mapper devices.

Enter: cd /etc/lvm
Enter: cp lvm.conf lvm.conf.original
Enter: ed lvm.conf
Enter: 20
Enter: .c
Enter: scan = [ "/dev", "/dev/mapper" ]
Chapter 4. Add Additional FCP Disks for Multipath LVM

Enter: .
Enter: 53
Enter: .c
Enter: filter = [ "a|/dev/mapper/.*/", "r|.*" ]
Enter: .
Enter: 84
Enter: .c
Enter: types = [ "device-mapper", 16 ]
Enter: .
Enter: 1, $w
Enter: q

At this point we should be ready to create a logical volume from the two multipath devices created by device mapper.

4.5 Create Physical Volumes

First designate the two multipath devices as lvm physical volumes

Enter: pvcreate /dev/mapper/mymp1p1
Enter: pvcreate /dev/mapper/mymp2p1

Note: in the above commands we are referring to the devices as /dev/mapper/mymp1p1 instead of just /dev/mapper/mymp1. This is because the underlying LUNs have been previously partitioned using fdisk to have 1 primary partition. When the device mapper devices are created the partitions on each device are detected, and code known as kpartx is invoked to create device-mapper logical devices for both the base, and partition. When we create a physical volume, we want to reference the partition and not the base device so we need to use the name that kpartx assigned which is our alias plus p1.

4.6 Create lvm Volume Group

Now that we have physical volumes we can create an lvm volume group consisting of these two physical devices.

Enter: vgcreate myvg /dev/mapper/mymp[1-2]p1

The command above will create a volume group named myvg consisting of two physical volumes named /dev/mapper/mymp1p1 and /dev/mapper/mymp2p1. Use the vgdisplay command to see the characteristics of the volume group.

Enter: vgdisplay myvg
4.7 Create lvm Logical Volume

Now that we have a volume group, we can create one or more logical volumes in that volume group. Since the size of our group is small, we will only create one logical volume that consumes all of the space in the volume group.

Enter: `lvcreate --name mylv --size 1.9G --stripes 2 --stripesize 16 myvg`

Note: in the command above, we are creating two stripes to distribute the I/O load across the two multipath devices. The device mapper (using the round-robin scheduler) will make sure that I/O is distributed evenly across each of the 4 paths for an underlying FCP LUN.

4.8 Create File System on Logical Volume

After creating the logical volume, we have a device on which we can place a file system. We will create a journaling file system (EXT4) on the logical device for fast recovery if the system crashes.

Enter: `mkfs.ext4 /dev/myvg/mylv`

After the file system is created, mount the new file system over /mnt

Enter: `mount /dev/myvg/mylv /mnt`

We can also display the file system using df.

Enter: `df`

The system will respond with messages similar to the figure below.
Chapter 4. Add Additional FCP Disks for Multipath LVM

4.9 Configure Necessary Boot Scripts

At this point we have completed the steps for creating the multipath devices and the logical volume. Before rebooting the Linux system to make sure everything was created properly we need to activate some boot scripts so that the device mapper is loaded and the multipath module is executed early in the boot process before the lvm code scans to find available volumes.

Enter: *chkconfig multipathd on*

With the command above we are ensuring that the multipathd daemon is started to monitor path status and dynamically make changes to the device mapper tables in response to events such as loss of a path etc.

One thing we have not done is update *fstab* to automatically mount our new file system at boot time. It is best at this point to test and make sure that all of the pieces are setup properly at boot time before updating *fstab*. That way if an error occurs, our guest will not abort the boot process because a file system cannot be automatically mounted.

At this time exit your ssh session (PuTTY) and switch to your 3270 session.

Enter: *shutdown -r now*

Keep clearing the messages on your 3270 session as the guest shuts down, and boots again. When you receive the login prompt:

Enter: *root*

Enter: *linlabnn (where nn is your team number)*
The system will respond by starting a shell session for root on the console device and displaying the shell prompt.

Make sure that your multipath devices were created at boot:

Enter: `multipath -l`

You should see a response listing mymp1 and mymp2 along with all of the individual /dev/sdx kernel devices, similar to the figure below.

```
[root@linlab02 ~]# multipath -l
multipath -l

[mymp2 (36005076303ffec0b9d000000000001056) dm-1 IBM,2107900 [On
size=1.0G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=-1 status=active
| 0:0:1:3 sdc 8:32 active undef running
| 0:0:0:3 sde 8:64 active undef running
| 1:0:0:3 sgg 8:96 active undef running
`- 1:0:1:3 sdi 8:128 active undef running

[mymp1 (36005076303ffec0b9d000000000001057) dm-0 IBM,2107900 [On
size=1.0G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=-1 status=active
| 0:0:1:2 sdb 8:16 active undef running
| 0:0:0:2 sde 8:48 active undef running
| 1:0:0:2 sdf 8:80 active undef running
`- 1:0:1:2 sdh 8:112 active undef running
```

Try to mount your new logical volume to make sure that the lvm scan worked properly.

Enter: `mount /dev/myvg/mylv /mnt`

The system should respond with messages indicating the EXT4 journal is starting, and then a shell prompt. To make sure the file system was indeed mounted, display the mounted file systems:

Enter: `df`

If no errors were encountered you are now ready to add the new file system to fstab and proceed with other chapters in the workshop.

First create a new mount point for the file system so /mnt is not permanently used.
Enter: `mkdir /appdata`

Next backup the fstab file and add the file system to fstab.

Enter: `cd /etc`
Enter: `cp fstab fstab.original`
Enter: `ed fstab`Enter: `9`Enter: `.a`Enter: `/dev/myvg/mylv /appdata ext4 defaults 1 2`Enter: `.`Enter: `1,$p`Enter: `1,$w`Enter: `q`

If you are currently logged in using ssh (PuTTY) exit from that session and switch to your 3270 session. At this point we will reboot once more to make sure that the new file system automatically mounts without any problems.

Enter: `shutdown -r now`

Continue clearing the messages on your screen while shutdown progresses and the reboot occurs. When prompted to login:

Enter: `root`Enter: `linlabnn (where nn is your team number)`

The system will start a shell session for root on your console device and return a shell prompt. Make sure that your file system is mounted.

Enter: `df`

In the response from df you should see that `/dev/mapper/myvg-mylv` is mounted at `/appdata`.

### 4.10 Optional - Test Multipath and LVM I/O Distribution

One last thing you might want to do is test that your multipath device is in fact using all of the defined paths, and that LVM is spreading the load across the two multipath devices.

To conduct this test we will use the open source file system benchmark bonnie along with the iostat command to see where the I/Os are going. You will need both your 3270 session and an ssh session (PuTTY).
Once again please double click on the PuTTY desktop icon. You will receive a window similar to the following figure:

![PuTTY Configuration Window](image)

Figure 57.

Click to highlight: **LINLAB**

Click: **Load**

The system will respond with a window similar to the following:
Click your mouse pointer in the text box named "**Host Name (or IP address)**".

Type: **9.82.56.(91-120)**

Click: **Open**

When prompted to enter a userid:

Enter: **root**

When prompted to enter a password:

Enter: **linlabnn (where nn is your team number)**
The system will respond by establishing a new SSH session for you as user root, and placing you in root's home directory. You should receive a screen similar to the following figure.

Enter: `iostat 5`

The system will respond by executing the `iostat` command and refreshing the display every 5 seconds. In the output from `iostat` (shown below) you should see all of the kernel devices that represent the paths to the alternate LUNs (i.e. `/dev/sdx`). You should also see devices named `dm-x` which represent the logical device mapper devices that were created for the multipath base.
devices, multipath partition devices, and logical volume. You will want to watch the BLK I/O counts as you invoke bonnie.

```
Figure 59.
```

Switch to your 3270 session in order to start bonnie.

```
Enter: cd /appdata
Enter: Bonnie -s 1024
```

In the commands above we switch to the mount point for our new file system (that is where bonnie will read from and write to by default), and then invoke bonnie specifying a file size of 1024M. We need to specify a file system greater than 768M so that operations are not just done in cache. After invoking bonnie switch back to your ssh session and watch the iostat output.
You should see the io counts increase evenly across all of the /dev/sdx devices (excluding /dev/sda our root device), as well as the dm-x devices, as shown in the figure below.

![Figure 60.](image)

When bonnie completes on the 3270 session you can stop iostat by entering CTRL-C on the ssh session.
4.11 Checkpoint

Congratulations! You have just added two new FCP LUNs to your Linux system, along with the additional devices that represent alternate paths to those devices. You have configured and activated the Linux multipath support so that the alternate paths to those devices can be used, and configured and created a logical volume that spans the two multipath devices. In addition, you have optionally verified that all of the paths and striping is working properly by using the open source bonnie benchmark along with iostat. You are now ready to continue with other Linux system administration and configuration tasks.
Chapter 5. Basic Linux For System z Administration

5.1 Setup z/VM Virtual Disk Swap Device

Please enter the following commands from your PuTTY session.

Enter: `mkswap /dev/dasda`
Enter: `swapon /dev/dasda`
Enter: `swapon -s`

You should see that your swap device is being used. In the response to `swapon -s` you should see /dev/dasda under the heading Filename, partition under the heading Type, 9992 under the heading Size, 0 under the heading Used, and -1 under the heading Priority.

So that the swap device will be used automatically when the guest is rebooted, we need to update the rc.local startup script. The rc.local script will automatically run during the Linux boot process, following all of the other startup scripts. As an alternative, we could have placed a swap entry in the Linux file system table (/etc/fstab). However, this would assume that the swap device was already formatted and available for use. Since we are using a z/VM virtual disk it might not be formatted prior to a particular system boot. Updating rc.local allows us to perform a mkswap prior to activating the device as swap space.

Enter: `cd /etc/rc.d`
Enter: `cp rc.local rc.local.original`
Enter: `ed rc.local`
Enter: `$`
Enter: `.a`
Enter: `mkswap /dev/dasda`
Enter: `swapon /dev/dasda`
Enter: `.`
Enter: `1,$p`
Enter: `1,$w`
Enter: `q`

5.2 Networking FYI

We are using a z/VM Guest LAN virtual QDIO interface for network configuration in this lab. You have seen what is needed from the Linux for System z side, but how about the VM TCP/IP side? So that you will be able to implement a similar setup in your own shop, excerpts have been included
from the PROFILE TCPIP file showing the clauses added to support connection to a qdio Guest Lan.
The following two statements define a QDIO device and link for connection to the system owned guest lan LINLAB:

```bash
DEVICE QDI3 OSD 9D0 PORTNAME LINLAB PRIR
LINK ETH3 QDIOETHERNET QDI3
```

The following statement was added to the HOME section of PROFILE TCPIP, after the address records for the main hardware supplied interfaces. It provides an IP address for the virtual network interface associated with link ETH3:

```bash
9.82.56.1 ETH3
```

The following statement was added to the GATEWAY section of PROFILE TCPIP. It instructs VM TCP/IP to route any packets addressed to the 9.82.56.1 network over the ETH3 link.

Notice, VM's presence on this network is the IP address defined under the HOME statement for ETH3.

```bash
9 = ETH3 1500 0.255.255.0 0.82.56.1
```

The following statement automatically starts the QDIO connection to the system owned guest lan LINLAB.

```bash
START QDI3
```

Figure 61. PROFILE TCPIP Excerpt

### 5.3 Add A New User

One of the first things you will need to do when creating a new Linux For System z system is add a new userid. This will allow you to login as someone other than root, and will allow for testing of Samba later on in this lab.

RedHat Enterprise Linux 6.1 provides a GUI application for user and group maintenance, or if you prefer, you can add users and groups using the command line. We will cover both options.
5.3.1 Add a New User Using GUI Interface

Since we will be running an X Windows application, we will need to start the vncserver on our Linux virtual machine.

Enter: \texttt{cd /root}
Enter: \texttt{vncserver}

The system will respond by asking you to enter a password to protect the vnc desktop.

Enter: \texttt{linlabnn (where nn is your team number)}
Enter: \texttt{linlabnn (when prompted to reenter your password)}

Now that the vncserver is running, you will need to switch back to your workstation desktop and double click the vncviewer icon. In response, your workstation will present a small window similar to the following figure.

![Figure 62. In the VNC server box enter the IP address of your Linux virtual machine: Enter: 9.82.56.(91-120):1 (note do not enter the parentheses) Click: Connect The workstation will respond with a password prompt window similar to the following:](image)

In the VNC server box enter the IP address of your Linux virtual machine:

Enter: \texttt{9.82.56.(91-120):1 (note do not enter the parentheses)}

Click: \texttt{Connect}

The workstation will respond with a password prompt window similar to the following:
Enter: linlabnn

The system will respond with a pop-up window warning against logging in as root similar to the following figure.

Click: Do not show me this again
Click: Close

The system will respond by displaying a window containing a desktop, similar to the following figure.

Figure 65.

Move your mouse pointer inside the vnc window, and:

- Click: Applications (Upper left corner)
- Select: System tools --> Terminal

In the new xterm window that opened:

- Type: system-config-users (at the shell prompt)
- Press: ENTER

The system will respond with a screen similar to the following figure.
Figure 66.

**Click: Add User**

The system will respond with a panel similar to the following figure.
Move your mouse pointer inside the text box associated with "User Name:" and click the left mouse button.

- Type: administrator
- Tab to: Full Name:
- Type: Samba User
- Tab to: Password:
- Type: linlabnn (where nn is your team number)
- Tab to: Confirm Password:
- Type: linlabnn (where nn is your team number)

At this point, your screen should look similar to the following figure.
Figure 68.

Click: Ok

The system will respond by displaying a pop-up window indicating the chosen password is weak, similar to the following figure.
Click: Yes

The system will respond by adding the new user, creating a home directory for the new user, and re-displaying the add user panel, similar to the following figure.
Figure 70.

To end the config users application click on the File menu, and select Quit.

To end the vncviewer session, click the X in the upper right corner of the window containing the desktop.

5.3.2 Add a New User Using Command Line

Now that you have experienced adding a user with the GUI, we will show you how to add a new user using the command line. Switch back to your PuTTY ssh session to enter the following commands.
Enter: `adduser -m llnn (where nn is your team number)`

Note: this will create an entry in the passwd file for llnn, and will create a directory for this user under /home (because the -m flag was specified). The entry created for llnn in the passwd file will not have a password assigned at this time. You will need to create a password for later use.

Enter: `passwd llnn (where nn is your team number)`

When you are prompted to enter the password

   Enter: `linlabnn (where nn is your team number)`

When you are prompted to retype the new password

   Enter: `linlabnn (where nn is your team number)`

5.4 Update Message Of The Day File

Now that you will have additional users logging in to your system, you might want to create a message-of-the-day so that users logging in are properly warned about the current state of this system :-) If you are working with a disconnected server, you might want to use this file as a running commentary on changes made to the system. That way, when you login to do maintenance, you will be able to see immediately what the last change was.

   Enter: `cd /etc`
   Enter: `ed motd`
   Enter: `.a`
   Enter: `Warning: this is a lab system, use at your own risk!`

Enter: `.`
Enter: `1,$w`
Enter: `q`

You will see this message next time you login

5.5 Monitoring Disk Usage

Since your system is limited in disk space, it would be helpful to know how to monitor disk usage

   Enter: `df` - to see the amount of disk space used and available on mounted file systems
Enter: `du` - to see the amount of disk space used by files in the current directory, or a specified directory tree. By default space used is reported in units of 1024 bytes.

Use the `find` command to locate files of a particular size, or files that have not been accessed in a reasonable amount of time. E.g.

Enter: `find / -name core -atime +3 -exec rm {} \`; (this command will remove any files named core (core dumps) that have not been accessed for several days)

Enter: `find /var -size +100 -atime -1 -ok rm {} \`; (this command will search through /var, or whatever other directory you specify, looking for files larger than 100 blocks that have been accessed within the past 24 hours. When a file is found matching this criteria, you are asked whether you want to delete the file) Note: you will receive some hits to this command, so be sure to enter `n` when you are prompted to remove the file!

Enter: `find / -type f -mtime -1 -print` (this command lists all of the regular files that have been modified within the past day. This can be useful to note what you have changed on the system, or what file has been dynamically updated by the system)

Use `find` with `grep` to locate files that contain a particular string in the file content

Enter: `set network` (Causes the string “network” to be the argument passed to the command below)

Enter: `find . -type f -exec grep -l "$1" {} \`; (Run this command from the command line, or better yet, follow the steps below and place this command in a shell script. Note: the flag to `grep` is a lowercase L (`-l`))

Enter: `cd /root`
Enter: `ed finder` Note: this is a new file
Enter: `.a`
Enter: `#!/bin/sh`
Enter: `find . -type f -exec grep -l "$1" {} \`; (Note: the flag to `grep` is a lowercase L (`-l`))

Enter: `.`
Enter: `1,$w`
Enter: `q`
Enter: `chmod +x finder`

You can then test it by entering `./finder find`
You should see a response that indicates the file finder contains the string 
find. If you want to search other directories, move the script to the /usr/sbin/
directory (which is in your execution PATH when logged in as root).

Enter: `cp finder /usr/sbin`

You can now change directory (cd) to any other location (for example, /etc) and enter `finder some_string`.

### 5.6 Additional Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>w</code></td>
<td>show who is logged on and resource usage</td>
</tr>
<tr>
<td><code>who</code></td>
<td>show who is logged on</td>
</tr>
<tr>
<td><code>last</code></td>
<td>show listing of last logged in users</td>
</tr>
<tr>
<td><code>lastlog</code></td>
<td>examine lastlog file, formats and prints contents of last login file</td>
</tr>
<tr>
<td><code>whoami</code></td>
<td>print effective userid</td>
</tr>
<tr>
<td><code>top</code></td>
<td>display top CPU processes (don't run on 3270 session unless you have the cancel key properly setup ^c)</td>
</tr>
<tr>
<td><code>uptime</code></td>
<td>tell how long system has been running</td>
</tr>
<tr>
<td><code>free</code></td>
<td>display amount of free and used memory</td>
</tr>
</tbody>
</table>

### 5.7 If You Have Extra Time

Since the shell is your interactive interface to the Linux system, if you finish
the tasks in this chapter early, it would be good for you to spend some time
reading about it. The most commonly used shell in a Linux environment is the
Bourne Again Shell (BASH). The manual page for BASH provides lots of
good information about using and customizing the shell environment. To view
the manual page you must be logged in to your Linux for System z guest
using an ssh session. Then:

Enter: `man bash`

To provide a purpose for your investigation of BASH, try to accomplish the
following tasks:

- Create a BASH profile in the home directory for user root (/root)
- Read about the PS1 variable and the section on PROMPTING
- In the new profile export the PS1 variable to change your prompt to
  include a full directory path instead of the current directory
- Don’t forget to make your shell script executable using the chmod
  command
Chapter 6. Configuring IPTABLES

6.1 Introduction

The Linux kernel includes code that allows you to use your Linux for System z instance as a packet filtering firewall. This means that you will be able to use your Linux for System z instance to prevent packets from leaving your system and going to certain destinations, as well as prevent packets from entering your system from sites on an external network.

A packet-filtering firewall typically operates at the IP network and transport protocol layers. In addition it is usually implemented within an operating system, as in the case of Linux. The basic way in which this type of firewall protects a system is by making routing decisions after filtering packets using information in the IP packet header. The firewall allows you to either discard a packet and notify the sender of this action, or simply discard a packet without notification. The packet-filtering firewall makes decisions based upon the network interface and host IP address over which a packet was received, the source and destination IP addresses, the TCP or UDP ports, the TCP connection flags, the ICMP message types and whether the packet is incoming or outgoing.

The packet-filtering firewall on your Linux for System z instance is known as netfilter. This name relates well to how the firewall operates. The firewall performs input and output filtering by applying lists of rules to individual packets. The lists of rules are defined as chains because a packet is matched against each rule in the list until a match is found or the list ends. For a particular network interface both an input chain and an output chain are maintained.

The kernel loaded during the installation process already has the appropriate configuration options selected to activate use of the netfilter functions. Therefore nothing further need be done at this point from a kernel perspective. In this section, we will use the administration tool (iptables) and experiment with the firewall.

6.2 Verify IPTABLES Version

To verify that the iptables utility is installed, and determine the version, enter the following command from your PuTTY ssh session:

Enter: `iptables --version`
You should receive a response indicating the version of iptables. If you get something else, it would mean that your kernel is not configured to include the firewall support.

From your PuTTY SSH session, find the ip address of your Windows (PC) workstation:

   Enter: w

The system will respond with a list of users logged into the system and the source IP address associated with a particular pseudo terminal or TTY device, similar to the following figure:

![Figure 71.

Note the IP address in the FROM column. This is the IP address of your Windows Workstation.

The `iptables` command can now be used to implement a packet filtering security policy. In this lab, we will simply use the `iptables` command to restrict your windows workstation from sending ping commands.
Enter: `ping 9.82.56.(91-120)` from an MS-DOS prompt on your windows workstation to ensure that you can ping your Linux for System z guest

Return to your PuTTY ssh session:

Enter: `iptables -A INPUT -s xxx.xxx.xxx.xxx -p icmp -j DROP` from a Linux prompt

Note: replace xxx in the IP address above with your workstation IP address obtained in the earlier step

On your windows workstation:

Enter: `ping 9.82.56.(91-120)` from an MS-DOS prompt

At this point you should be unsuccessful in your ping attempt. Return back to your PuTTY ssh session:

Enter: `iptables -D INPUT 1` from a Linux prompt

This will remove the rule defined earlier. Try the `ping` command from your workstation once again to see that you are now able to pass icmp packets to your linux guest.

### 6.3 Configuring iptables

Since the filtering chains are dynamically defined to the kernel, you will most likely want to save your filtering rules in a script that can be executed during the startup process. RedHat makes this very easy by providing a predefined script, and operand to the service command. Enter the following commands from your PuTTY ssh session.

Enter: `iptables -A INPUT -s xxx.xxx.xxx.xxx -p icmp -j DROP`

Enter: `service iptables save`

This command will use the iptables-save utility to store the current rules in a file named iptables, located in the `/etc/sysconfig` directory. When the system is restarted, one of the init scripts will use the contents of this file, and the iptables-restore utility to establish the rule set you had in place when the save command was executed.

Test your change. Exit from your PuTTY ssh session and return to your 3270 console session.

Enter: `shutdown -h now`
When you receive message HCPGSP2630I
Enter: IPL A00

When prompted for a userid and password
Enter: root
Enter: linlabnn

6.3.1 Check Netfilter Operation
From your workstation MS-DOS prompt try pinging your Linux for System z guest again to make sure you are blocked by the filtering rules. You can also list the rules in the INPUT chain of the Filter table by entering the following command from your 3270 console session:

Enter: iptables -L INPUT

The system response should list the rule blocking icmp packets from your workstation IP address.

6.4 Checkpoint 2
At this point, you now have a running Linux for System z server. You have customized your server to:

- Support new users, one of which is named administrator

You have also seen some Linux commands that enable you to monitor disk usage, and search for files with particular attributes in the file system.

In addition, you used the iptables command to define a firewall rule, and then saved that rule so that it would be activated whenever the system was started.

You are now ready to continue and customize another application that will allow your Linux for System z to serve as a local caching only name server.

Congratulations!

6.5 If You Have Extra Time
If you have finished the activities of this chapter early, you might want to continue exploring BASH. One nice feature of UNIX, and Linux shells is that
they maintain a history file. The history file records some number of commands that you have entered at the shell prompt (the number maintained is set when the shell environment is customized). One very useful way to exploit this file is to quickly recall a command that was entered and reexecute it. It is also possible to recall a command entered, and modify it slightly before executing it.

To learn more about this capability, read the BASH man pages, and look for the paragraph under the heading HISTORY, and HISTORY EXPANSION. You can also read about the fc and history shell commands under the SHELL BUILTIN COMMANDS heading. After reading about this capability, experiment with reexecuting commands from your history file, and editing commands before reexecuting them. Note: since the editing function uses VI by default, you should perform these experiments from an ssh session into you Linux for System z guest.
7.1 Introduction

BIND stands for Berkeley Internet Name Domain. BIND is an implementation of the Domain Name System (DNS). DNS servers provide translation services between fully qualified host names (endvmst.endicott.ibm.com) and IP addresses (9.82.1.89).

One of the packages we installed during system installation was BIND-9. The rpm tool installed the runtime files, and configuration files in their appropriate locations.

BIND can be configured as an authoritative or master for a domain, a secondary or slave name server, or as a caching name server. A caching name server does not have any zone data files, but rather obtains all of it's information from another authoritative name server.

7.2 Customize Configuration Files

The bind-9 package comes with a cache hints file that provides the addresses of the root name servers for the internet. However, in our lab environment we do not want to access those name servers, but rather want to access a local name server for all domain name information, and to answer all queries.

Return to your workstation desktop, and double click on the PuTTY icon to begin a session. The system will respond with a window similar to the following.
Click to highlight: **LINLAB**

Click: **Load**

The system will respond with a window similar to the following:
Click your mouse pointer in the text box named “*Host Name (or IP address)*”.

Type: `9.82.56.(91-120)`

Click: Open

When prompted to enter a userid:

Enter: root

When prompted to enter a password:

Enter: linlabnn (*where nn is your team number*)
The system will respond by establishing a new SSH session for you as user root, and placing you in root’s home directory. You should receive a screen similar to the following figure.

![SSH session login](image)

**Figure 74.**

To setup named as a caching only nameserver, copying sample files installed with the distribution will be sufficient

```
Enter: cd /usr/share/doc/bind-9.7.3/sample/var/named
Enter: cp -r * /var/named/chroot/var/named/
Enter: cd ../../etc
Enter: cp -r * /var/named/chroot/etc
```
Rather than do a lot of typing, you will download from the ftp server a customized copy of the cache hints file.

Enter: cd /var/named/chroot/var/named
Enter: ftp linftp2

When prompted for a userid
Enter: anonymous

Enter: cd /pub
Enter: ascii
Enter: get named.ca
Enter: quit

In order to cause our caching name server to always obtain information from local name servers instead of the internet root name servers, we need to modify the primary configuration for named. In addition, we will need to generate a key and update named.conf with it. So before updating named.conf it will be best to generate the key and copy it to the clipboard of your terminal session.

Enter: rndc-confgen -a -r /dev/urandom -c /etc/rndc.key
Enter: cat /etc/rndc.key

Copy the portion of output inbetween the quote marks on the line with the keyword secret to the clipboard as shown below in a PuTTY session:
Enter: cd /var/named/chroot/etc
Enter: cp named.conf named.conf.original
Enter: ed named.conf
Enter: 13
Enter: .a
Enter: forward only;
Enter: forwarders { 9.82.56.121; };
Enter: .
Enter: 56,64d
Enter: .
Enter: 103,154d
Enter: .
Enter: 110,133d
Enter: .
Enter: 107
Enter: .c
Enter: secret "<paste clipboard here>";
Enter: .
Enter: 1,$w
Enter: q

Modify resolv.conf to point to your local name server now
Enter: cd /etc
Enter: cp resolv.conf resolv.conf.orig
Enter: ed resolv.conf
Enter: 3
Enter: .c
Enter: \texttt{nameserver 127.0.0.1}
Enter: .
Enter: 1,$w
Enter: q

### 7.3 Start NAMED And Test

Start the DNS daemon

Enter: \texttt{cd /etc/rc.d/init.d}
Enter: \texttt{./named start}

You should see confirmation messages that it successfully started

You can also

Enter: \texttt{tail /var/log/messages}

and look at the startup messages that were written to syslog

Test domain name services

Enter: \texttt{nslookup}

The response should be:

```
>#
```

This indicates that nslookup is communicating with your local named

Enter: \texttt{linftp2}

you should see a response indicating the fully qualified name of linftp2 (linftp2.endicott.ibm.com), and it's IP address 9.82.56.242

Enter: \texttt{endvmtst}

you should see similar results

Enter: \texttt{exit}

### 7.4 Configure NAMED To Start Automatically

In order to start named each time you boot your system, you need to make sure that a link to the proper init script exists in the default run level for your system. Rather than creating the link by hand, RedHat makes this process
very easy using the chkconfig command, or GUI based system-config-services program.

To use the GUI configuration program, you will first need to once again start the vncserver. Enter the following two commands from your PuTTY ssh session.

Enter: cd /root
Enter: vncserver

When you see the messages indicating that the server has started, go to your workstation desktop, and double click on the vncviewer icon.

The workstation will respond with a small panel similar to the following figure:

![New TightVNC Connection](image)

*Figure 75.*

In the VNC server box enter the IP address of your Linux virtual machine:

Enter: 9.82.56.(91-120):1 (note do not enter the parentheses)

Click: Connect

The workstation will respond with a password prompt window similar to the following:
Enter: `linlabnn`

The system will respond with a large window with a desktop inside. Click on the Applications pull down, and then System tools ---> Terminal. Move your cursor inside the xterm window and type at the shell prompt

Enter: `system-config-services`

The system will respond with a panel similar to the following figure.
Figure 77.

Use your mouse pointer to scroll the slider bar down until you see the entry for named.

Click on: **named**

At this point, your screen should look similar to the following figure.
Figure 78.

Click: **Enable**

The system will respond by updating the window as shown in the following figure.
Figure 79.

The window now indicates that the named service is enabled to automatically start at boot.

At this point you can exit from the configuration program.

Click: **Program**
Select: **Quit**

At this point, you should be back to the large window with the xterm. Exit from this vncviewer session by clicking on the x in the upper right corner of the large window.
The command line facility provided with RedHat to manipulate which processes are started in a particular run level is named chkconfig. You could accomplish what we just did using the GUI by entering `chkconfig --level 3 named on`. We can also use the chkconfig command to make sure that the GUI did in fact update the runlevel to include named. Return to your PuTTY ssh session to enter the following command.

Enter: `chkconfig --list named`

The system will respond with a line indicating whether named is on or off in each of the runlevels from 0 to 6. You should see a status of on for runlevel 3.

### 7.5 Test your change

Exit your PuTTY ssh session and switch back to your 3270 console session.

Enter: `shutdown -h now`

Notice the message informing you that named was shutdown. When you receive message HCPGSP2630I:

Enter: `IPL A00`

Notice the messages from named as it starts, and the [OK].

When the boot process completes and you are prompted for a login userid and password:

Enter: `root`

Enter: `linlabnn`

At this point, you should reestablish your PuTTY ssh session. Return to your workstation desktop, and double click on the PuTTY icon to begin a session. The system will respond with a window similar to the following.
Figure 80.

Click to highlight: **LINLAB**

Click: **Load**
The system will respond with a window similar to the following:

Click your mouse pointer in the text box named "Host Name (for IP address)."

Type: 9.82.56.(91-120)

Click: Open

When prompted to enter a userid:

Enter: root

When prompted to enter a password:

Enter: linlabnn (where nn is your team number)
The system will respond by establishing a new SSH session for you as user root, and placing you in root's home directory. You should receive a screen similar to the following figure.

![Screen capture of SSH session]

*Figure 82.*

**Check DNS operation:**

Try `nslookup` again, and make sure you are getting the same responses as before.
7.6 If You Have Extra Time

If you finish the activities of this chapter early, you might want to continue your exploration of BASH. This time, read about aliases in the BASH manual pages. There is a section giving general information about aliases and a syntax specific section about alias under the heading SHELL BUILTIN COMMANDS.

If you created a BASH profile in the first “Extra Time” section, then add some alias definitions to that profile and test them out. For example, add an alias named filel for the ls -al command.
Chapter 8. Configuring Samba

8.1 Introduction

Samba is an implementation of a Server Message Block (SMB) protocol server that can be run on a wide variety of Unix platforms. Microsoft clients can use this protocol to access files and printers located on a Unix system, as if that system were a native Windows server. Samba is an open source project like Linux, and is available for free. It was first developed by Andrew Tidgell in 1991 at the Australian National University in Canberra, Australia. As with Linux, the source (written in c) is available for exploration and change. With Samba installed on Linux for System z, you can now provide services to Microsoft Windows clients using the resources of your System z! For more information about Samba see http://www.samba.org.

While Samba has extensive capabilities, we will not be able to explore all of these possibilities during the time of this lab. We will only customize Samba, enough to share directories, with your Windows workstation.

8.2 Customize Samba

The main configuration file for Samba is smb.conf. Since our implementation will be rather basic, we can use the example smb.conf supplied, with only a few changes. Within the smb.conf file, comments are identified by a # symbol in column 1, or a ; in column 1. It will be easiest to perform the following commands from your PuTTY ssh session.

Enter: cd /etc/samba
Enter: cp smb.conf smb.conf.orig

Edit the /etc/samba/smb.conf file using vim (or anything else you can find that might be less painful)

Change the line with workgroup to be workgroup = LinuxLab
Change the security mode line to be security = share
Add new line below security mode line with password level = 8
Add another new line with username level = 8
Add another new line with encrypt passwords = yes
Add another new line with guest account = nobody
Add another new line with client lanman auth = yes
Add another new line with lanman auth = yes
Add another new line with client plaintext auth = yes
In the share definition block for [homes] add a line after the line beginning with writable = yes that specifies path = /home/%u
Add another new line with [tools]
Add another new line with comment = Test Tools Share
Add another new line with path = /tmp/tools
Add another new line with guest ok = yes
Add a final new line with writable = no
File these changes

Create new directory for tools share.
   Enter: mkdir /tmp/tools
   Enter: cp /root/finder /tmp/tools

Add samba credentials for user administrator.
   Enter: smbpasswd -a administrator

When prompted to enter a password enter linlabnn
When prompted to retype the password enter linlabnn

Startup the Samba daemons
   Enter: cd /etc/rc.d/init.d
   Enter: ./smb start
   Enter: ./nmb start

You should see messages indicating that smb is starting and nmb is starting
   Enter: ps -ef - look to see that smbd and nmbd are running with the -D flag

8.3 Test Use Of Samba From Linux

One of the files installed is a program named smbclient. This is a command line Samba client that can be used for some initial testing
   Enter: smbclient -L LINLABnnn -U administrator (where nn corresponds to your team number)

When prompted for a password
   Enter: linlabnn (where nn corresponds to your team number)

The resulting display should show a Domain of LINUXLAB.

Load content for testing
Enter: cd /home/administrator
Enter: ftp linftp2, when prompted for a userid enter anonymous
Enter: cd /pub
Enter: binary
Enter: get endlab.pdf
Enter: quit

8.4 Test Use Of Samba From Workstation

When your workstation was started, a prompt for userid and password was displayed. In the prompt the userid administrator was specified, with no password defined. Since we do not have a WINS server available, and we are not using Samba for that purpose (it could be used as such), we need to update the Windows lmhosts file. The lmhosts file matches IP addresses to NetBios names in a manner similar to the way an /etc/hosts file matches IP addresses to IP host names.

Open an command prompt window
Enter: cd \windows\system32\drivers\etc
Enter: copy lmhosts.sam lmhosts
Enter: edit lmhosts

Go to the bottom of the file and add a line containing
9.82.56. (91-120) linlabnn (where nn corresponds to your team number)
Enter: alt+f and select save, then alt+f and select exit

From the command prompt window
Enter: net use z: \linlabnn\administrator (where nn corresponds to your team number)

If you are prompted for a password,
Enter: linlabnn (where nn corresponds to your team number)

When the command completes, you can enter z: at the DOS prompt, and then create a file, and display the file. Everything you create on this drive is created in the home directory of administrator on your Linux for System z system

Use the start and programs menu to start adobe acrobat
Select open from the file pulldown, and select z: as the drive
Select endlab.pdf on the z: and view a pdf of the class notes for this lab
8.5 Additional Configuration Programs

Samba has an outstanding web based administration tool named SWAT (Samba Web Administration Tool). This facility was new with Samba 2.0.0, consisting of a miniature web server and CGI scripting application, designed to run from inetd. Its primary function is to provide access to the smb.conf file, and enable updates to this file. It also provides a very nice remote status capability.

Unfortunately, RedHat does not include SWAT binaries with the Samba rpms. Instead RedHat used to include a GUI program for maintaining the smb.conf file named system-config-samba. With RHEL 6.1, this configuration tool is no longer available. All customization of Samba needs to be done via the command line, editing the smb.conf file as we have just done.

8.5.1 Test Tools Share

Return to the workstation and open a command prompt window.

Enter: net use p: \linlabnn\tools (where nn corresponds to your team number)

If you are prompted for a password,

Enter: linlabnn (where nn corresponds to your team number)

When the command completes, you can enter p: at the DOS prompt, and view the file you placed into /tmp/tools earlier named finder, using the DOS dir command.

8.6 Setup Samba To Automatically Start At Boot

Return to your PuTTY SSH session to perform the following tasks.

Create a link to the startup script in the appropriate rc subdirectory.

Enter: chkconfig --level 3 smb on
Enter: chkconfig --level 3 nmb on

Exit from your PuTTY ssh session and switch to your 3270 console session. In order to test and make sure Samba will automatically start, you will need to shutdown your Linux virtual machine.

Enter: shutdown -h now

When you receive message HCPGSP2630I,
Enter: `ipl A00`  
Watch for message "Starting SMB services"  
When prompted for a userid,  
   Enter: `root`  
When prompted for a password,  
   Enter: `linlabnn`  
(where nn corresponds to your team number)  
At this point, you should reestablish your PuTTY ssh session. Return to your workstation desktop, and double click on the PuTTY icon to begin a session. The system will respond with a window similar to the following.
Click to highlight: **LINLAB**

Click: **Load**

The system will respond with a window similar to the following:
Click your mouse pointer in the text box named “Host Name (or IP address)”.  
Type: 9.82.56.(91-120)
Click: Open

When prompted to enter a userid:
Enter: root

When prompted to enter a password:
Enter: linlabnn (where nn is your team number)
The system will respond by establishing a new SSH session for you as user root, and placing you in root's home directory. You should receive a screen similar to the following figure.

Figure 85.

Enter the following commands from your PuTTY ssh session.

Enter: `ps -ef` - look for smbd -D and nmbd -D
Enter: `smbclient -L LINLABnn -U administrator` (where nn corresponds to your team number)

When prompted for a password:

Enter: `linlabnn` where nn is your team number
Look for same response as earlier, this will verify that your Samba system was correctly started.

Again, sit back and take a deep breath. You have reached the third checkpoint.

8.7 Checkpoint 3

You have just finished customizing BIND-9 and creating a local caching only name server for DNS resolution. To make your Linux for System z system a zone master name server involves only adding additional zone files, and perhaps taking the "forward first" and "forwarders" directives out of the named.conf file.

Your server also now provides disk sharing resources for Windows based workstation clients through Samba. You have customized Samba and configured it to allow sharing of home directories, and a common tools directory. As part of customizing Samba, you configured your Linux for System z system to automatically start Samba when the system boots. Congratulations!
Chapter 9. Customizing Apache

9.1 Introduction

During the initial installation we installed the Apache Web server. In this chapter, we will perform some customization that allows use of the Web Server, then use the new cmsfs-fuse file system support to serve content from a z/VM CMS formatted minidisk. This new file system driver makes use of the fuse (file system in user space) support added to the kernel in previous releases. Prior to RHEL 6.1 we would use NFS as the mechanism to access content on a z/VM CMS formatted minidisk. Now with the new cmsfs-fuse support we can mount a CMS formatted minidisk in r/w mode directly into the file system tree. This allows for serving content from the CMS formatted minidisk, and modification of html files from within the Linux environment.

The Apache server has proven to be one of the most popular web servers on the Internet. Some estimates go as high as stating that more than 50% of the Web servers in the world are using Apache. The Apache project was started by people running the original National Center for Super Computing Applications (NCSA) web server. When the primary developer for that server left the NCSA, people using the NCSA server began exchanging patches. Before long the group realized they needed a forum to manage the patches, and the Apache project was born. Apache is a very full function web server, that performs quite well. Many people contribute to the project, as well as vendors such as IBM. It continues to evolve toward meeting today's e-business needs at a rapid pace. For more information see http://www.apache.org

It would be best to perform the following tasks from your PuTTY ssh session.

9.2 Customize Apache Configuration File

Before the Apache server can be used you must modify the configuration file (httpd.conf) slightly. RedHat supplies a GUI configuration program named system-config-httpd. In order to use this program you need to establish a vncserver environment again, and use the vncviewer (system-config-httpd is an X windows program). However, this program is not very comprehensive at the moment, and does not allow for changes that we want to make to the configuration file. In addition, it does not support both manual updates to the configuration file, and updates from the GUI program.
Thus, we will make manual updates to the httpd.conf file, rather than use the GUI program. If you would like to see the GUI program, startup vncserver again, and use the vncviewer from your workstation to establish a vnc session with the server. From the xterm shell prompt type system-config-httpd, and the GUI program will be started.

Use your PuTTY ssh session to perform following tasks.

Comments within httpd.conf have a # symbol in column 1

Enter: cd /etc/httpd/conf
Enter: cp httpd.conf httpd.conf.orig

Use vim to edit the httpd.conf file:

Uncomment the line beginning with \#ServerName and change www.example.com:80 to linlabnn.endicott.ibm.com:80

Uncomment the block beginning with <Location /server-status>
Change the line in this block beginning with "Deny from all" to "Allow from all" (note: do not include the quotes)

Delete the line beginning with "Allow from .your_domain.com"

Uncomment the block beginning with <Location /server-info>
Change the line in this block beginning with "Deny from all" to "Allow from all"

Delete the line beginning with "Allow from .your_domain.com"

File these changes

---

**9.3 Run Apache**

You are now ready to start the Apache web server. The document root for Apache is specified in the httpd.conf file, and is /var/www/html. Files within this directory tree will be served when you connect to your server.

Enter: cd /etc/rc.d/init.d
Enter: ./httpd start
Enter: ps -ef - look for multiple occurrences of /usr/sbin/httpd

You are now ready to connect with your web browser

On your windows workstation start a browser (IE or Firefox).
In the location field enter http://9.82.56.(91-120)/
You should see the RedHat Enterprise Linux Test Page, indicating that the server was installed.

Change to url \texttt{http://9.82.56.(91-120)/server-info}

This will display a page showing information about your Apache server

Change to url \texttt{http://9.82.56.(91-120)/server-status}

This will display a page showing the status of your Apache server

9.4 Change Content

At this point, we will show how easy it is to serve some existing content that might reside on your VM system from your new Apache server. To do this, we are going to make use of the new cmsfs-fuse file system driver.

cmsfs-fuse is part of the s390utils package, but is not yet included in the base for RHEL 6.1. It is found in the optional channel on rhn (RedHat Network). So the first thing we will need to do is install the new file system driver.

Use your PuTTY ssh session to enter the following commands.

Enter: \texttt{cd /root}

Enter: \texttt{ftp linftp2}, when prompted for a userid specify \texttt{anonymous}

Enter: \texttt{cd /pub/rhel61}

Enter: \texttt{binary}

Enter: \texttt{get s390utils-cmsfs-fuse-1.8.2-36.el6.s390x.rpm}

Enter: \texttt{quit}

Install cmsfs-fuse

Enter: \texttt{rpm -ivh s390utils-cmsfs-fuse-1.8.2-36.el6.s390x.rpm}

Since cmsfs-fuse relies upon the fuse kernel support being already loaded, we need to make sure that a modprobe for fuse is done whenever Linux boots. To do this we will create/modify \texttt{rc.modules}.

Enter: \texttt{cd /etc}

Enter: \texttt{echo "modprobe fuse" >> rc.modules}

The command above will place the text \texttt{modprobe fuse} at the end of the \texttt{rc.modules} file if it already exists, or will create the \texttt{rc.modules} file with the text if it does not exist. In order to test mount the file system before we reboot, we need to manually load the fuse driver now.
Enter: `modprobe fuse`

The system will respond by just displaying a shell prompt following the command execution.

Since the minidisk we want to use is not currently managed by the Linux dasd driver, we need to bring it online, and then make sure that it is always brought online when we reboot Linux.

Enter: `chccwdev -e 0291`

The system will respond by registering this new device with the Linux dasd device driver. In addition, udev will build entries in the device file system so that the new device can be used. In the /dev/disk/by-path directory an entry will be created that allows us to make sure we access this particular dasd in any reference. The entry will be of the form ccw-0.0.0291. Rather than use dasda, or dasdb which are not guaranteed to reference the same device across reboots, we will use the persistent name ccw-0.0.0291.

To make sure this dasd device is available each time we boot Linux, we need to add an entry to the dasd.conf file in /etc.

Enter:
```
  cd /etc
  ed dasd.conf
  $a
  0.0.0291 use_diag=0 readonly=0 erplog=0 failfast=0
  .
  1,$p
  1,$w
  q
```

We are now ready to test mount the file system.

Enter: `cmsfs-fuse -t -o allow_other /dev/disk/by-path/ccw-0.0.0291 /var/www/html`

The system will respond with a shell prompt if the command is successful.

The above command causes the cmsfs-fuse file system driver to mount the file system on the CMS formatted minidisk (291) at the mount point of /var/www/html. This mount point is specified in the httpd.conf file as the current document root for apache. In addition, the file system is mounted with automatic ebcdic to ascii translation based upon the file type portion of a CMS file name, and is mounted with access permissions that allow any user
to view files in the file system. Automatic translation is based upon the CMS file types listed in the `/etc/cmsfs-fuse/filetypes.conf` file. If you want to include additional file type names you can modify this file.

To make sure your fuse file system is mounted use the Linux `df` command.

Enter: `df`

The system will respond with output similar to the figure below:

```
[root@linlab10 ~]# df
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/sda1              5159552   2975680   1921780  61% /
tmpfs                   509360         0    509360   0% /dev/shm
/dev/disk/by-path/ccw-0.0.0291 1440        32      1408   3% /var/www/html
[root@linlab10 ~]#
```

Check new content.

Use your browser to view `http://9.82.56.(91-120)/`

You should now see new content instead of the RedHat Test Page.

As mentioned in the introduction to this chapter, previous versions of the workshop used NFS as the mount vehicle, and placed the minidisk mount point in a subdirectory of `/var/www/html`. The current `INDEX.HTML` reflects this setup, so if you click on the words “Link to page on your NFS mounted minidisk” you will get an error. To illustrate the r/w nature of this mount and the ease with which you can change files, we will correct the `INDEX.HTML` file now.

Enter: `cd /var/www/html`
Enter: `ed INDEX.HTML`
Enter: `13`
Enter: `.c`
When you refresh your browser, you should now be able to click on the hyperlink text "Link to second page".

If you change the content on your linlabnn 291 mdisk, the changes will show up when you next point your browser at the url.

9.5 Integrate NFS Mount Into Linux Startup

So the Apache server can automatically serve the new content when the system is rebooted, we need to make sure that the cmsfs-fuse file system is mounted during the startup process. cmsfs-fuse is not yet intergrated into the mount command as a file system type so we cannot just put the file system into /etc/fstab. Rather we will need to create a startup script that will run during boot before the httpd daemon (apache) is started, and will unmount after httpd is stopped.

From your PuTTY ssh session:

```
Enter: cd /etc/rc.d/init.d
Enter: ed cmsfs_mount
Note: this is a new file that does not exist yet.
Enter: .a
Enter: #!/bin/sh
Enter: # chkconfig: 35 21 16
Enter: # description: Issue cmsfs-fuse mount command
Enter: case "$1" in
Enter: start)
Enter: cmsfs-fuse -t -o allow_other
    /dev/disk/by-path/ccw-0.0.0291 /var/www/html
Enter: ;;
Enter: stop)
Enter: fusermount -u /var/www/html
Enter: ;;
Enter: esac
Enter: .
Enter: 1,$p - review to make sure everything entered as above
```

- review to make sure everything entered as above
Enter: 1, $w
Enter: q
Enter: chmod +x cmsfs_mount
Enter: chkconfig --level 3 cmsfs_mount on
Enter: chkconfig --level 3 httpd on

9.6 Test System Changes

At this point, you should test your changes to make sure that the cmsfs-fuse mount works properly at boot time, and that the new Apache server starts at boot time. Exit your PuTTY ssh session and switch to your 3270 console session.

Enter: shutdown -h now

When you receive message HCPGSP2630I,

Enter: IPL A00

Note: on startup you should now see that starting httpd receives an OK. You should also see that your cmsfs_mount script (executing a cmsfs-fuse) ran OK.

When prompted for a userid and password,

Enter: root
Enter: linlabnn

At this point, you should reestablish your PuTTY ssh session. Return to your workstation desktop, and double click on the PuTTY icon to begin a session. The system will respond with a window similar to the following.
Figure 86.

Click to highlight: **LINLAB**

Click: **Load**

The system will respond with a window similar to the following:
Figure 87.

Click your mouse pointer in the text box named “**Host Name (or IP address)**”.

Type: **9.82.56.(91-120)**

Click: **Open**

When prompted to enter a userid:

Enter: **root**

When prompted to enter a password:

Enter: **linlabnn (where nn is your team number)**
The system will respond by establishing a new SSH session for you as user root, and placing you in root's home directory. You should receive a screen similar to the following figure.

![Login screen](image)

**Figure 88.**

Enter the following commands from your PuTTY ssh session.

Enter: `df` - look to see that the cmsfs-fuse file system was mounted at `/var/www/html`

Enter: `ps -ef` - look to see that your Apache daemons are running
Test your web server again by using your workstation browser

9.7 Apache Extra Credit

If you have finished the Apache assignment early, and would like to work with Apache further, try the following extra credit assignment. Read the Apache online documentation, and enable your Apache server to execute Common Gateway Interface (CGI) programs. To test your customization, create a simple PERL CGI program that returns Hello World when executed.

```perl
#!/usr/bin/perl
print "Content-type: text/html\n\n";
print <<HTML;

<html>
<body>
<h1> hello world </h1>
</body>
</html

HTML
```

Figure 89. Apache Extra Credit Hint
10.1 Introduction

KDE is a GUI Desktop Environment for Unix workstations. It is an OpenSource project that was initiated by Matthias Ettrich in 1996. Development for KDE takes place on the Internet, and is contributed to by a wide variety of programmers. Since KDE is an OpenSource project, all source code is available for KDE. KDE is licensed under the GNU license. KDE is available at no cost.

Most of the KDE developers use Linux, however, the desktop is available on a wide range of systems. A partial list includes the following:

- Linux
- Solaris
- FreeBSD
- IRIX
- HP-UX

KDE provides a complete desktop environment, including a file manager, a window manager, a help system, a configuration system, many tools and utilities, and an ever increasing number of applications, including but not limited to mail and news clients, drawing programs, a PostScript and DVI viewer and so forth. See http://www.kde.org for more information.

In this last segment we will configure the K Display Manager (KDM) so we can receive a GUI login panel, when contacting a specific vncserver port instead of the user specific minimal desktop we have been receiving.

10.2 Configure KDM

The KDE desktop comes with it's own display manager named KDM. The purpose of a display manager is to present a graphical login window on a local or remote XServer, and then process any login requests, validating the userid and password and establishing the XWindows environment for that login. This is the most convenient way to use KDE in the System z environment. The following steps will walk you through the process of configuring KDM for use with vncserver.

Please use your PuTTY ssh session for the following commands.

Enter: cd /etc
Enter: cp inittab inittab.save
Enter: `vim /etc/inittab`

**Change** the record beginning with `id:3` to `id:5` and FILE this change.

Enter: `cd /etc/sysconfig`

Enter: `ed desktop` (note: this is a new file)

Enter: `.a`

Enter: `DISPLAYMANAGER="KDE"`

Enter: `.`

Enter: `1,$w`

Enter: `q`

Enter: `cd /etc/kde/kdm`

Enter: `vim Xaccess`

**Uncomment** line beginning with `#*` where the comment says any host can get a login window. Make sure you leave the * in column 1

**Uncomment** line beginning with `#*` where the comment reads CHOOSER BROADCAST. Make sure you leave the * in column 1

FILE these changes

Enter: `vim kdmrc`

Under the heading `[Xdmcp]` Change Enable=false to Enable=true

FILE these changes

In order to start Xvnc automatically when a request comes into port 5901, it is best to run Xvnc out of xinetd. In order to do that, a service file needs to be created in the /etc/xinetd.d directory. To reduce typing, we will download one from the ftp server.

Enter: `cd /etc/xinetd.d`

Enter: `ftp linftp2`, when prompted for a userid specify anonymous

Enter: `cd /pub`

Enter: `get vnc`

Enter: `quit`

Up to this point, we have been running out of runlevel 3. Since we are changing to runlevel 5 (graphical runlevel) now, we need to make sure that all of the services we have enabled in previous chapters are enabled also for runlevel 5.

Enter: `chkconfig --level 5 xinetd on`

Enter: `chkconfig --level 5 named on`

Enter: `chkconfig --level 5 httpd on`

Enter: `chkconfig --level 5 smb on`

Enter: `chkconfig --level 5 nmb on`
Enter: `chkconfig --level 5 cmsfs_mount on`

We could manually start kdm and xinetd at this point, however, it is easier to just reboot Linux. In addition, this will verify that all of the changes were properly made. At this point, exit your PuTTY ssh session and switch to your 3270 console session.

Enter: `shutdown -h now`

When you receive message HCPGSP2630I,

Enter: `IPL A00`

When Linux completes the boot process and you are prompted for a userid and password,

Enter: `root`

Enter: `linlabnn`

You can now test using KDE from vncviewer.

Return to your workstation desktop, and double click on the vncviewer icon.

The workstation will respond with a small panel similar to the following figure:

![New TightVNC Connection](image)

*Figure 90.*

In the VNC server box enter the IP address of your Linux virtual machine, and screen 1 as in the past:

Enter: `9.82.56.(91-120):1` (note do not enter the parentheses)

Click: `Connect`
The system will respond with a kdm login prompt window similar to the following:

Figure 91.

Enter: root
Enter: linlabnn
Press: Enter key

The system will respond by logging in user root and returning a kde desktop in the vncviewer window.

When you logout from the KDE desktop, your vnc session will end.

Congratulations, you are done with the workshop!
Chapter 11. Summary

OK, take a moment to let all of this sink in! You have just:

- Started with an empty virtual machine, and using the initial ram disk created and populated a Linux for System z system with RedHat Enterprise Linux 6.1.

- Following the installation on FCP attached SCSI disks, you added two additional LUNs to your system, and configured them as a multipath logical volume.

- You have connected your Linux for System z system to the network using a virtual network interface card connection to a z/VM QDIO Guest LAN.

- You have created swap space for your Linux guest on a VM virtual disk (no real I/O when Linux swaps!)

- You have configured netfilter code using the iptables utility to create a very simple firewall.

- You have done some system administration, added users, monitored disk usage, as well as configured BIND-9, as a caching only DNS.

- You have customized Samba to share System z resources with your Windows workstation clients. You have also done the system administration to make sure that Samba properly starts at boot time.

- You have customized Apache, and done the system work necessary to allow the Apache server to access content in a VM file system. You have also done the system administration necessary to properly start this server at boot time (along with mounting a cmsfs-fuse file system).

- You have customized kdm for use with Xvnc allowing for graphical logins, and use of the kde desktop applications with vncviewer.

- What you now have, is a server virtual machine, that can be autologged, and used for disk sharing, web serving, and as a firewall. Congratulations, you are a guru!
Appendix A. VIM Reference

Before you enter any commands such as /pattern to search for a string, or w to save the file, you must first enter a : (colon). This will place you into command mode. Other commands such as the cursor movement keys, or commands to insert, add, replace, or delete characters do not require entering a colon first. In order to leave insert, add, or replace mode, you must press the Esc key. At this point you can then use the cursor movement keys (j, k, l, h), or enter command mode (press :).

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vim file</td>
<td>Begin editing a file</td>
</tr>
<tr>
<td>:</td>
<td>Enter command mode</td>
</tr>
<tr>
<td>Esc</td>
<td>Exit from a particular command</td>
</tr>
<tr>
<td>j</td>
<td>Move cursor down one line</td>
</tr>
<tr>
<td>k</td>
<td>Move cursor up one line</td>
</tr>
<tr>
<td>l</td>
<td>Move cursor to the right</td>
</tr>
<tr>
<td>h</td>
<td>Move cursor to the left</td>
</tr>
<tr>
<td>Ctrl G</td>
<td>Display current line number</td>
</tr>
<tr>
<td>:set number</td>
<td>Show line numbers</td>
</tr>
<tr>
<td>nG</td>
<td>Move to line number n</td>
</tr>
<tr>
<td>G</td>
<td>Move to last line in file</td>
</tr>
<tr>
<td>i</td>
<td>Insert text before cursor (insert)</td>
</tr>
<tr>
<td>a</td>
<td>Insert text after cursor (add)</td>
</tr>
<tr>
<td>R</td>
<td>Type over characters (change)</td>
</tr>
<tr>
<td>x</td>
<td>Delete character</td>
</tr>
<tr>
<td>dd</td>
<td>Delete current line</td>
</tr>
<tr>
<td>:)1,3d</td>
<td>Delete multiple lines (e.g. delete lines 1 through 3)</td>
</tr>
<tr>
<td>:$,d</td>
<td>Delete from current line to end of file</td>
</tr>
<tr>
<td>:1,3m24</td>
<td>Move multiple lines (e.g. move lines 1 through 3 after 24)</td>
</tr>
<tr>
<td>:1,3co24</td>
<td>Copy multiple lines (e.g. copy lines 1 through 3 after 24)</td>
</tr>
<tr>
<td>/pattern</td>
<td>Search forward for pattern</td>
</tr>
<tr>
<td>?pattern</td>
<td>Search backward for pattern</td>
</tr>
<tr>
<td>:w</td>
<td>Save file</td>
</tr>
</tbody>
</table>
:\q Quit file
:\q! Quit file discarding changes
:\1,$w newfile Write edit contents to a new file (save under new name)
:\x Save file and quit
Appendix B. ED Reference

The ed command starts a line editing program, the ed editor, that works on only one file at a time by copying it into a temporary edit buffer, and making changes to that copy. The ed editor makes the changes you specify in that buffer. It does not alter the file itself until you use the write subcommand. The ed editor is the only editor you can use from the Linux virtual machine console, since the device interface is a line mode 3215 interface. An ed editor subcommand consists of zero, one, or two addresses followed by a single letter subcommand, possibly followed by parameters to that subcommand. The address specifies one or more lines in the buffer. The ed editor allows you only to edit the current line, unless you address another line in the buffer. The ed editor operates in one of two modes, command mode, and text input mode. In command mode, the editor recognizes and operates on subcommands. When you start the editor it is in command mode. You can also enter command mode at any time by entering a lone period at the beginning of a line. In text input mode, the editor allows you to enter text into the file buffer. In text input mode, the editor does not recognize subcommands. You enter text input mode by using the a, c, or i subcommand. You leave text input mode by entering a lone period at the beginning of a line.

- ed: Begin editing a file
- a: Add text after the addressed line
- i: Insert text before the addressed line
- c: Change the addressed lines (deletes current line)
- d: Delete addressed line
- .: Enter subcommand mode
- 1,$p: Print the addressed lines (e.g. lines 1 through end ($))
- digit: Display a particular line
- =: Display current line number
- $: Display last line in file
- 1,$n: Display addressed lines and show line numbers
- /pattern/n: Search for a pattern and display line number with hit
- 1,$w: Write file to disk
- Q: Quit without changing file
q Quit and check to make sure changes written first
## Appendix C. FCP SCSI LUN ID Table

<table>
<thead>
<tr>
<th>Userid</th>
<th>Install LUN</th>
<th>Addtl LUN1</th>
<th>Addtl LUN2</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINLAB01</td>
<td>0x0040</td>
<td>0x0000</td>
<td>0x0001</td>
</tr>
<tr>
<td>LINLAB02</td>
<td>0x0041</td>
<td>0x0002</td>
<td>0x0003</td>
</tr>
<tr>
<td>LINLAB03</td>
<td>0x0042</td>
<td>0x0004</td>
<td>0x0005</td>
</tr>
<tr>
<td>LINLAB04</td>
<td>0x0043</td>
<td>0x0006</td>
<td>0x0007</td>
</tr>
<tr>
<td>LINLAB05</td>
<td>0x0044</td>
<td>0x0008</td>
<td>0x0009</td>
</tr>
<tr>
<td>LINLAB06</td>
<td>0x0045</td>
<td>0x000a</td>
<td>0x000b</td>
</tr>
<tr>
<td>LINLAB07</td>
<td>0x0046</td>
<td>0x000c</td>
<td>0x000d</td>
</tr>
<tr>
<td>LINLAB08</td>
<td>0x0047</td>
<td>0x000e</td>
<td>0x000f</td>
</tr>
<tr>
<td>LINLAB09</td>
<td>0x0048</td>
<td>0x0010</td>
<td>0x0011</td>
</tr>
<tr>
<td>LINLAB10</td>
<td>0x0049</td>
<td>0x0012</td>
<td>0x0013</td>
</tr>
<tr>
<td>LINLAB11</td>
<td>0x004a</td>
<td>0x0014</td>
<td>0x0015</td>
</tr>
<tr>
<td>LINLAB12</td>
<td>0x004b</td>
<td>0x0016</td>
<td>0x0017</td>
</tr>
<tr>
<td>LINLAB13</td>
<td>0x004c</td>
<td>0x0018</td>
<td>0x0019</td>
</tr>
<tr>
<td>LINLAB14</td>
<td>0x004d</td>
<td>0x001a</td>
<td>0x001b</td>
</tr>
<tr>
<td>LINLAB15</td>
<td>0x004e</td>
<td>0x001c</td>
<td>0x001d</td>
</tr>
<tr>
<td>LINLAB16</td>
<td>0x004f</td>
<td>0x001e</td>
<td>0x001f</td>
</tr>
<tr>
<td>LINLAB17</td>
<td>0x0050</td>
<td>0x0020</td>
<td>0x0021</td>
</tr>
<tr>
<td>LINLAB18</td>
<td>0x0051</td>
<td>0x0022</td>
<td>0x0023</td>
</tr>
<tr>
<td>LINLAB19</td>
<td>0x0052</td>
<td>0x0024</td>
<td>0x0025</td>
</tr>
<tr>
<td>LINLAB20</td>
<td>0x0053</td>
<td>0x0026</td>
<td>0x0027</td>
</tr>
<tr>
<td>LINLAB21</td>
<td>0x0054</td>
<td>0x0028</td>
<td>0x0029</td>
</tr>
<tr>
<td>LINLAB22</td>
<td>0x0055</td>
<td>0x002a</td>
<td>0x002b</td>
</tr>
<tr>
<td>LINLAB23</td>
<td>0x0056</td>
<td>0x002c</td>
<td>0x002d</td>
</tr>
<tr>
<td>LINLAB24</td>
<td>0x0057</td>
<td>0x002e</td>
<td>0x002f</td>
</tr>
<tr>
<td>LINLAB25</td>
<td>0x0058</td>
<td>0x0030</td>
<td>0x0031</td>
</tr>
<tr>
<td>Userid</td>
<td>Install LUN</td>
<td>Addtl LUN1</td>
<td>Addtl LUN2</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>LINLAB26</td>
<td>0x0059</td>
<td>0x0032</td>
<td>0x0033</td>
</tr>
<tr>
<td>LINLAB27</td>
<td>0x005a</td>
<td>0x0034</td>
<td>0x0035</td>
</tr>
<tr>
<td>LINLAB28</td>
<td>0x005b</td>
<td>0x0036</td>
<td>0x0037</td>
</tr>
<tr>
<td>LINLAB29</td>
<td>0x005c</td>
<td>0x0038</td>
<td>0x0039</td>
</tr>
<tr>
<td>LINLAB30</td>
<td>0x005d</td>
<td>0x003a</td>
<td>0x003b</td>
</tr>
</tbody>
</table>
Appendix D. Special notices

The information in this publication is not intended as the specification of any programming interfaces that are provided by Linux for System z. See the PUBLICATIONS section of the IBM Programming Announcement for Linux for System z for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.
Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

- AFP
- AIX
- APPN
- AS/400
- AT
- C/MVS
- CICS
- CT
- CUA
- Current
- DB2
- DB2 Connect
- DB2 Universal Database
- DRDA
- ECKD
- ESCON
- FICON
- Home Director
- Hummingbird
- IBM
- IBM.COM
- Language Environment
- Linux for S/390®
- MQSeries
- Multiprise
- Netfinity
- Network Station
- Nways
- OpenEdition
- OS/2
- OS/390
- OS/400
- Parallel Sysplex
- PR/SM
- RACF
- RAMAC
- RETAIN
- RMF
- RS/6000
- S/370
- S/390
- SP
- System/370
- System/390
- TCS
- ThinkPad
- VisualAge
- VM/ESA
- VSE/ESA
- VTAM
- WebSphere
- XT
- 3090
- 400

The following terms are trademarks of other companies:

- Tivoli, Manage. Anything. Anywhere., The Power To Manage., Anything. Anywhere., TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

- C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.
Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Linux is a registered trademark of Linux Torvalds

Other company, product, and service names may be trademarks or service marks of others.