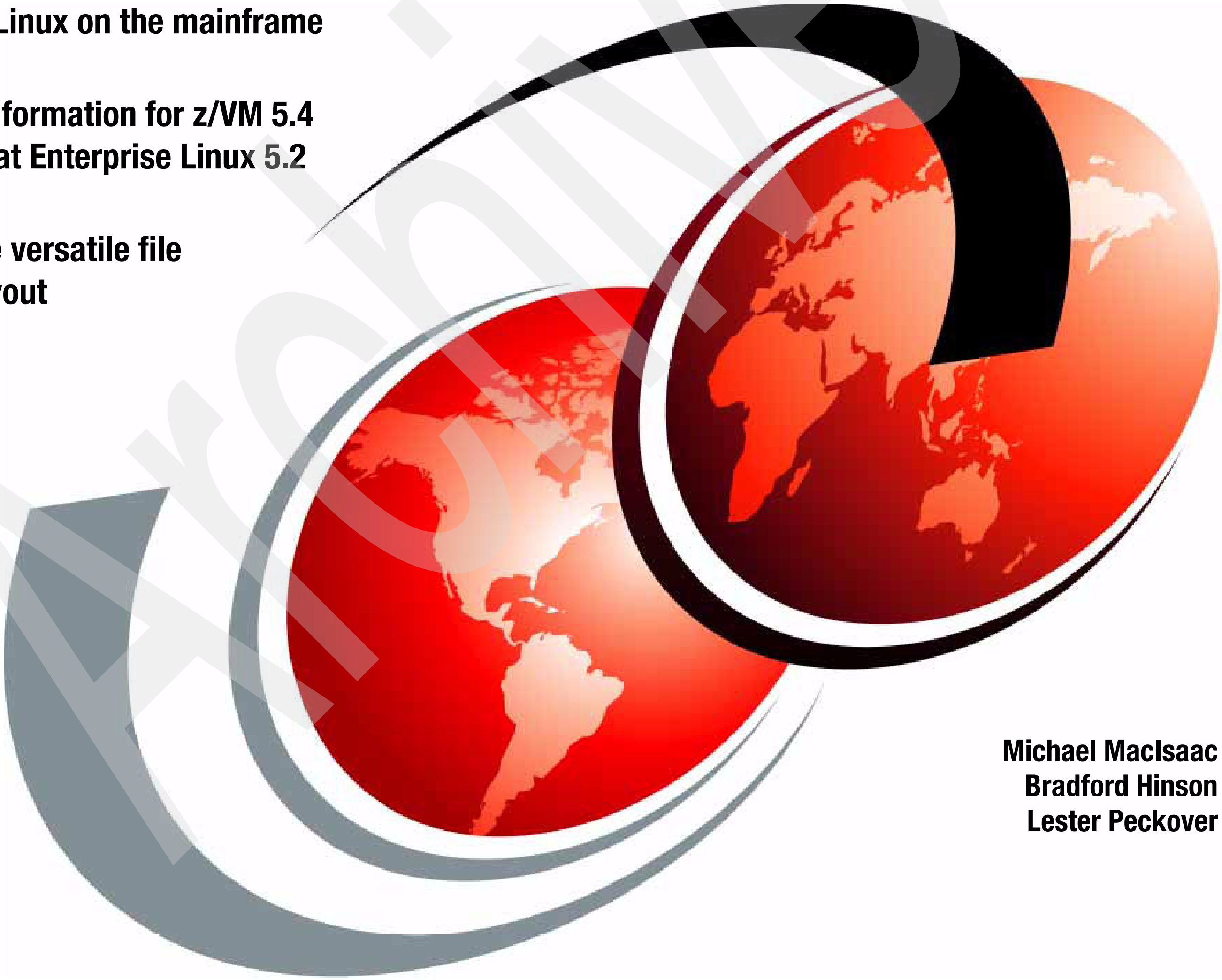
# z/VM and Linux on IBM System z The Virtualization Cookbook for Red Hat Enterprise Linux 5.2

Hands-on instructions for installing z/VM and Linux on the mainframe

Updated information for z/VM 5.4 and Red Hat Enterprise Linux 5.2

New, more versatile file system layout



Michael MacIsaac
Bradford Hinson
Lester Peckover

Redbooks

ibm.com/redbooks

International Technical Support Organization

# The Virtualization Cookbook for Red Hat Enterprise Linux 5.2

October 2008

SG24-7492-00

**Note:** Before using this information and the product it supports, read the information in “Notices” on page ix.

**First Edition (October 2008)**

This edition applies to Version 5, Release 4, Modification 0 of z/VM (product number 5741-A05) and Version 5, Release 2 of Red Hat Enterprise Linux.

# Contents

**Notices** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . ix
Trademarks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .x

**Preface** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . xi
The team that wrote this book . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . xi
History. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .xii
Chapters and appendixes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . xiii
Summary of changes in October 2008 version . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . xiv
Conventions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .xv
Comments welcome. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .xv

**Chapter 1. Introduction to z/VM and Linux** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1
1.1 What is virtualization . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
1.2 This book's approach . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
1.3 Choices and decisions made in this book . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
1.4 Infrastructure design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
1.5 Usability tests performed for this book . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

**Chapter 2. Planning** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
2.1 Bill of materials . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
2.1.1 Hardware resources . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
2.1.2 Software resources . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
2.1.3 Networking resources . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
2.2 z/VM conventions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
2.2.1 Volume labeling convention . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
2.2.2 Backup file naming convention . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
2.2.3 The command retrieve convention . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
2.3 Disk planning. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
2.4 Memory planning. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
2.5 Password planning . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
2.6 Planning worksheets . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
2.6.1 z/VM resources used in this book . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
2.6.2 z/VM DASD used in this book. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
2.6.3 Linux resources used in this book. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15
2.6.4 Linux user IDs used in this book . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15
2.7 Blank worksheets . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
2.7.1 z/VM resources worksheet . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
2.7.2 z/VM DASD worksheet . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
2.7.3 Linux resources worksheet . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
2.7.4 Linux user ID worksheet . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18

**Chapter 3. Configuring a desktop machine** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19
3.1 PuTTY: a free SSH client for Windows . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
3.2 Setting up a VNC client . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
3.2.1 Downloading and running RealVNC . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
3.2.2 Customizing RealVNC . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
3.3 3270 emulators . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25

**Chapter 4. Installing and configuring z/VM**. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27

4.1 Installing z/VM from DVD 28
4.1.1 Booting z/VM from DVD 28
4.1.2 Copying a vanilla z/VM system to DASD 31
4.1.3 IPL the vanilla z/VM from DASD 35
4.1.4 Completing the z/VM installation 36
4.2 Configuring TCP/IP 38
4.2.1 Use the IPWIZARD tool 38
4.3 Configuring the XEDIT profile 41
4.4 Customizing the SYSTEM CONFIG file 42
4.5 Configuring TCP/IP to start at IPL time 44
4.5.1 Renaming the TCPIP configuration file 45
4.5.2 Copy the PROFILE XEDIT file 45
4.5.3 Configuring the FTP server 46
4.5.4 Shutting down and reIPLing the system 47
4.6 Adding paging volumes 47
4.6.1 Formatting the paging volumes 48
4.6.2 Formatting DASD for minidisks 50
4.6.3 Updating the SYSTEM CONFIG file 52
4.6.4 Testing the changes 53
4.7 Creating a user ID for common files 54
4.7.1 Define the user in the USER DIRECT file 54
4.7.2 Logging and customizing the new user ID 56
4.7.3 Copying a PROFILE XEDIT 57
4.7.4 Creating a PROFILE EXEC 57
4.7.5 Copying files associated with this book to LNXMAINT 58
4.8 Customizing system startup and shutdown 59
4.8.1 Configuring the AUTOLOG1 PROFILE EXEC 59
4.8.2 Testing the changes 60
4.9 Addressing z/VM security issues 60
4.9.1 VM security products 61
4.9.2 High level z/VM security 61
4.9.3 Changing passwords in USER DIRECT 61
4.10 Backing up your z/VM system to tape 62
4.11 Relabeling the system volumes 63
4.11.1 Modifying labels in the SYSTEM CONFIG file 64
4.11.2 Modifying labels in the USER DIRECT file 66
4.11.3 Changing the labels on the five volumes 66
4.11.4 Shutting down your system and restarting it 67
4.12 Restoring your z/VM system from tape 68

**Chapter 5. Servicing z/VM** 71
5.1 Applying a PTF 72
5.1.1 Determining whether a PTF has been applied 72
5.1.2 Getting service using Internet FTP 73
5.1.3 Downloading the service to z/VM 74
5.1.4 Receiving, applying, and building service 75
5.1.5 Putting the service into production 76
5.2 Applying a Recommended Service Upgrade 77
5.2.1 Getting service from the Internet 77
5.2.2 Downloading the service to z/VM 78
5.2.3 Receiving, applying, and building the service 80
5.2.4 Putting the service into production 81
5.3 Determining the z/VM service level 82

**Chapter 6. Configuring a Network File System server for RHEL 5.2** . . . . . . . . . . . . . . 85
6.1 Installing Linux on the PC . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 86
6.2 Downloading files associated with this book . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 86
6.3 Setting up a RHEL 5.2 install tree . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 86
6.3.1 Copying from physical DVD . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 87
6.3.2 Verifying the ISO image . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 87
6.3.3 Copying the DVD contents . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 88
6.3.4 Building the repository directory . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 88
6.4 Enabling the NFS server . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 88

**Chapter 7. Installing RHEL 5.2 on the controller** . . . . . . . . . . . . . . . . . . . . . . . . . . . . 91
7.1 Installing the controller . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 92
7.1.1 Creating the user ID LNXINST . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 92
7.1.2 Adding LNXINST to AUTOLOG1's PROFILE EXEC . . . . . . . . . . . . . . . . . . . . . 94
7.1.3 Preparing LNXINST bootstrap files. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 94
7.1.4 Beginning the Linux installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 97
7.1.5 Stage 1 of the RHEL 5.2 installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 99
7.1.6 Stage 2 of the RHEL 5.2 installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 101
7.1.7 Booting your new Linux system from disk. . . . . . . . . . . . . . . . . . . . . . . . . . . 108
7.2 Configuring the controller . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 109
7.2.1 Copying files to the controller . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 109
7.2.2 Retiring the PC NFS server. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 110
7.2.3 Configuring VDISK swap . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 110
7.2.4 Configuring yum . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 110
7.2.5 Adding additional RPMs . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 111
7.2.6 Turning off unneeded services . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 112
7.2.7 Configuring the VNC server . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 113
7.2.8 Setting the system to halt on SIGNAL SHUTDOWN . . . . . . . . . . . . . . . . . . . 115
7.2.9 Turning on the NFS server . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 115
7.2.10 Configuring SSH keys. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 116
7.2.11 Inserting the vmcp module . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 116
7.2.12 Rebooting the system . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 117
7.2.13 Verifying the changes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 117

**Chapter 8. Installing and configuring RHEL 5.2**. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 119
8.1 Installing the golden image . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 120
8.1.1 Creating the user ID RHEL52 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 120
8.1.2 Adding RHEL52 to the AUTOLOG1 PROFILE EXEC. . . . . . . . . . . . . . . . . . . . 121
8.1.3 Preparing RHEL52 bootstrap files. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 122
8.1.4 Installing RHEL 5.2 to the golden image. . . . . . . . . . . . . . . . . . . . . . . . . . . . 123
8.1.5 Verifying the installation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 124
8.2 Configuring the golden image . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 125
8.2.1 Configuring swap spaces . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 125
8.2.2 Configuring automount of install tree . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 125
8.2.3 Configuring yum for online updates . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 126
8.2.4 Adding additional RPMs . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 127
8.2.5 Turning off unneeded services . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 127
8.2.6 Configuring the VNC server . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 127
8.2.7 Setting system to halt on SIGNAL SHUTDOWN . . . . . . . . . . . . . . . . . . . . . . 127
8.2.8 Configuring SSH keys. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 127
8.2.9 Rebooting the system . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 128
8.2.10 Verifying the changes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 128

**Chapter 9. Configuring RHEL 5.2 for cloning** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 129

9.1 Formatting DASD for minidisks 130
9.1.1 Defining a new user ID for a virtual server 131
9.1.2 Adding LINUX01 to AUTOLOG1's PROFILE EXEC 132
9.2 Cloning a virtual server manually 133
9.3 Cloning one new virtual server 137
9.3.1 Creating a configuration file for LINUX01 137
9.3.2 Using the clone script 139
9.4 Cloning three more virtual servers 141
9.4.1 Defining three more user IDs 142
9.4.2 Creating three new configuration files 143
9.4.3 Granting user IDs access to VSWITCH 143
9.4.4 Testing logging on to a new user ID 144
9.5 Reviewing system status 145

**Chapter 10. Installing Linux with kickstart** 147
10.1 Configure the controller for kickstart 148
10.2 Configure the LINUX02 user for kickstart 150
10.3 Kickstart the LINUX02 user 150

**Chapter 11. Servicing Linux with Red Hat Network** 153
11.1 Registering your system with RHN 154
11.2 Installing and updating packages using yum 154
11.3 Managing your Linux guest through RHN 155
11.4 Updating a system without Internet access 156
11.4.1 Configuring the server 156
11.4.2 Configuring the client 156
11.4.3 Updating packages on the server 157

**Chapter 12. Cloning open source virtual servers** 159
12.1 Creating a virtual Web server 160
12.1.1 Installing Apache RPMs 160
12.1.2 Testing Apache 161
12.1.3 Turning on a firewall 161
12.1.4 Configuring SSL for Apache 163
12.1.5 Populating your Web site 163
12.1.6 Apache resources 164
12.2 Creating a virtual LDAP server 164
12.2.1 Cloning a Linux virtual server 164
12.2.2 Installing the OpenLDAP server 164
12.2.3 Configuring the OpenLDAP server 165
12.2.4 Migrating existing users to LDAP 166
12.2.5 Configuring an LDAP client 169
12.2.6 Adding new user accounts 170
12.3 Creating a virtual file and print server 171
12.3.1 Cloning a Linux virtual server 172
12.3.2 Installing the necessary RPMs 172
12.3.3 Configuring the Samba configuration file 173
12.3.4 Adding a Samba user 173
12.3.5 Starting Samba at boot time 174
12.3.6 Testing your changes 174
12.3.7 Configuring printing 175
12.4 Creating a virtual application development server 176
12.4.1 Additional resources 176

**Chapter 13. Miscellaneous recipes** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 179
13.1 Adding a logical volume . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 180
13.1.1 Adding minidisks to the z/VM directory entry . . . . . . . . . . . . . . . . . . . . . . . . . . . 180
13.1.2 Making the new minidisks available . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 180
13.1.3 Formatting and partitioning the minidisks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 181
13.1.4 Create the logical volume and file system. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 182
13.1.5 Updating the file system table. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 184
13.2 Extending an existing logical volume . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 185
13.3 Centralizing home directories for LDAP users . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 187
13.3.1 Configuring the NFS server. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 187
13.3.2 Configuring the golden image for LDAP and automount. . . . . . . . . . . . . . . . . . 188
13.4 Rescuing a Linux system . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 190
13.4.1 Entering single user mode . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 190
13.4.2 Entering a rescue environment . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 190

**Chapter 14. Monitoring z/VM and Linux** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 193
14.1 Using INDICATE and other commands . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 194
14.1.1 Using the INDICATE command . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 194
14.1.2 Using other basic commands . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 196
14.2 z/VM Performance Toolkit. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 198
14.2.1 Configuring the z/VM Performance Toolkit . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 198
14.2.2 Configuring Web Browser support . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 199
14.2.3 Configuring PERFSVM . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 200
14.2.4 Starting the z/VM Performance Toolkit . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 202
14.2.5 Using the z/VM Performance Toolkit. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 202
14.3 Monitoring Linux . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 205
14.3.1 Monitoring Linux with rmfpms . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 206
14.3.2 Monitoring Linux performance data from the kernel . . . . . . . . . . . . . . . . . . . . . 207
14.4 Viewing Linux data in the Performance Toolkit. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 208
14.5 Common Performance Toolkit issue . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 209

**Appendix A. References** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 211
z/VM configuration files . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 212
Quick reference sheets . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 212
XEDIT quick reference sheet . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 212
vi quick reference sheet . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 213

**Appendix B. z/VM source code** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 215
Obtaining and using the Web material. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 215
z/VM REXX EXECs and XEDIT macros . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 216
The CPFORMAT EXEC . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 216
The CHPW540 XEDIT macro . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 220
The LABEL540 EXEC. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 223
The LABEL540 XEDIT macro . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 225
PROFILE EXEC for Linux user IDs. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 226

**Appendix C. Linux source code**. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 227
Obtaining and using the Web material. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 227
The clone script . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 227

**Related publications** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 239
IBM Redbooks publications . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 239
Other publications . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 239
Online resources . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 240

How to get Redbooks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 240
Help from IBM . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 240

**Index** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 241

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described 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:
*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

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

AIX®
DirMaint™
ECKD™
eServer™
IBM®
OMEGAMON®
RACF®
Redbooks®
Redbooks (logo) ®
REXX™
RMF™
S/390®
System p®
System x™
System z10™
System z9®
System z®
Tivoli®
Workplace™
z/OS®
z/VM®
z10™
z9®
zSeries®

The following terms are trademarks of other companies:

NOW, and the NetApp logo are trademarks or registered trademarks of NetApp, Inc. in the U.S. and other countries.

Novell, SUSE, the Novell logo, and the N logo are registered trademarks of Novell, Inc. in the United States and other countries.

Java, MySQL, Solaris, Sun, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

# Preface

This IBM® Redbooks® publication describes how to “roll your own” Linux® virtual servers on IBM System z® hardware under z/VM®. This edition applies to Version 5, Release 4, Modification 0 of z/VM (product number 5741-A05) and Version 5, Release 2 of Red Hat Enterprise Linux.

With a z/VM and Linux infrastructure, you can reduce the time between deciding on the acquisition of new servers and then implementing them because new servers can be deployed in a matter of minutes. This powerful “build and clone” capability can enable you to launch new products and services without the exhaustive planning, purchasing, installing and configuring of new hardware and software that can be associated with conventional discrete hardware servers. Development groups that need test environments built and rebuilt rapidly to enable them to efficiently deliver their projects, while handling change management in the process, can also benefit from this unique advantage.

The publication adopts a cookbook format that provides you with a concise, repeatable set of procedures for installing and configuring z/VM in a logical partition (LPAR), and then installing and customizing Linux. To implement these instructions, you need an IBM System z logical partition with associated resources, z/VM 5.4 media, and a Linux distribution.

This book assumes that you have a general familiarity with System z technology and terminology. It does not assume an in-depth understanding of z/VM and Linux. It is written for those who want to get a quick start with z/VM and Linux on the mainframe.

## The team that wrote this book

This book was originally written in 2005 by Michael MacIsaac, Jin Xiong and Curtis Gearhart. It was updated in 2006 by Michael MacIsaac, Carlos Ordonez and Jin Xiong. It was updated a third time in late 2006 and early 2007 by Marian Gasparovic taking the lead on the SLES 10 version and Brad Hinson of Red Hat working on the RHEL 5 version.

In late 2007 and early 2008, Michael MacIsaac, Brad Hinson, and Lester Peckover updated it for z/VM 5.4, SLES 10 SP2, and RHEL 5.2.

**Michael MacIsaac** supports Linux and z/VM on IBM System z® in Poughkeepsie, NY.

**Bradford Hinson** is the Sr. Support Engineering lead for System z at Red Hat in Raleigh, North Carolina. He specializes in Linux support for the mainframe, and leads technical oversight of all System z projects and activities. He holds a BS degree in Computer Science from the University of North Carolina at Chapel Hill, and MS® degrees from UNC Greensboro and UNC Chapel Hill.

**Lester Peckover** is an Infrastructure, Enterprise Performance, and Capacity Architect in the United Kingdom. He has 32 years of experience in IT covering IBM System x™ and System p, and has also worked on a number of other system platforms in performance and other areas, specializing in virtualization. His areas of expertise include System z and Linux running on the mainframe, as well as performance analysis and modelling. He helped to develop the IBM VM Performance Toolkit and other IBM VM program products and tools from other vendors. Lester has written, taught, and presented worldwide on performance and capacity and other system management topics.

Production of this IBM Redbooks publication was managed by:

**Lydia Parziale** is a Project Leader for the ITSO team in Poughkeepsie, New York with domestic and international experience in technology management including software development, project leadership, and strategic planning. Her areas of expertise include e-business development and database management technologies. Lydia is a Certified IT Specialist with an MBA in Technology Management and has been employed by IBM for more than 23 years in various technology areas.

Sincere thanks to the following people who contributed to this project in many different ways:

Jerry Epker
**Fidelity Information Services**

Bill Bitner, Bruce Hayden, Denny Refsnider, Roger Lunsford, Jim Switzer and Romney White
**IBM Endicott**

Carsten Otte, Claudia Prawirakusumah
**IBM Germany**

Ray Mansell
**IBM Hawthorne**

Kim Betros, Duane Beyer, Michel Considine, Roy Costa, Greg Geiselhart, Dionne Graff, Susan Greenlee, Ernest Horn, Rong Lee, Bill Norton, Lydia Parziale, Thang Pham, Bruce Smilowitz, Kyle Smith, Paul Sutera, Helen Tsang, Donna Von Dehsen, Hossee Wakil, Jeng-Fang Wang and Dennis Wunder
**IBM Poughkeepsie**

Bernard Wu
**NCCI**

Mark Post
**Novell®**

Matthew Eernisse
**OpenSource Applications**

David Boyes, Adam Thornton
**Sine Nomine Associates**

Chester Hood
**State of Tennessee**

Thanks the zBMC team in IBM Poughkeepsie and to the many who answered questions on the linux-390 list server. Special thanks to Tony Giaccone and Roy Costa from IBM Poughkeepsie for always finding another free LPAR and more DASD.

## History

There are a number of *Virtualization Cookbooks*, so a brief history follows.

### 2008

In October 2008, two cookbooks were published, one targeting the Novell/SUSE® SLES distribution, the other targeting the Red Hat RHEL distribution:

- *z/VM and Linux on IBM System z The Virtualization Cookbook for SLES 10 SP2*, SG24-7493
- This book: *z/VM and Linux on IBM System z The Virtualization Cookbook for Red Hat Enterprise Linux 5.2*, SG24-7492

Associated with the Redbooks are REXX™ EXECs and Linux scripts to help you install and configure z/VM and Linux. These tools are not IBM products and are not formally supported. However, they are informally supported and are available on the Web.

### 2007

In March 2007, two cookbooks were published on linuxvm.org/present, each book targeting a different distribution:

- *z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES 10*
- *z/VM and Linux on IBM System z: The Virtualization Cookbook for RHEL 5*

### 2006

In September 2006, a cookbook was published that addressed both 31-bit and 64-bit RHEL 4:

- *IBM z/VM and Linux on IBM System z: Virtualization Cookbook for Red Hat Enterprise Linux 4*, SG24-7272

In August 2006, a cookbook was published on linuxvm.org/present that addressed both 31-bit and 64-bit SLES 9:

- *z/VM and Linux on IBM System z: The Virtualization Cookbook 2*

## Chapters and appendixes

The remaining chapters and appendixes in this book are summarized in the following list:

- Chapter 1, "Introduction to z/VM and Linux" on page 1, explains the concept of virtualization and z/VM infrastructure design, and describes the usability tests performed during the development of this book.
- Chapter 2, "Planning" on page 7, describes how to plan hardware, software, and networking resources. It discusses DASD labeling conventions used in the book and password planning. Sample worksheets are provided for the examples used in the book, along with blank copies for your use.
- Chapter 3, "Configuring a desktop machine" on page 19, describes how to set up Windows® desktops. Specifically, the following tools are discussed:
  - How to obtain and set up PuTTY: a commonly used SSH client
  - How to obtain and set up a VNC client: a tool for running graphical applications
  - 3270 emulator applications
- Chapter 4, "Installing and configuring z/VM" on page 27, shows how to install and configure z/VM.
- Chapter 5, "Servicing z/VM" on page 71, describes how to apply service to z/VM both in the form of Programming Temporary Fixes (PTFs) and Recommended Service Upgrades (RSUs).

- Chapter 6, “Configuring a Network File System server for RHEL 5.2” on page 85, explains how to set up a temporary NFS server on a Linux PC for the purpose of installing the first two Linux images. After the System z controller Linux is installed, you can copy the Linux install tree to it and retire the Linux PC server.
- Chapter 7, “Installing RHEL 5.2 on the controller” on page 91, describes how to install and configure two Linux images onto the first Linux user ID: the *golden image*, which it is cloned from, and the *controller*, which does the cloning among other tasks.
- Chapter 8, “Installing and configuring RHEL 5.2” on page 119, describes how to install and configure two Linux images onto the first Linux user ID: the *golden image*, which it is cloned from, and the *controller*, which does the cloning among other tasks.
- Chapter 9, “Configuring RHEL 5.2 for cloning” on page 129 explains how to prepare z/VM user IDs and clone your first virtual server.
- Chapter 12, “Cloning open source virtual servers” on page 159, shows how to configure cloned Linux images into the following virtual servers:
  - Web server virtual server
  - LDAP virtual server
  - File and print virtual server
  - Basic application development system
- Chapter 13, “Miscellaneous recipes” on page 179 describes how to add a logical volume to a Linux system and how to set up a z/VM Discontiguous Saved Segment (DCSS) in conjunction with the Linux eXecute In Place 2 (xip2) file system.
- Chapter 14, “Monitoring z/VM and Linux” on page 193, describes basic steps to begin monitoring z/VM and your new Linux virtual servers.
- Appendix A, “References” on page 211, provides references Web sites, books and other pertinent information.
- Appendix B, “z/VM source code” on page 215 lists the z/VM source code associated with this book: z/VM REXX EXECs and XEDIT macros.
- Appendix C, “Linux source code” on page 227 lists the one Linux script associated with this book.

## Summary of changes in October 2008 version

There are significant changes in this book:

- The z/VM sections are updated for V5.4.
- The Red Hat sections are updated for Red Hat Enterprise Linux 5.2.
- The Linux system that is cloned is called *golden image* instead of *master image*.
- The controller and the golden image are installed onto two separate user IDs, not on the same user ID as in previous books.
- The file system layout recommends two 3390-3s (100 and 101) instead of just one (100) and implements logical volumes for more adaptable file systems.
- Sections were added on Cooperative Memory Management (CMM1).

The release of this book also removes some sections:

- The DAZL application has been removed.
- The “Backup and Restore” chapter has been removed.

## Conventions

The following font conventions are used in this book:

**`Monospace and bold`** Commands entered by the user on the command line

**`<value>`** Values inside angle brackets are examples and are to be replaced with values correct for your enterprise.

`monospace` File, directories, user ID and minidisk names

The following command conventions are used in this book:

- z/VM commands are prefixed with **==>**
- z/VM XEDIT subcommands are prefixed with **====>**
- Linux commands running as root are prefixed with `#`
- Linux commands running as non-root are usually prefixed with `$`

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  **ibm.com**/redbooks

- Send your comments in an e-mail to:

  redbooks@us.ibm.com

- Send your comments directly to:

  mikemac@us.ibm.com

Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

# 1

# Introduction to z/VM and Linux

Virtualization is a hot topic in the IT industry. The IBM mainframe, z/VM and its predecessors have been performing virtualization for four decades. Today, it is the most functionally rich virtualization platform available. When Linux came to the IBM mainframe in 2000, it was a natural fit to run under z/VM. You can run many tens of Linux images on the same System z logical partition (LPAR). Some customers are running hundreds in production mode.

With a z/VM and Linux infrastructure, you can reduce the time between deciding on the acquisition of new servers and then implementing them because new servers can be deployed in a matter of minutes. This powerful build and clone capability can enable you to launch new products and services without the exhaustive planning, purchasing, installing and configuring of new hardware and software that can be associated with conventional discrete hardware servers. Development groups who need test environments built and rebuilt rapidly to enable them to efficiently deliver their projects, while handling change management in the process, can also benefit from this unique advantage.

Listed here are some of the most significant strengths offered by the IBM mainframe and z/VM:

- Their virtualization capabilities are more mature and robust than any other hardware and hypervisor combination.
- z/VM provides a rich, functional, and sophisticated level of system management which can greatly benefit running large numbers of Linux servers.
- z/VM’s virtual switch (VSWITCH) makes networking Linux much simpler.
- Full volume backup of systems allows for complete disaster recovery when another data center is available.
- z/VM is one of the easiest operating systems to customize at the base installation level, with only a relatively small number of configuration files. After it is set up, z/VM will run for months with little maintenance and administration required.

Much function has been added to z/VM since Version 5.2, as described in the following section.

### z/VM 5.3

z/VM 5.3 became generally available in June 2007. Scalability was extended to allow 256 GB of real memory, a total of 8 TB of virtual storage, and 32 real processors. z/VM V5.3 also added support for the Collaborative Memory Management Assist (CMMA) on the z9® EC and the z9 BC processors or later. Virtual Machine Resource Manager (VMRM) detects when memory is constrained and notifies the Linux guests, which can then adjust their memory consumption to help relieve the memory constraint.

In the previous major release (z/VM 5.2), many memory contention issues were removed with the Control Program (CP) now using memory above 2 GB for a much broader set of operations. Previously, guest pages had to be moved below 2 GB for many reasons, for example in both standard I/O and Queued Direct I/O (QDIO). Now I/O can be performed using buffers anywhere in real memory, and QDIO structures can reside above 2 GB, as can most CP control blocks. These improvements offer constraint relief for large-real-memory virtual server environments that are memory-intensive.

### z/VM 5.4

z/VM 5.4, available in August 2008, provides major improvements when operating on System z servers with large memory configurations. It improves scalability and can help support increased workloads on IBM System z servers. This release exploits new capabilities of the System z10™ including:

- Greater flexibility, with support for the new z/VM-mode logical partitions, allowing all System z processor-types (CPs, IFLs, zIIPs, zAAPs, and ICFs) to be defined in the same z/VM LPAR for use by various guest operating systems
- The capability to install Linux on System z from the HMC, which eliminates network setup or a connection between an LPAR and the HMC
- Enhanced physical connectivity by exploiting all OSA-Express3 ports, thus helping to service the network and reduce the number of required resources

z/VM 5.4 dynamic memory upgrade support allows real memory to be added to a running z/VM system, thereby avoiding the need to shut down z/VM and its guests, deactivate the LPAR, change its memory allocation, reactivate the LPAR, re-IPL z/VM, and restart its guests. Memory can be added non-disruptively to individual guests that support the dynamic memory reconfiguration architecture.

Read more about System z virtualization capabilities on the Web at:

http://www.vm.ibm.com

## 1.1 What is virtualization

Virtualization is the ability for a computer system to share resources so that one physical server can act as many *virtual* servers. z/VM allows the sharing of the mainframe's physical resources such as disk (DASD), memory (sometimes called storage), network adapters (OSA cards) and CPU (CPs or IFLs). These resources are managed by a *hypervisor*. The z/VM hypervisor is called Control Program (CP).

When a user logs onto z/VM, the hypervisor creates a virtual machine which can run one of many different operating systems. The two operating systems discussed in this book are the z/VM native operating system, known as the Conversational Monitoring System (CMS), and Linux. CMS can be thought of as a z/VM shell. Virtual machines running Linux as guests of a z/VM host become the virtual servers.

## 1.2 This book's approach

Today there are numerous technical publications that discuss virtualization, but few of them demonstrate how to achieve it. This book gives you the "hands on" instructions needed to achieve a virtualized environment on your enterprise.

The approach adopted is to keep all solutions simple, as expressed in the well-known quote from Albert Einstein "Everything should be made as simple as possible, but not simpler". This book attempts to use a clear and insightful presentation to explain the somewhat complex topics covered here.

## 1.3 Choices and decisions made in this book

When deciding on installing, maintaining and provisioning (cloning) Linux virtual servers under z/VM, there are many basic choices to make. Here are some of the choices and assumptions made in this book:

- Use of "roll your own" cloning versus cloning products

  A discussion of cloning products such as Aduva Onstage, the Mainstar Provisioning Expert, IBM Tivoli® Provisioning Manager and IBM Director is beyond the scope of this book. Although these are all viable solutions, the cloning described in this book allows you to roll your own Linux images without requiring such products. However, these products are more sophisticated than the simple clone script and z/VM configuration described in this book.

- Use of the `USER DIRECT` file versus a directory maintenance product

  The use of the `USER DIRECT` file was chosen over a directory maintenance product such as IBM DirMaint™ or CA's VM:Direct.

  If using DirMaint as a directory maintenance product would be better for your enterprise, you can refer to *Getting Started With Linux*, SC24-6096, to configure z/VM. To configure Linux, however, use this book.

- Use of predefined user IDs versus provisioning

  z/VM user IDs must be predefined to clone. There is no attempt to *provision* them (define and bring Linux user IDs online automatically) as part of the cloning process. The target Linux user ID must exist with the appropriate minidisks defined.

- Use of read-write versus shared read-only Linux `/usr/` file system

  Some cloning solutions use an environment that shares the `/usr/` file system. This choice often makes the solution more complex, especially when adding software to the virtual servers. For this book, a read-write `/usr/` file system on the virtual servers was chosen to keep things as simple as possible.

- Use of conventional 3390 ECKD™ DASD versus FBA disks accessed using SCSI over FCP

  The System z server has traditionally only supported 3390 DASD. Support has been extended to include SCSI/FBA disks in storage area networks (SANs). The support of FBA disks is slightly more complicated than conventional DASD. In keeping things as simple as possible, only conventional DASD is described in this book.

- Use of manual installation versus cloning script or EXEC

  Two methods of cloning are described: manually, and using a Linux bash script. The manual method was chosen so that you will better learn the described concepts. However, the Linux script is also provided so you can save time.

## 1.4 Infrastructure design

To install and configure z/VM, and install, configure and clone Linux, or to provision virtual servers, there must be a certain infrastructure design in place. A System z server with associated resources and the z/VM operating system define much of this infrastructure.

Figure 1-1 shows a block diagram of a System z10 with multiple LPARs. z/VM 5.4 is installed in one of these LPARs. z/VM comes with many user IDs predefined. The most important six IDs are shown in the z/VM LPAR above the dashed line. Below the dashed line, you see the user IDs described in this book.

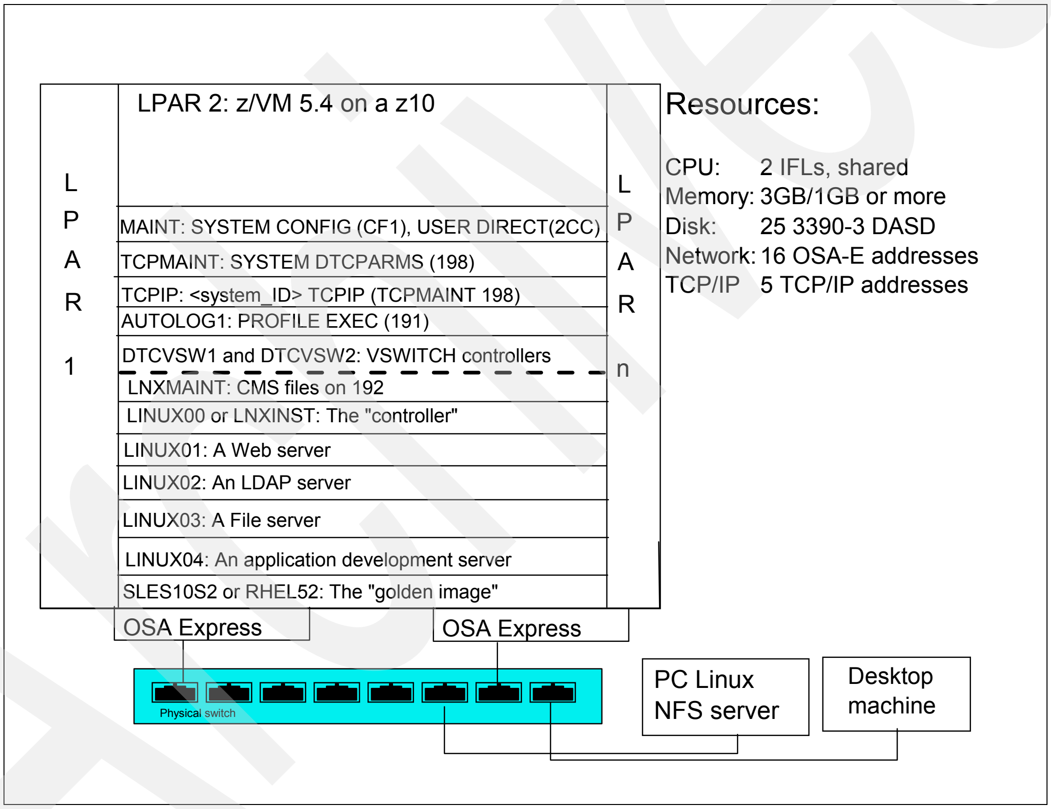

LPAR 1
LPAR 2: z/VM 5.4 on a z10
MAINT: SYSTEM CONFIG (CF1), USER DIRECT(2CC)
TCPMAINT: SYSTEM DTCPARMS (198)
TCPIP: <system_ID> TCPIP (TCPMAINT 198)
AUTOLOG1: PROFILE EXEC (191)
DTCVSW1 and DTCVSW2: VSWITCH controllers
LNXMAINT: CMS files on 192
LINUX00 or LNXINST: The "controller"
LINUX01: A Web server
LINUX02: An LDAP server
LINUX03: A File server
LINUX04: An application development server
SLES10S2 or RHEL52: The "golden image"
LPAR n
Resources:
CPU: 2 IFLs, shared
Memory: 3GB/1GB or more
Disk: 25 3390-3 DASD
Network: 16 OSA-E addresses
TCP/IP 5 TCP/IP addresses
OSA Express
OSA Express
Physical switch
PC Linux NFS server
Desktop machine

*Figure 1-1 System infrastructure and z/VM user IDs*

The user IDs that are described in this book have the following functions:

**LNXMAINT** This is a user ID on which to store files that will be used by both CMS and Linux.

**LINUX00** or **LNXINST** This is the controller that does the cloning. It also serves as the Linux install server, and has other functions.

**LINUX01-04** This is the user IDs that will be cloned to. Each virtual server is configured with a two 3390-3 minidisks to allow for slightly more than 4 GB of space.

**SLES10S2** or **RHEL52** This is the SLES10 SP2 or RHEL 5.2 golden image. This is the Linux system that is cloned.

## 1.5 Usability tests performed for this book

During the writing of this book, many usability tests were conducted. The participants had a variety of skills, but none had both Linux and z/VM system administration skills.

By the end of the first day in all of the formal tests, most participants had all completed up to and including Chapter 5, “Servicing z/VM” on page 71, so z/VM was installed, serviced and customized for TCP/IP communications with a highly available VSWITCH.

By the end of the second day, most participants had cloned their first Linux virtual server. You should be able to complete most steps in the book in four solid days of work, if all goes well.

2

# Planning

This chapter covers the planning needed before you install z/VM. It begins by discussing a *bill of materials*, or all the resources that you need. Then it describes conventions adopted for labeling 3390 volumes. Finally, it presents resource worksheets for:

- z/VM resources other than direct access storage device (DASD)
- DASD resources
- Linux resources
- Linux user IDs

# 2.1 Bill of materials

The resources needed for a Linux on System z project can be divided into:

- Hardware
- Software
- Networking

## 2.1.1 Hardware resources

The following hardware is needed:

- A System z or zSeries logical partition (LPAR); z800, z900, z890 or z990, System z9® or System z10
  - Processors or CPUs: One IFL (or CP) minimum; using two or more is strongly recommended
  - Memory: 3 GB central/1 GB expanded minimum; using 6 GB/2 GB or more is recommended. This 3:1 ratio of central to expanded storage is a useful starting point for relatively small systems such as in these example sizes.

    See the following Web site for a discussion about how to apportion memory:

    http://www.vm.ibm.com/perf/tips/storconf.html
  - DASD: 25 3390-3s or 9 3390-9s at a minimum
  - Open Systems Adapter (OSA) network cards: One card minimum with 8 device numbers (technically 6, but OSA "triplets" usually start on an even address). Two OSA Express cards with eight device numbers on one and four on the other is recommended for high availability.
- A network-attached computer that will act as an Network File System (NFS) server temporarily with at least 6 GB of disk space, but more may be needed. Setting up a Linux PC or UNIX® server is described.

  If you only have access to a Windows machine, AllegroNFS has been suggested as an NFS server; for information refer to

  http://nfsforwindows.com/home
- A workstation or desktop with network access to the mainframe

## 2.1.2 Software resources

The following software resources are needed:

- z/VM 5.4 install media with documentation (installation from DVD is described).
- SLES 10 SP2 (for the book SG24-7493) or RHEL 5.2 (for the book SG24-7492) Linux install media
- An operating system for the NFS server
- The code associated with this book
- Tools on the workstation and desktop:
  - A 3270 Emulator such as Attachmate Extra, Hummingbird Host Explorer, or IBM Personal Communications for Windows desktops
  - A Linux SSH client such as PuTTY (recommended) or TeraTerm
  - A VNC viewer

These resources are described in more detail in the chapters that follow.

### 2.1.3 Networking resources

The following network resources are needed:

- A TCP/IP address for z/VM
- One TCP/IP address for each Linux virtual server
- Associated TCP/IP information:
  - DNS host name
  - DNS domain
  - DNS server TCP/IP address
  - TCP/IP gateway
  - TCP/IP subnet mask
  - TCP/IP broadcast address (usually calculated from address and subnet mask)
  - TCP/IP MTU size

The TCP/IP addresses must be routed to the OSA cards.

## 2.2 z/VM conventions

It is good practice to use naming conventions so that you and others can recognize z/VM resources by their names. This section discusses such conventions for DASD volume names and backup file names.

### 2.2.1 Volume labeling convention

Use a convention for labeling DASD. Your enterprise may already have a labeling convention which will largely determine the labels to be given to the DASD used by your z/VM and Linux LPAR.

Each System z DASD is addressed with a device number consisting of four hexadecimal digits. Each System z DASD has a six-character label. It is convenient to include the four-digit address in the label so that you can easily tell the address of each DASD from its label. When followed, this convention guarantees that no two DASDs will have the same label. This can be an important issue, especially when z/OS® has access to the DASD.

Sometimes DASD is shared among LPARs, in which case your z/VM LPAR can see DASD that is owned by other LPARs. In this situation, it is convenient to identify the LPAR that *owns* the DASD. The volume labeling convention used in this book identifies the LPAR using the first character. That leaves the second character in the label to identify the basic function of the DASD.

The LPAR used in this book is identified by the character M. The following characters are used for the types of DASD in the second character of the label:

| | |
|---|---|
| **M** | Minidisk space (PERM) |
| **P** | Paging space (PAGE) |
| **S** | Spool space (SPOL) |
| **T** | Temporary disk space (TDISK) |
| **V** | z/VM operating system volumes |

For example, Figure 2-1 shows the labeling convention for the DASD in LPAR *M*, of type *minidisk* at real address *A700*.

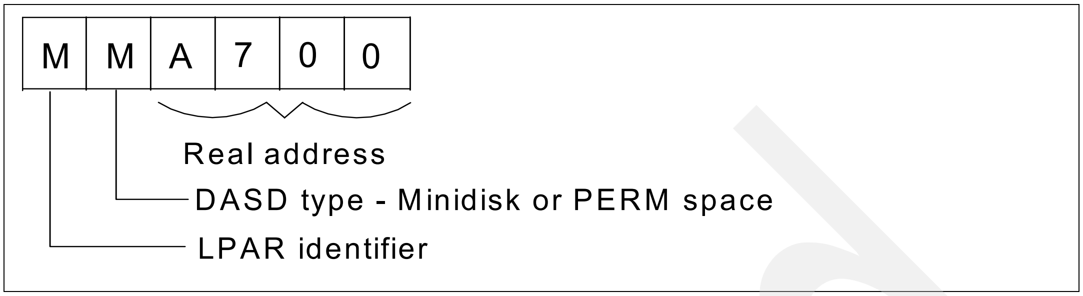


*Figure 2-1 DASD labeling convention*

The letter M is hard-coded into REXX EXECs that adopt this convention. If you want a different LPAR identifier character, they can easily be changed.

### 2.2.2 Backup file naming convention

It is recommended that you keep copies of important z/VM and Linux configuration files. You should always keep copies of original configuration files in case you need to go back to them.

Because z/VM file names are limited to 16 characters (eight for the file name and eight for the file type), only the last four characters of the file type are used. This often requires some characters to be overwritten. For the original file, the suffix `ORIG` is used, and for the most recent working copy, the suffix `WRKS` (for “it WoRKS”) is used. For example, the original `USER DIRECT` file is copied to the file `USER DIREORIG` before it is modified the first time.

### 2.2.3 The command retrieve convention

The ability to retrieve past commands is a common tool. Often it is useful to retrieve in both directions in case you “pass” the command that you are looking for. The default Linux shell, **`bash`**, does this by default with the up arrow and down arrow keys.

There is a convention in z/VM to use the F12 function key (labeled `PF12` on physical 3270 devices) to retrieve the last command, although it is not defined to all user IDs. There is no convention retrieve commands in the other direction, but it is possible to set another key to that function. Therefore, F11 is used to *retrieve forward* because it is next to F12. Also, the same function is useful in the editor, **`XEDIT`**. The **`?`** subcommand retrieves past commands, so it is recommended that you assign it to F12.

## 2.3 Disk planning

There are different aspects to consider when planning how to choose and allocate disk storage, including:

- Conventional ECKD DASD versus FBA disks over SCSI/FCP
- 3390-3s versus 3390-9s or large disk support
- Amount of disk storage per Linux image and how to allocate file systems

### DASD versus SCSI/FCP

This book describes how to use conventional ECKD DASD and does not discuss FBA disks accessed over SCSI/FCP. This is not because either technology is superior, but simply because DASD seems to be much more common than SCSI/FCP disks. If you were to use SCSI/FCP disks, cloning using the **`clone.sh`** script would have to be modified to account for World Wide Port Names and Numbers.

Sometimes a combination of these two types of disk storage is used; in that case the ECKD emulated DASD is often used for the root file system and SCSI/FCP disks are used for large data storage areas.

### 3390-3s versus 3390-9s

Emulated 3390-3s format to about 2.3 GB. In contrast, 3390-9s are three times the size, or about 6.8 GB. Either size will work, although 3390-3s have been recommended over 3390-9s by some performance analysts.

This book describes mainly using 3390-3s; however, comments are added where using 3390-9s differs, especially with installing z/VM.

### Disk storage per Linux image

Disk storage has the following characteristics

- This version of the book recommends using two 3390-3 DASD to create minidisks at virtual addresses 100 and 101. Previous versions only recommended a single minidisk at virtual address 100.
- The root file system is on `/dev/dasda1` with a recommended size of 384 MB. It is not a logical volume so that if there are any problems with LVM, the system will still be able to boot.
- Other file systems are on logical volumes that are part of a single volume group with the characteristics listed in Table 2-1.

*Table 2-1 Recommended logical volume file systems and sizes*

| Mount point | Logical volume name | Size |
|---|---|---|
| `/usr/` | `usr-lv` | 2 GB |
| `/var/` | `var-lv` | 512 MB |
| `/opt/` | `opt-lv` | 384 MB |
| `/tmp/` | `tmp-lv` | 384 MB |

This layout uses about 3.5 GB out of 4.5 GB of disk space. You could choose to use disk sizes other than 3338 cylinders (3390-3 minus cylinder 0). For example, if you chose to use 3390-9s, you could give 100 and 101 each half of the volume, giving each Linux about 6.8 GB of disk space.

**Important:** However you choose to lay out the minidisks, it is important that the golden image and all target Linux user IDs have two minidisks of the same size at virtual addresses 100 and 101. These assumptions are coded into the **`clone.sh`** script.

## 2.4 Memory planning

Planning memory may be the most difficult issue with z/VM and Linux on System z, but it is the most important to ensure adequate performance. The simplest solution may appear to involve having enough central memory (storage) in the LPAR so that z/VM never pages and Linux never swaps. However, realistically such resource is often not available.

A useful rule of thumb is to allocate memory on a “just enough” basis for each Linux server. A starting point is to set a virtual machine size by changing the memory allocation value at just over the value at which the guest starts to swap at the Linux system level when under normal loading. If some level of sustained swapping is inevitable due to the nature of the workloads, then ensure virtual disks are used for the swap media.

An understanding of memory planning is recommended, and resources that cover this important topic are listed here.

- IBM Redbooks publication *Linux on IBM System z: Performance Measurement and Tuning*, SG24-6926:

  `http://www.redbooks.ibm.com/redpieces/abstracts/sg246926.html?Open`
- IBM z/VM Performance Resource pages:

  `http://www.vm.ibm.com/perf/`
- The IBM z/VM page specifically discussing memory allocation:

  `http://www.vm.ibm.com/perf/tips/storconf.html`

One rule that can be recommended is to only have as few virtual machines logged on (or disconnected) as possible to handle the workload being presented. Every virtual machine that is not required should be logged off where appropriate, because this will mean more memory for the other virtual servers that remain running.

## 2.5 Password planning

Secure passwords are critical to reliable security. However, requiring many different passwords generally leads to people writing them down, which clearly detracts from security. Sometimes it is difficult to balance these two extremes, especially when dealing with system administration roles.

This book considers different system administration roles:

- The z/VM system administrator
- The Linux system administrator
- The Linux virtual server users

The z/VM and Linux system administrator may be the same person.

The method of backing up z/VM data onto the Linux controller means that the Linux administrator will have access to all z/VM passwords. Therefore, the examples in this book set all z/VM and Linux system administration passwords to the same value: lnx4vm. If the z/VM and Linux system administrator roles must be kept separate and the Linux administrator is not to have access to the z/VM passwords, then a different method of backing up z/VM data must be chosen.

You may want to define a finer granularity for passwords based on the following system administration roles:

- The main z/VM system administrator (`MAINT`)
- The z/VM network administrator (`TCPMAINT`)
- The z/VM Linux administrator (`LNXMAINT`, Linux controller, Linux virtual server user IDs)
- The Linux virtual server users (with or without access to 3270 sessions, with or without the root passwords)

The sets of passwords that you define will depend on the roles that your organization will adopt.

## 2.6 Planning worksheets

Four worksheets are included in this section. They are populated with the resources used in writing this book. There are also four corresponding blank worksheets in 2.7, “Blank worksheets” on page 16, for your use.

### 2.6.1 z/VM resources used in this book

Table 2-2 lists the z/VM resource values used in the examples in this book. You can use these values as a reference for completing the blank worksheets that follow.

*Table 2-2 z/VM resources worksheet*

| Name | Value | Comment |
|---|---|---|
| LPAR name | P21 | 3 GB main storage/1 GB expanded, 4 shared IFLs |
| CPC name | PELCP01 | Name of CPC on which the LPAR is located |
| z/VM system name | VMLINUXA | Name to be assigned to z/VM system |
| TCP/IP host name | virtc522 | Assigned by a network administrator; helpful to set in DNS beforehand, but not necessary |
| TCP/IP domain name | itso.ibm.com | Helpful to set in DNS beforehand |
| TCP/IP gateway | 9.12.4.1 | The router to and from the local subnet |
| DNS server 1 | 9.12.6.7 | Assigned by the network administrator |
| DNS server 2/3 (optional) | | Not used |
| OSA device name | eth0 | Name of the interface to be assigned by IPWIZARD |
| OSA starting device number | 3020 | Start of OSA *triplet* for the z/VM TCP/IP stack |
| TCP/IP address | 9.12.5.22 | The TCP/IP address of the z/VM system |
| Subnet mask | 255.255.255.0 | Assigned by network administrator |
| OSA device type | QDIO | Often “QDIO” for OSA/Express cards |
| Network type | Ethernet | Usually “Ethernet” |
| Port name (optional) | | Not required by z/VM |
| Router type | None | Usually “None” |

| Name | Value | Comment |
|---|---|---|
| MTU size | 1500 | Check with network administrator |
| Primary OSA device number for VSWITCH | 3024 | Specify the first device number (must be even number) and the next two device numbers will also be used |
| Secondary OSA device number for VSWITCH | 3028 | Should be on a different CHPID/OSA card |

## 2.6.2 z/VM DASD used in this book

Table 2-3 lists the z/VM DASD resource values used in the examples in this book.

*Table 2-3 z/VM DASD used in this book*

| Device number | Label | Type | Notes |
|---|---|---|---|
| D850 | 540RES | CP owned | z/VM system residence volume |
| D851 | 540SPL | CP owned | z/VM spool volume 1 |
| D852 | 540PAG | CP owned | z/VM paging volume 1 |
| D853 | 540W01 | CP owned | z/VM first work volume |
| D854 | 540W02 | CP owned | z/VM second work volume |
| D855 | MPD855 | CP owned | Paging volume 2 |
| D856 | MMD856 | System minidisk | `LINUX00` or `LNXINST 100` (the controller) |
| D857 | MMD857 | System minidisk | `LNXMAINT 191`, `192` (common CMS files), `LINUX00` or `LNXINST 101` (used for `/nfs/` logical volume) |
| D950 | MPD950 | CP owned | Paging volume 3 |
| D951 | MPD951 | CP owned | Paging volume 4 |
| D952 | MMD952 | System minidisk | `LINUX00` or `LNXINST 102` |
| D953 | MMD953 | System minidisk | `LINUX00` or `LNXINST 103` |
| D954 | MMD954 | System minidisk | `LINUX00` or `LNXINST 104` |
| D955 | MMD955 | System minidisk | `SLES10S2` or `RHEL52 100` (the golden image) |
| D956 | MMD956 | System minidisk | `SLES10S2` or `RHEL52 101` |
| D957 | MPD957 | CP owned | `LINUX01 100` |
| DA50 | MMDA50 | System minidisk | Paging volume 5 |
| DA51 | MMDA51 | System minidisk | `LINUX01 101` |
| DA52 | MMDA52 | System minidisk | `LINUX02 100` |
| DA53 | MMDA53 | System minidisk | `LINUX02 101` |
| DA54 | MMDA54 | System minidisk | `LINUX03 100` |
| DA55 | MMDA55 | System minidisk | `LINUX03 101` |

| Device number | Label | Type | Notes |
|---|---|---|---|
| DA56 | MMDA56 | System minidisk | `LINUX04 100` |
| DA57 | MMDA57 | System minidisk | `LINUX04 101` |
| DB50 | MPDB50 | CP owned | Paging volume 6 |

## 2.6.3 Linux resources used in this book

Table 2-4 lists the Linux PC NFS server resources used for the first System z Linux install:

*Table 2-4 Linux NFS server resources used in this book*

| Name | Value | Comment |
|---|---|---|
| TCP/IP address | 9.12.4.185 | |
| User/password | root/lnx4vm | |
| NFS-exported install directory | `/nfs/sles10sp2/` or `/nfs/rhel52/` | Directory with DVD 1 |

Table 2-5 lists the Linux resources used in the examples in this book.

*Table 2-5 Linux resources used in this book*

| Name | Value | Comment |
|---|---|---|
| Linux install password | lnx4vm | |
| TCP/IP gateway | 9.12.4.1 | Obtain from network administrator |
| Subnet mask | 255.255.252.0 | Obtain from network administrator |
| DNS server | 9.12.6.7 | Obtain from network administrator |
| VNC installation password | lnx4vm | |

## 2.6.4 Linux user IDs used in this book

Table 2-6 lists the z/VM user IDs for Linux used in the examples in this book.

*Table 2-6 Linux user ID used in this book*

| Linux user ID | IP address | DNS name | Notes |
|---|---|---|---|
| `SLES10S2` or `RHEL52` | 9.12.5.29 | virtc529.itso.ibm.com | SLES 10 SP2 or RHEL 5.2 golden image |
| `LINUX00` or `LNXINST` | 9.12.5.30 | virtc530.itso.ibm.com | The controller |
| `LINUX01` | 9.12.5.31 | virtc531.itso.ibm.com | A Web virtual server |
| `LINUX02` | 9.12.5.32 | virtc532.itso.ibm.com | An LDAP virtual server |
| `LINUX03` | 9.12.5.33 | virtc533.itso.ibm.com | A file and print virtual server |
| `LINUX04` | 9.12.5.34 | virtc534.itso.ibm.com | An application development virtual server |

# 2.7 Blank worksheets

Blank copies of the same four worksheets are provided for your use.

## 2.7.1 z/VM resources worksheet

Use the worksheet in Table 2-7 to document the z/VM resources that you will use.

*Table 2-7 z/VM resources blank worksheet*

| Name | Value | Comment |
|---|---|---|
| LPAR name | | |
| CPC name | | |
| System name | | |
| TCP/IP host name | | |
| TCP/IP domain name | | |
| TCP/IP gateway | | |
| DNS server 1 | | |
| DNS server 2/3 (optional) | | |
| OSA device name | | Often “eth0” |
| OSA starting device number | | |
| TCP/IP address | | |
| Subnet mask | | |
| OSA device type | | Often “QDIO” |
| Network Type | | Often “Ethernet |
| Port name (optional) | | |
| Router Type | | Often “None” |
| Primary OSA device number for VSWITCH | | |
| Secondary OSA device number for VSWITCH | | Should be on a different CHPID/OSA card than primary |

## 2.7.2 z/VM DASD worksheet

Use the worksheet in Table 2-8 to document the z/VM DASD that you will use.

*Table 2-8 z/VM DASD blank worksheet*

| Device number | Label | Type | Notes |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

## 2.7.3 Linux resources worksheet

Use the worksheet in Table 2-10 to document the resources associated with the NFS server that will be used to be the install source of the first System z Linux.

*Table 2-9 Linux NFS server resources blank worksheet*

| Name | Value | Comment |
|---|---|---|
| TCP/IP address | | |
| User/password | | |
| NFS-exported install directory | | |

Use the worksheet in Table 2-11 to document your System z Linux resources.

*Table 2-10 Linux resources blank worksheet*

| Name | Value | Comment |
|---|---|---|
| Linux install password | | |
| Linux root password | | |
| Apache user ID and password | | |
| Linux TCP/IP gateway | | |
| Linux TCP/IP broadcast | | |
| Linux DNS server | | |
| VNC Installation password | | |

## 2.7.4 Linux user ID worksheet

Use the worksheet in Table 2-11 to document the Linux user IDs that you will create.

*Table 2-11 Linux user ID blank worksheet*

| Linux user ID | IP address | DNS name | Notes |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

# 3

# Configuring a desktop machine

Many people use Microsoft® Windows as a desktop operating system. This chapter addresses the use of the following tools, which are recommended for accessing z/VM and Linux from a Windows desktop:

- An SSH client: PuTTY is recommended
- A VNC client: RealVNC is recommended
- A 3270 emulator: Many choices are available

## 3.1 PuTTY: a free SSH client for Windows

Throughout this book, SSH is used to log into Linux systems. It is easy to use and cryptographically secure. If you are using a Linux desktop system, an SSH client is built in. But if you are using a Windows desktop, you will need a useful SSH client.

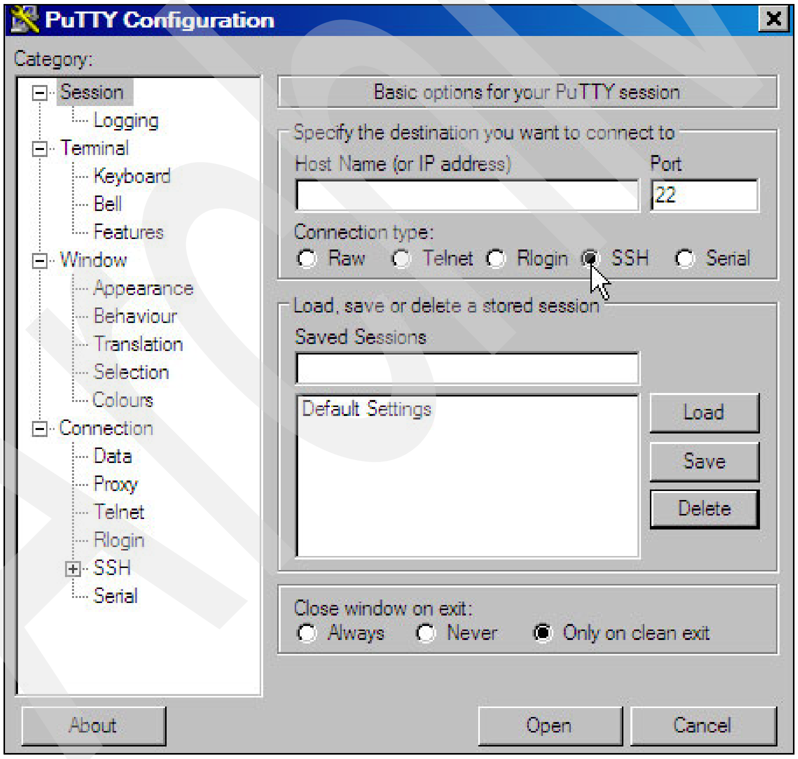
PuTTY is probably the most commonly used SSH client. You can find a PuTTY client for Windows on CD1 of a SLES 10 distribution in the `/dosutils/putty` directory. You can download PuTTY from the Web at:

`http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html`

To download from this page, click the **putty.exe** link for your architecture. Save the file in a directory path such as `C:\WINNT`. PuTTY is a standalone executable (no installation is needed, other than copying the file). You may also want to create a shortcut on your desktop or task bar.

Open PuTTY and the configuration window shown in Figure 3-4 on page 22 should open. If you spend a few minutes at this point configuring PuTTY, it may pay off in a time savings later. The examples shown use PuTTY Release 0.60.

1. In the PuTTY Configuration window, in the left Category window, click **Session**.
2. Under the Connection Type heading on the top right, click the **SSH** radio button as shown in Figure 3-1. This specifies the use of the SSH protocol.



PuTTY Configuration

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address):

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

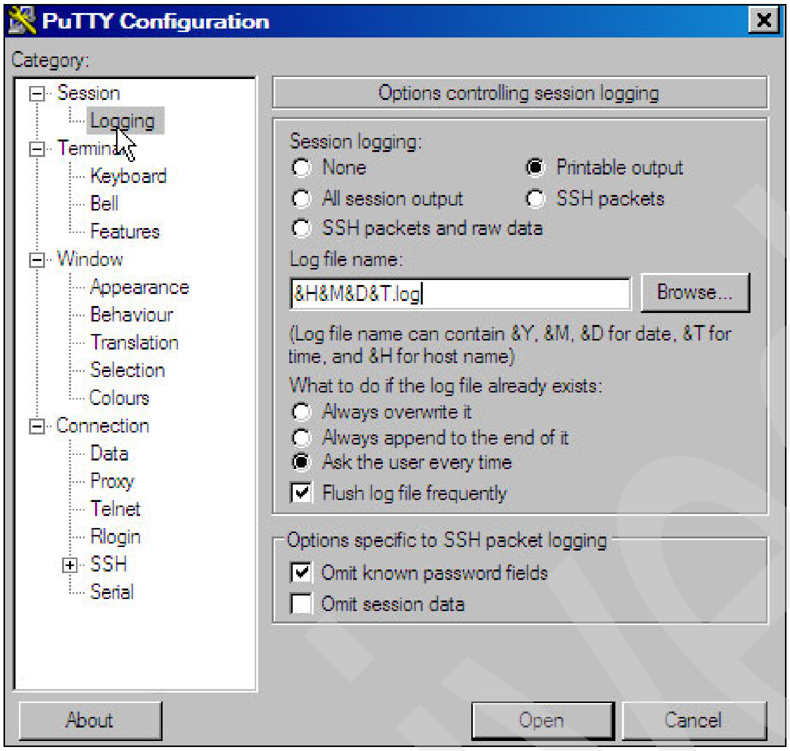
Saved Sessions:

Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

*Figure 3-1 PuTTY Configuration window*

3. Click **Logging** in the left window as shown in Figure 3-2 on page 21.
   – Click the radio button **Printable output** in the Session logging radio group. This will allow you to go back and check the output of certain commands.
   – Set the Log file name to `&H&M&D&T.log` so a time stamp will be in the file name.



Options controlling session logging

Session logging:

○ None ● Printable output

○ All session output ○ SSH packets

○ SSH packets and raw data

Log file name:

&H&M&D&T.log

(Log file name can contain &Y, &M, &D for date, &T for time, and &H for host name)

What to do if the log file already exists:

○ Always overwrite it

○ Always append to the end of it

● Ask the user every time

☑ Flush log file frequently

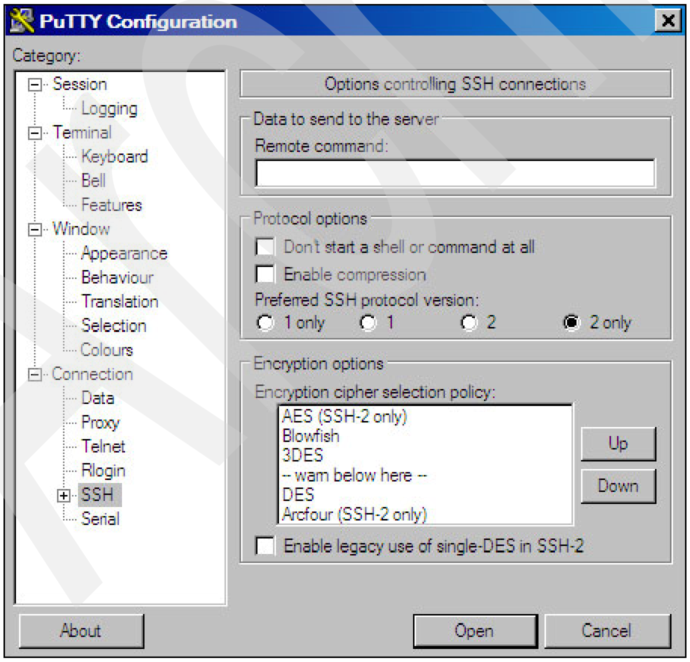
Options specific to SSH packet logging

☑ Omit known password fields

□ Omit session data

*Figure 3-2 Setting logging*

4. In the left window, click **SSH** near the bottom, as shown in Figure 3-3.
5. On the right side, under Preferred SSH protocol version, click the **2 only** radio button.



Options controlling SSH connections

Data to send to the server

Remote command:

Protocol options

□ Don't start a shell or command at all

□ Enable compression

Preferred SSH protocol version:

○ 1 only ○ 1 ○ 2 ● 2 only

Encryption options

Encryption cipher selection policy:

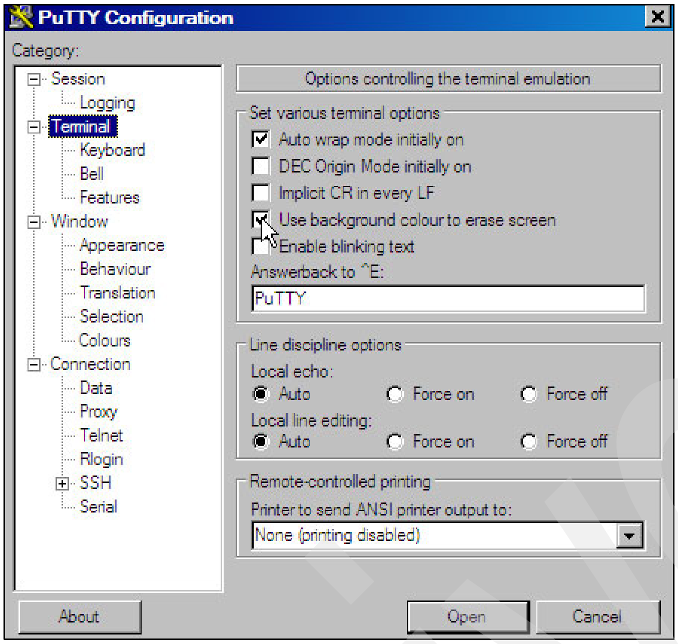
AES (SSH-2 only)
Blowfish
3DES
-- warn below here --
DES
Arcfour (SSH-2 only)

□ Enable legacy use of single-DES in SSH-2

*Figure 3-3 Setting SSH Protocol 2*

6. In the left Category window, click **Terminal** as shown in Figure 3-4 on page 22.

7. Select the **Use background colour to erase screen** check box, which results in a better job of painting the window for applications that use curses (block graphics).



Options controlling the terminal emulation

Set various terminal options

☑ Auto wrap mode initially on
□ DEC Origin Mode initially on
□ Implicit CR in every LF
☑ Use background colour to erase screen
□ Enable blinking text

Answerback to ^E: PuTTY

Line discipline options

Local echo: ● Auto ○ Force on ○ Force off

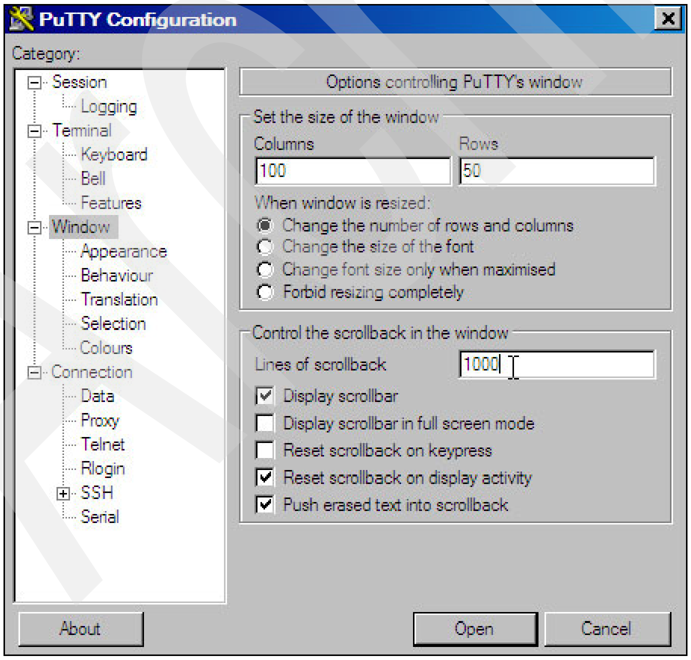
Local line editing: ● Auto ○ Force on ○ Force off

Remote-controlled printing

Printer to send ANSI printer output to: None (printing disabled)

*Figure 3-4 Customizing PuTTY SSH settings (Part 1 of 4)*

8. Click **Window** in the left pane, as shown in Figure 3-5.
9. You may choose a larger window size and more lines of scrollback. In this example, 50 rows, 100 columns are and 1000 lines of scrollback are set.



Options controlling PuTTY's window

Set the size of the window

Columns: 100 Rows: 50

When window is resized:

● Change the number of rows and columns
○ Change the size of the font
○ Change font size only when maximised
○ Forbid resizing completely

Control the scrollback in the window

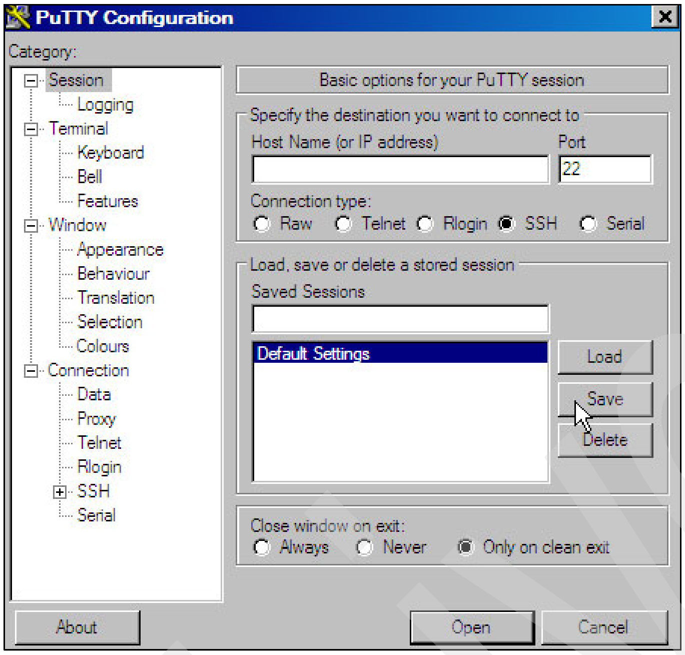
Lines of scrollback: 1000

☑ Display scrollbar
□ Display scrollbar in full screen mode
□ Reset scrollback on keypress
☑ Reset scrollback on display activity
☑ Push erased text into scrollback

*Figure 3-5 Setting Window and scrollback size*

10. Click **Session** in the left pane, as shown in Figure 3-6.

11. Click **Default Settings** in the Saved Sessions pane, then click the **Save** button. This makes all future sessions that you define inherit the preferences you just set.



Host Name (or IP address):
Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Saved Sessions:

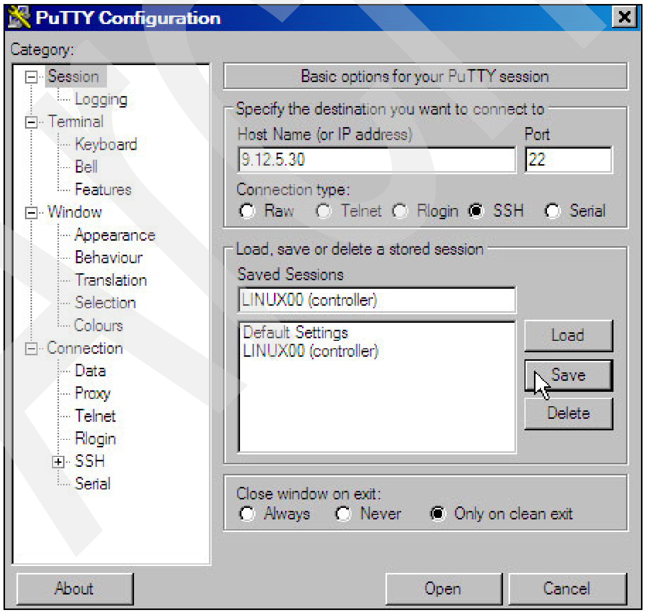
Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

Figure 3-6 Saving new default settings

## Saving sessions

To save sessions, perform the following steps. In this example a session for LINUX00, or the controller, is saved.



Host Name (or IP address): 9.12.5.30
Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Saved Sessions: LINUX00 (controller)

Default Settings
LINUX00 (controller)

Close window on exit: ○ Always ○ Never ● Only on clean exit

Figure 3-7 Customizing PuTTY window settings (part 4 of 4)

To save a session for each virtual server, perform the following:

1. In the Host Name (or IP address) field, enter the TCP/IP address (or DNS name).

2. Under the Saved Sessions text area, choose a name that you will remember. In this example, the name `LINUX00 (controller)` is used.
3. Again click **Save** and you should see the name added to the Saved Session list.

   Now whenever you start PuTTY, you can simply double-click any saved session name, and an SSH session to the desired Linux system will be invoked.

# 3.2 Setting up a VNC client

A VNC client allows access to a graphical windowing environment with System z Linux.

If you are using a Linux desktop you probably have, or at least have access to, a VNC client named **`vncviewer`**. It is part of the tightvnc package.

## 3.2.1 Downloading and running RealVNC

If you have a Windows desktop, the VNC client from RealVNC is a popular choice (you can also find a TightVNC client for Windows on CD1 of a SLES 10 distribution in the `/dosutils/tightvnc` directory).

You can purchase a full function RealVNC client, or there is a free version. The RealVNC home page is:

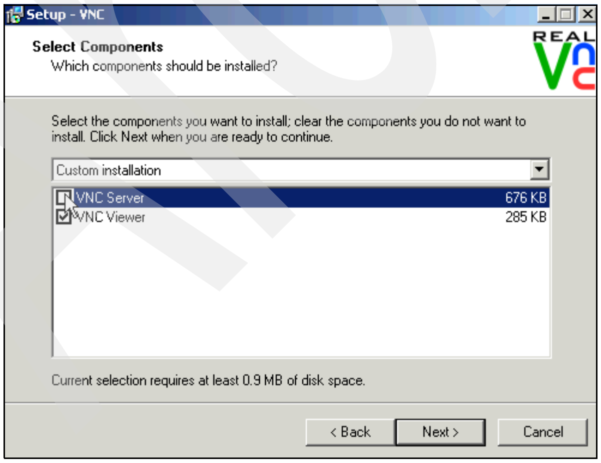
`http://www.realvnc.com`

The download page is:

`http://www.realvnc.com/download.html`

Fill out the Web form and download the executable. When you have downloaded it, run it and an install program will start. At the time of writing, RealVNC 4.1.2 was the current version.

Accept all defaults, however, you probably do not need a VNC server on your desktop. So you can **deselect VNC Server** from the *Select Components* window as shown in Figure 3-8.



Select Components
Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

Custom installation

☐ VNC Server 676 KB
☑ VNC Viewer 285 KB

Current selection requires at least 0.9 MB of disk space.

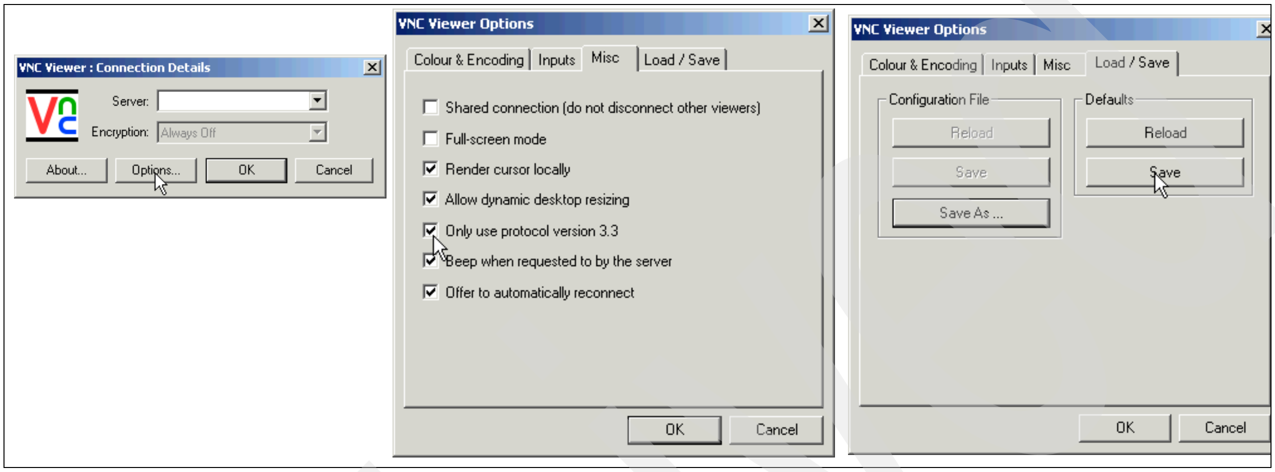
*Figure 3-8 RealVNC Select Components window*

Complete the windows and the installation process should go quickly.

### 3.2.2 Customizing RealVNC

The latest VNC protocol is Version 4, which is the default with the VNC client. This version will work with the VNC servers shipped with SLES 10 or RHEL5.

If, however, you need to use protocol Version 3.3 for SLES 9, open the VNC client and click the **Options** button, as shown in the left side of Figure 3-9. Click the **Misc** tab. Click the check box **Use only protocol version 3.3** as shown in the center of the figure. Finally, click the **Load/Save** tab and click **Save** to save the changes.



☐ Shared connection (do not disconnect other viewers)
☐ Full-screen mode
☑ Render cursor locally
☑ Allow dynamic desktop resizing
☑ Only use protocol version 3.3
☑ Beep when requested to by the server
☑ Offer to automatically reconnect

*Figure 3-9 Setting VNC client to use protocol 3.3 - required for SLES 9 only*

Your VNC client should now be ready to connect to the VNC server that your Linux systems will have.

## 3.3 3270 emulators

To access a logon session with z/VM, it is common to use a 3270 emulator that runs on Windows. Many commercial products are available. Some of the more common ones are:

- Attachmate Extra!
- Hummingbird Host Explorer
- IBM Personal Communications
- Quick3270

It is beyond the scope of this book to explain the details of configuring all the various emulators. However, it is recommended that you investigate the following settings for your emulator:

- Set the Enter and Clear function keys to be where you would expect them.

  On some emulators, the default Enter key action is set to the right Ctrl key of modern keyboards. Likewise, the Clear key action is sometimes set to the Esc key in the upper left corner of modern keyboards, or to the Pause key in the upper right.

- Set a larger window.

  Often the default number of lines in an emulator session is 24. You will probably be more productive with a 32, 43, or more lines if they can easily fit in a window, given your desktop display size and resolution.

- Have the session automatically reconnect after logoff.

  Having a new logon window come back immediately after you log off can also save you time in the long run. This is often not the default behavior.

- Save your connection sessions.

  Rather than continually typing in the IP address or DNS name of the z/VM system to which you want to connect, spend a few minutes defining and saving a session for each system to which you may connect, as described for PuTTY. Then you can usually double-click the saved connection to quickly access a new 3270 session.

# 4

# Installing and configuring z/VM

To complete this chapter, you must complete the majority of Chapter 6, “Configuring a Network File System server for RHEL 5.2” on page 85.

However, we recommend that you start here, because the **`instdvd`** step (used when installing z/VM) listed in 4.1.2, “Copying a vanilla z/VM system to DASD” on page 31, takes two or more hours to complete. While that process is running, you can configure the Network File System (NFS) server. Alternatively, if you have other personnel who can work on the project, you can start both chapters at the same time on the different systems.

In this chapter, you perform the following steps:

- Installing z/VM from DVD
- Configuring TCP/IP
- Configuring the XEDIT profile
- Customizing the SYSTEM CONFIG file
- Configuring TCP/IP to start at IPL time
- Adding paging volumes
- Creating a user ID for common files
- Addressing z/VM security issues
- Relabeling the system volumes

## 4.1 Installing z/VM from DVD

The section that follows assumes a first-level installation of z/VM from DVD onto DASD. If you have not already done so, complete the worksheet that is provided in 2.7.1, “z/VM resources worksheet” on page 16. Note the following points:

- For System z9 hardware and older, you will need access to the Hardware Management Console (HMC) with a user ID that has authority to go into single object operations mode.
- For System z10 hardware and newer, the requirement to be in single object operations mode to access the Load from CD-ROM, DVD or Server function has been removed.

z/VM 5.4 is shipped on tape and DVD. z/VM should install faster from tape due to faster I/O speeds; however, installing from tape may require more trips between and the HMC and the tape drive.

- If you are familiar with the use of the HMC, you can use the two-page *z/VM Summary for Automated Installation and Service (DVD Installation)* to replace or augment the description that follows.
- If you are not familiar with the HMC and z/ VM, you may want to use the complete installation manual *z/VM Guide for Automated Installation and Service Version 5 Release 4.0*, GC24-6099.
- If you are installing z/VM at the *second level* (that is, z/VM under z/VM) or onto SCSI disk, use the GC24-6099 manual because the sections that follow do not address these options.

### 4.1.1 Booting z/VM from DVD

This section explains how to install z/VM 5.4 from an HMC with a DVD-ROM onto 3390-3 equivalent DASD. Some words are included for installing onto the larger 3390-9 DASD. For alternative configurations, such as installing from tape or onto SCSI disks, refer to the z/VM documentation.

1. **Logon** to the Hardware Management Console. You should see the HMC Workplace™ window.
2. Select the LPAR on which you want to install z/VM; this is often reachable by clicking the **CPC images** icon.

> **Note:** Ensure that you have the *correct* LPAR selected. If you are not completely sure, check with someone who is sure.

3. If necessary, click the **racetrack** buttons (two buttons that are circular arrows on the bottom right corner) to traverse to the Recovery or CPC Recovery menu.
4. On the Recovery or CPC Recovery menu, double-click the **Integrated 3270 Console**, as shown at the bottom of Figure 4-1 on page 29.

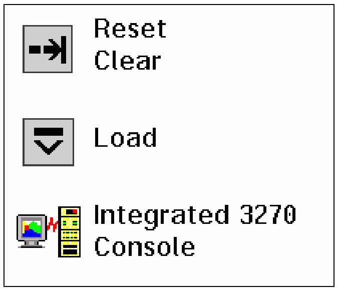


*Figure 4-1 Integrated 3270 Console icon*

A window entitled Integrated 3270 Console for <your CPC> will open. (On earlier HMC levels, the window may be entitled Personal Communications).

> **Hint**: It is convenient to use the Alt-Tab key sequence to move between the HMC window and the 3270 console.

5. Insert the z/VM Product Package Version 5 Release 4.0 DVD into the HMC DVD drive.

> **Important:** On z10 HMCs and later, it is no longer required to be in Single Object Operations mode in order to install z/VM.

6. On a z9 HMC and earlier model, get into Single Object Operations mode (as mentioned, for z10 HMCs and later, this step is not necessary). To get into this mode, perform the following steps:
   a. Double-click the **Groups** icon in the Views Area.
   b. Double-click **Defined CPCs** in the Groups Work Area.
   c. Select your CPC.
   d. If necessary, go around the **racetrack** (the buttons with circular arrows on the bottom right corner) to the CPC Recovery menu.
   e. Double-click the **Single Object Operations** icon. Click **yes** to confirm. A new window Primary Support Element Workplace should appear (on older HMC levels, it will be a “window within a window”). A window about a certificate not being valid may appear; in this case, press **OK**.
   f. Double-click **Groups** near the top of this window.
   g. Double-click **Images** in the Groups Work Area.

> **Important:** If you are unable to get into Single Object Operations mode, it may be because you do not have sufficient permission. Check with the system administrator.

7. The LPAR that z/VM will be installed into should still be selected. On the right you should still see the (CPC) Recovery menu. Double-click the **Load from CD-ROM or Server** icon as shown in Figure 4-2 on page 30.

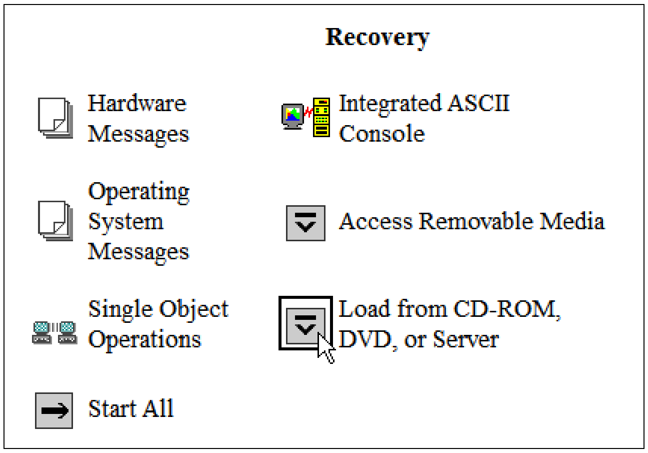
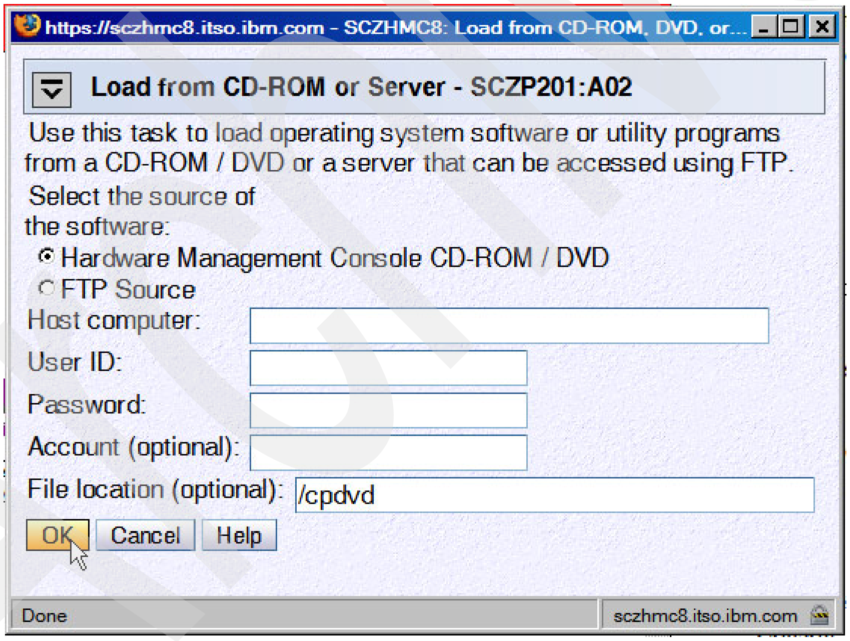


*Figure 4-2 CPC Recovery menu with Load from CD-ROM or Server icon present*

8. On the Load CD-ROM or Server window as shown in Figure 4-3, the radio button **Hardware Management Console CD-ROM / DVD** should be selected.
9. In the same Load CD-ROM or Server window, fill in the File location field with `/cpdvd`. This is the directory on the DVD with the z/VM 5.4 installation code.
10. Click **OK**.



https://sczhmc8.itso.ibm.com - SCZHMC8: Load from CD-ROM, DVD, or...

Load from CD-ROM or Server - SCZP201:A02

Use this task to load operating system software or utility programs from a CD-ROM / DVD or a server that can be accessed using FTP.

Select the source of the software:

● Hardware Management Console CD-ROM / DVD

○ FTP Source

Host computer:

User ID:

Password:

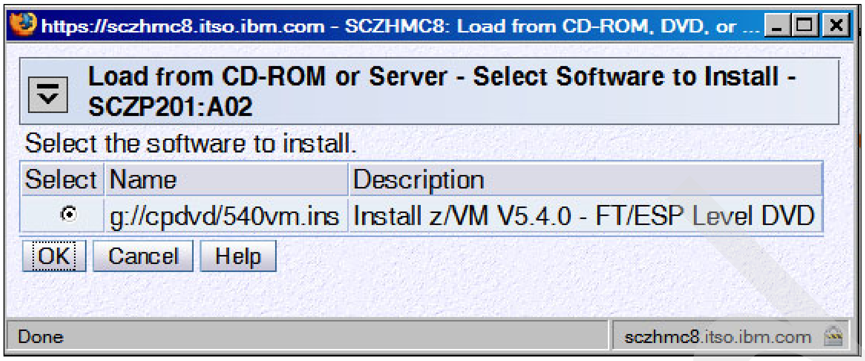
Account (optional):

File location (optional): /cpdvd

*Figure 4-3 Load from CD-ROM or Server window*

11. Load the RAMDISK:

    a. From the Load from CD-ROM or Server window, the software **540vm.ins** should be selected as shown in Figure 4-4 on page 31. Click **OK**.


https://sczhmc8.itso.ibm.com - SCZHMC8: Load from CD-ROM, DVD, or ...
OK
Cancel
Help
Done
sczhmc8.itso.ibm.com

Load from CD-ROM or Server - Select Software to Install - SCZP201:A02

Select the software to install.

| Select | Name | Description |
|---|---|---|
| ● | g://cpdvd/540vm.ins | Install z/VM V5.4.0 - FT/ESP Level DVD |

*Figure 4-4 Selecting z/VM 5.4 RAMdisk system*

b. From the Confirm the action window, click **Yes**. You should see the Load from CD-ROM, DVD or Server Progress window. The green light on the DVD drive should light up.

c. When you see the message `Completed successfully.`, click **OK** to close. This should normally take about two to four minutes.

> **Important:** Normally, the z/VM RAMdisk (IBMVMRAM) loads in about four minutes. However, slow load times have been observed (15 to 18 minutes). When the green light on the DVD drive is solid, the load time will be acceptable. When it is intermittently dark more than it is green, long load times can result. This will also balloon the estimated run time of 2 to 2.5 hours of the **`INSTDVD`** process, which may consume 5 to 6 hours. PMR 58909,180 was opened to address this issue.

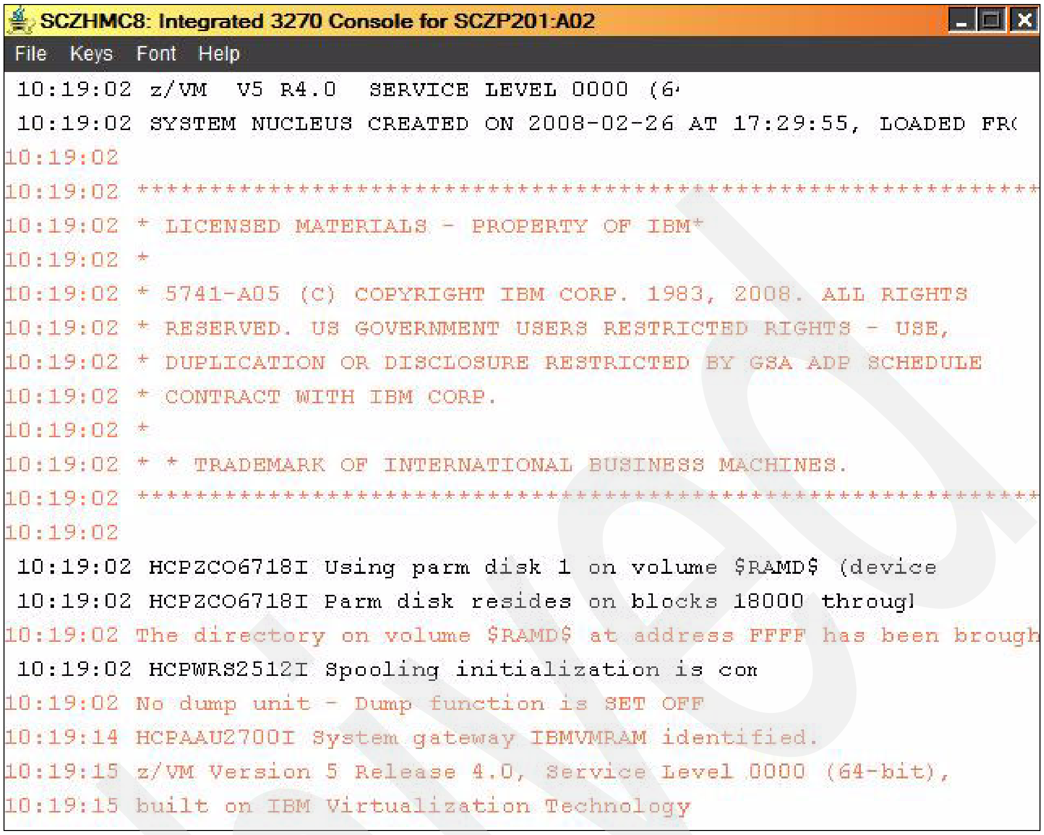
At this point, you should now have an in-memory z/VM 5.4 system running.

### 4.1.2 Copying a vanilla z/VM system to DASD

This section describes the steps to copy z/VM to DASD.

1. You can now get out of Single object operations mode (if you are in it). To do so, log off the primary SE window by closing that window.
2. Move to the Integrated 3270 Console window (you can use the **Alt-Tab** sequence). The RAMdisk should IPL and you should see z/VM boot as shown in Figure 4-5 on page 32.

> **Note:** The Esc key clears the window on the HMC.



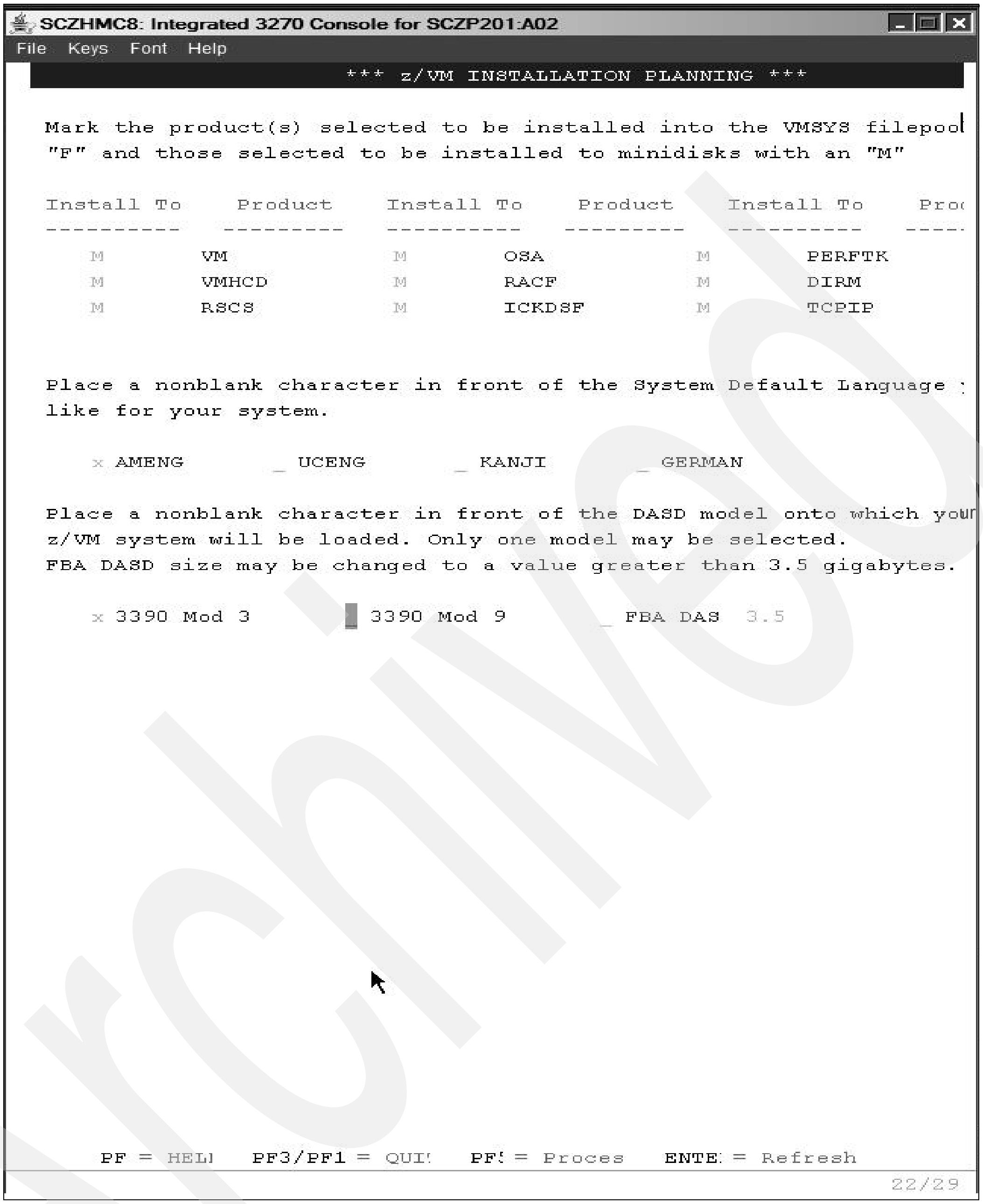
```
 10:19:02 z/VM  V5 R4.0  SERVICE LEVEL 0000 (6
 10:19:02 SYSTEM NUCLEUS CREATED ON 2008-02-26 AT 17:29:55, LOADED FR(
10:19:02
10:19:02 ***************************************************************
10:19:02 * LICENSED MATERIALS - PROPERTY OF IBM*
10:19:02 *
10:19:02 * 5741-A05 (C) COPYRIGHT IBM CORP. 1983, 2008. ALL RIGHTS
10:19:02 * RESERVED. US GOVERNMENT USERS RESTRICTED RIGHTS - USE,
10:19:02 * DUPLICATION OR DISCLOSURE RESTRICTED BY GSA ADP SCHEDULE
10:19:02 * CONTRACT WITH IBM CORP.
10:19:02 *
10:19:02 * * TRADEMARK OF INTERNATIONAL BUSINESS MACHINES.
10:19:02 ***************************************************************
10:19:02
 10:19:02 HCPZCO6718I Using parm disk 1 on volume $RAMD$ (device
 10:19:02 HCPZCO6718I Parm disk resides on blocks 18000 througl
10:19:02 The directory on volume $RAMD$ at address FFFF has been brough
 10:19:02 HCPWRS2512I Spooling initialization is com
10:19:02 No dump unit - Dump function is SET OFF
10:19:14 HCPAAU2700I System gateway IBMVMRAM identified.
10:19:15 z/VM Version 5 Release 4.0, Service Level 0000 (64-bit),
10:19:15 built on IBM Virtualization Technology
```

*Figure 4-5   z/VM first boot on Integrated console*

3. Invoke the **instplan** command. This will allow you to choose associated z/VM products to install, as well as the language to use and the type of DASD on which to install:

```
==> instplan
```

4. You should see the Installation Planning window display shown in Figure 4-6 on page 33. We recommend that you leave the Ms in the top section as is.



```
                    *** z/VM INSTALLATION PLANNING ***

 Mark the product(s) selected to be installed into the VMSYS filepool
 "F" and those selected to be installed to minidisks with an "M"

 Install To    Product    Install To    Product    Install To    Proc
 ----------   ---------   ----------   ---------   ----------   -----
     M        VM              M        OSA             M        PERFTK
     M        VMHCD           M        RACF            M        DIRM
     M        RSCS            M        ICKDSF          M        TCPIP


 Place a nonblank character in front of the System Default Language :
 like for your system.

     x AMENG       _ UCENG       _ KANJI       _ GERMAN

 Place a nonblank character in front of the DASD model onto which your
 z/VM system will be loaded. Only one model may be selected.
 FBA DASD size may be changed to a value greater than 3.5 gigabytes.

     x 3390 Mod 3       _ 3390 Mod 9       _ FBA DAS  3.5









     PF = HELI   PF3/PF1 = QUI!   PF! = Proces   ENTE! = Refresh
                                                             22/29
```

*Figure 4-6 Installation Planning window*

5. On this window, type `X` next to `AMENG` (or select your language) and type `3390 Mod 3` (or the type of DASD you will use), as shown in Figure 4-6.
6. Press **F5**. You should the message `HCPINP8392I INSTPLAN EXEC ENDED SUCCESSFULLY`.
7. Attach the DASD devices onto which z/VM will be installed as defined in your planning worksheet in 2.7.2, “z/VM DASD worksheet” on page 17. In this example, the devices are `d850-d854`.

```
==> att <d850-d854> *
d850-d854 ATTACHED TO MAINT
```

**Important:** The angle brackets <> in the example should not be typed. They are used throughout the book to signify that you should replace the example value with the correct value for your site. For example, if you are installing z/VM onto DASD 1200-1204, you would type the following:

```
==> att 1200-1204 *
```

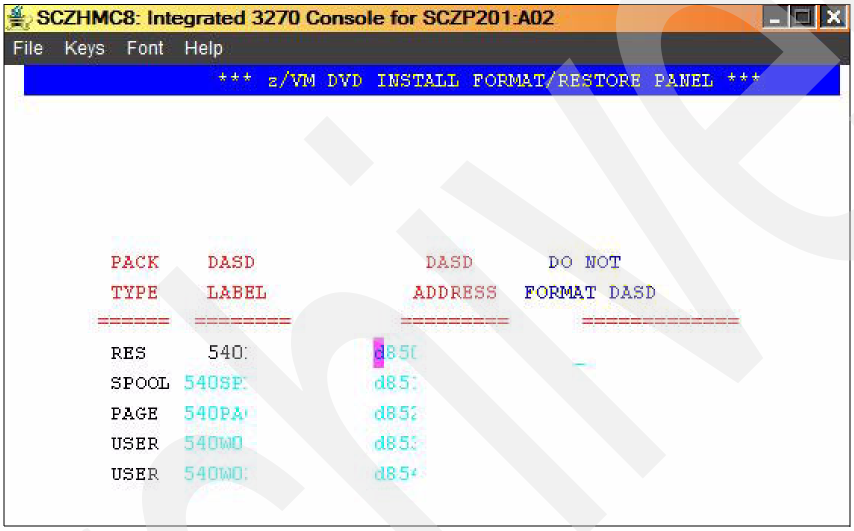
## Running INSTDVD

The INSTDVD EXEC copies the z/VM system from DVD to disk.

1. Execute the **INSTDVD EXEC**:

   ```
   ==> instdvd
   ```

2. If you are using 3390-3s, you see a window asking for the five volumes as shown in Figure 4-7 (if you are using 3390-9s, you will only see three lines).



```
                   *** z/VM DVD INSTALL FORMAT/RESTORE PANEL ***




    PACK     DASD                DASD       DO NOT
    TYPE     LABEL              ADDRESS   FORMAT DASD
   ======  ========            =========      =============
    RES      540                d850
    SPOOL  540SP                d851
    PAGE   540PA                d852
    USER   540W0                d853
    USER   540W0                d854
```

*Figure 4-7 INSTDVD DASD address window*

   a. Enter the addresses of the five volumes (or three volumes, for 3390-9s) that z/VM will be installed on.

   b. Press **F5** to start the installation.

3. Verify that the five DASD addresses to be installed onto are correct. When you see the question `DO YOU WANT TO CONTINUE?`, type **Y.** You should see the message `NOW FORMATTING DASD <D850>`.

**Important:** INSTDVD can take from 45 minutes to two hours or more. You may want to use this time to set up an NFS server, as explained in Chapter 6, “Configuring a Network File System server for RHEL 5.2” on page 85.

Also, read errors have been observed resulting in **INSTDVD** failing. If this is the case, you can try the command **`instdvd (restart`** and the install process should pick up where the read error occurred. This error can be caused by dirt or fingerprints on the DVD.

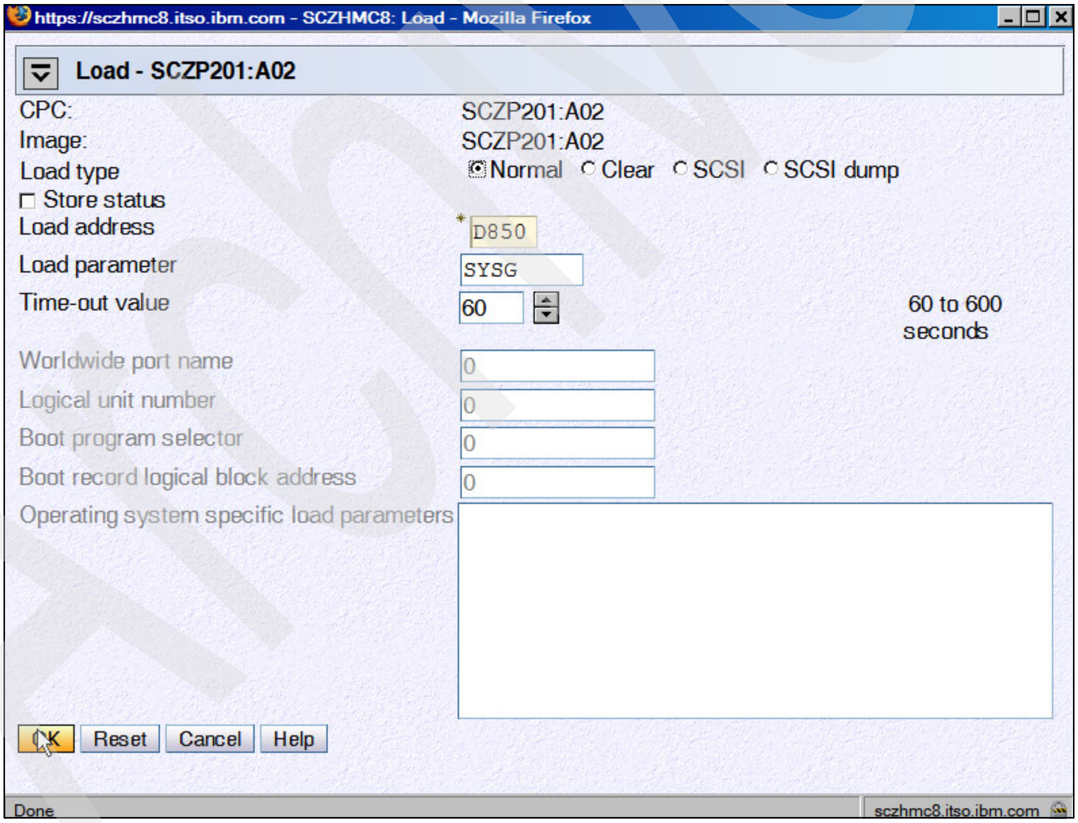
4. You are asked to place the system RSU in the drive. Insert the z/VM Stacked Recommended Service Upgrade 5401 DVD into the HMC DVD-ROM drive.

5. At the Integrated 3270 Console, type `GO`. You should see a messages of the form `DVDLOAD: LOADING FILE CKD5000x IMAGE *.` This step should take two to four minutes.
6. Finally, you should see the message `HCPIDV8329I INSTDVD EXEC ENDED SUCCESSFULLY`.

### 4.1.3 IPL the vanilla z/VM from DASD

IPL your initial z/VM system now on DASD. Your 3270 Integrated Console session should still be running.

1. From the HMC Workplace window, **select your LPAR** by clicking it. You may have to first double-click **Groups**.
2. You should see the CPC Recovery (sometimes entitled simply Recovery) menu. Double-click the **Load** icon in the menu at the right side.
3. The Load window opens as shown in Figure 4-8. Follow these steps:
   a. Set the load address to the new system residence (540RES) volume, which is `D850` in this example.
   b. Set the load parameter to `SYSG`. This specifies the use of the Integrated 3270 console.
   c. Click **OK** to IPL.



https://sczhmc8.itso.ibm.com - SCZHMC8: Load - Mozilla Firefox

Load - SCZP201:A02

CPC: SCZP201:A02
Image: SCZP201:A02
Load type ● Normal ○ Clear ○ SCSI ○ SCSI dump
□ Store status
Load address * D850
Load parameter SYSG
Time-out value 60 60 to 600 seconds
Worldwide port name 0
Logical unit number 0
Boot program selector 0
Boot record logical block address 0
Operating system specific load parameters

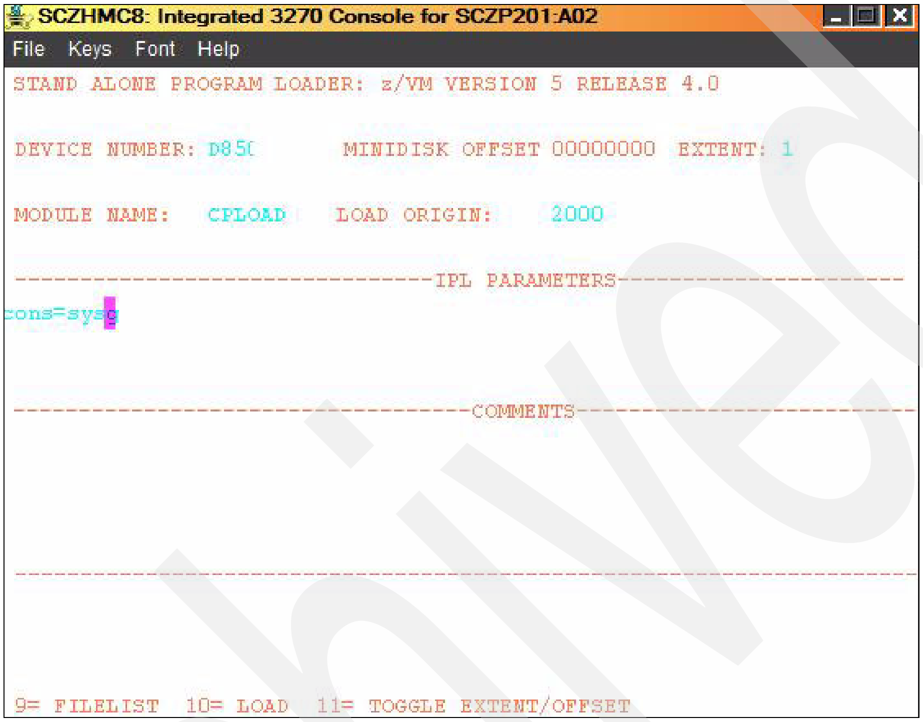
OK
Reset
Cancel
Help
Done
sczhmc8.itso.ibm.com

*Figure 4-8 Load window*

4. When you see the Load Task Confirmation window, click **Yes**.
5. After 1 to 3 minutes, you should see `Success` in the Load Program window. Click **OK**.

6. Move back to the Integrated 3270 console window. You should see the Standalone Program Loader window as shown in Figure 4-9 on page 36.

   a. Press the **Tab** key to traverse to the IPL Parameters section and enter the value **`cons=sysg`**. This specifies the use of the Integrated 3270 console.



```
STAND ALONE PROGRAM LOADER: z/VM VERSION 5 RELEASE 4.0

DEVICE NUMBER: D850       MINIDISK OFFSET 00000000  EXTENT: 1

MODULE NAME:   CPLOAD    LOAD ORIGIN:      2000

--------------------------------IPL PARAMETERS----------------------
cons=sysg


-----------------------------------COMMENTS--------------------------



---------------------------------------------------------------------



9= FILELIST  10= LOAD  11= TOGGLE EXTENT/OFFSET
```

*Figure 4-9 The Standalone Program Loader window*

   b. Press the **F10** key to continue the IPL of your z/VM system. This should take about 1 to 3 minutes.

7. At the `Start (Warm|Force|COLD|CLEAN)` prompt, enter:

   ```
   ==> cold drain noautolog
   ```

8. At the Change TOD clock prompt, enter:

   ```
   ==> no
   ```

9. The last message should be `HCPCRC8082I EREP records are accumulating for userID EREP`. Disconnect from the `OPERATOR` user ID using the **`DISCONNECT`** command:

   ```
   ==> disc
   ```

   Press **Enter** to get a new logon window.

## 4.1.4 Completing the z/VM installation

Follow these steps to complete the z/VM installation

1. On the z/VM login window, logon as MAINT. The password is MAINT. You may receive messages `HCPLNM102E` or `HCPLNM101E` about disks not linked or attached. This is not a problem. Press **Enter** when you see the `VM Read` prompt in the lower right corner.

**Important:** When logging onto a z/VM user ID that runs CMS, you should usually press **Enter** at the `VM READ` prompt. This will result in a prompt of the form:

```
Ready; T=0.01/0.01 11:14:20
```

2. IPL CMS, then press **Enter** at the `VM READ` prompt in the lower right corner. You should see the `Ready;` prompt.

```
==> ipl cms
==> Press Enter at the VM READ prompt
```

3. Run the **`instvm dvd`** command:

```
==> instvm dvd
...
HCPPLD8329I POSTLOAD EXEC ENDED SUCCESSFULLY
...
HCPIVM8392I INSTVM ENDED SUCCESSFULLY
```

This EXEC continues the installation process. This step should take about 4 to 8 minutes. The last message should be `HCPIVM8392I INSTVM ENDED SUCCESSFULLY`.

4. Load the recommended service. First **`IPL CMS`**, then press **Enter** at the VM READ prompt:

```
==> ipl cms
==> Press Enter at the VM READ prompt
Ready;
```

5. For z/VM 5.4, the service name is **5401RSU1**. Verify this file exists on the `MAINT 500` disk:

```
==> acc 500 c
DMSACC724I 500 replaces C (2CC)
==> listfile * * c
5401RSU1 SERVLINK C1
```

6. Run the **`SERVICE ALL`** command to apply the service:

```
==> service all 5401rsu1
```

This step should take about 3 to 6 minutes. The last message should be:

```
VMFSRV2760I SERVICE processing completed successfully.
```

7. Now IPL CMS and run the **`put2prod`** command. This puts the service into production:

```
==> ipl cms
==> Press Enter
Ready;
==> put2prod
```

This step should take about 2 to 4 minutes. The last message should be:

```
VMFP2P2760I PUT2PROD processing completed successfully.
```

A return code of zero (0) is ideal. You may get a return code of 4 and the message:

```
VMFP2P2760I PUT2PROD process completed with warnings.
```

In general on z/VM, a return code of 4 is acceptable. That means that only warnings were issued. A return code of 8 or greater generally means that errors were encountered.

8. Enter the following command to shut down and re-IPL your system:

```
==> shutdown reipl
SYSTEM SHUTDOWN STARTED
```

9. You will lose the current session on the Integrated 3270 Console, but the system should come back in about 2 to 4 minutes.

10. After it comes back, the last message should be `Press enter or clear key to continue`. Press **Enter** and you should see a z/VM logon window.

At this point, you should now have a vanilla z/VM system installed.

# 4.2 Configuring TCP/IP

We recommend that you initially configure TCP/IP using the **IPWIZARD** command, which is generally used just once. After **IPWIZARD** creates the initial configuration files, they are typically maintained manually.

From the z/VM logon window, logon to MAINT. The default password for all z/VM user IDs is the same as the user ID. So enter a password of **maint**, which will not be echoed on the window.

```
USERID   ==> maint
PASSWORD ==>
```

After entering the user ID and password, press **Enter** when the status area in the lower right shows `VM READ`.

## 4.2.1 Use the IPWIZARD tool

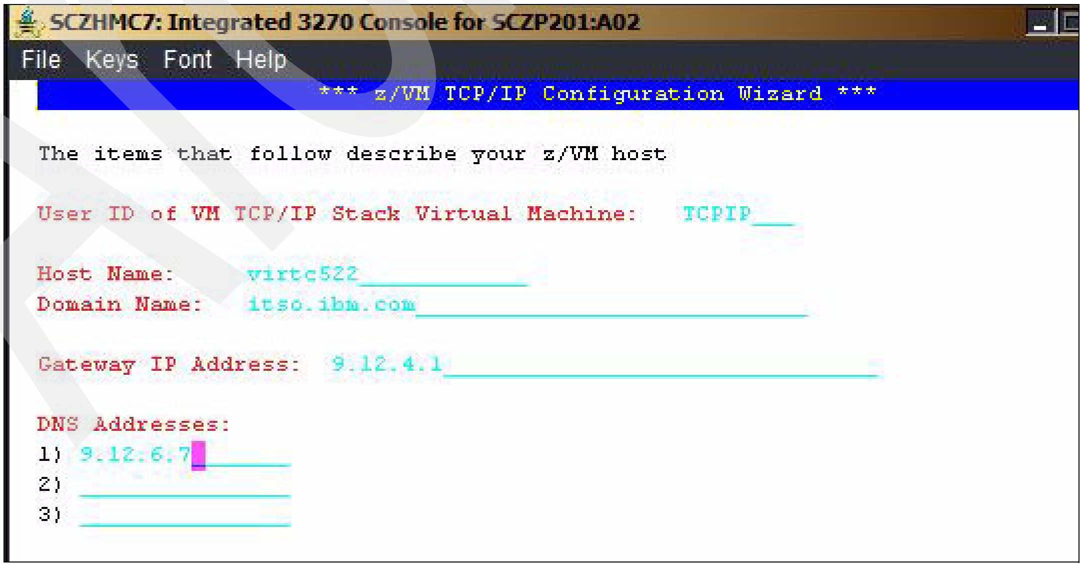
The **IPWIZARD** command is on the `MAINT 193` disk. You will need to access it as file mode G using the **ACCESS** command so that you will pick up **IPWIZARD** from that minidisk.

1. Access the `MAINT 193` disk:

```
==> acc 193 g
```

2. Invoke **IPWIZARD**.

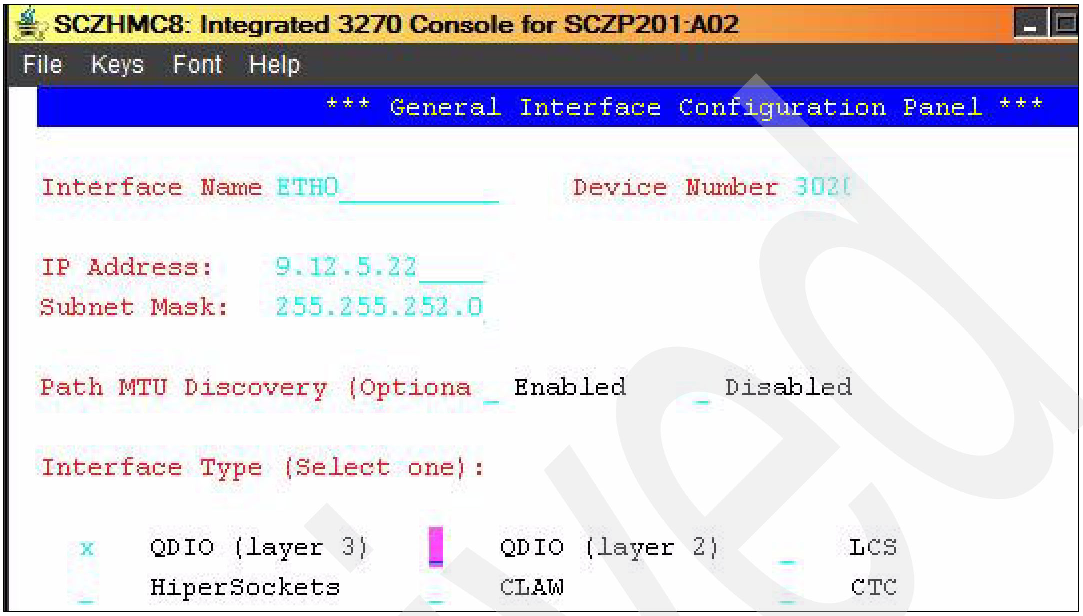
```
==> ipwizard
```



```
                    *** z/VM TCP/IP Configuration Wizard ***

The items that follow describe your z/VM host

User ID of VM TCP/IP Stack Virtual Machine:   TCPIP___

Host Name:      virtc522____________
Domain Name:    itso.ibm.com____________________________

Gateway IP Address:  9.12.4.1________________________________

DNS Addresses:
1) 9.12.6.7_______
2) _______________
3) _______________
```

*Figure 4-10 IPWIZARD window 1*

3. The z/VM TCP/IP Configuration Wizard opens as shown in Figure 4-10. The first field, User ID, should always be `TCPIP`. Obtain the remaining values from the 2.7.1, “z/VM resources worksheet” on page 16 and press **F8**.

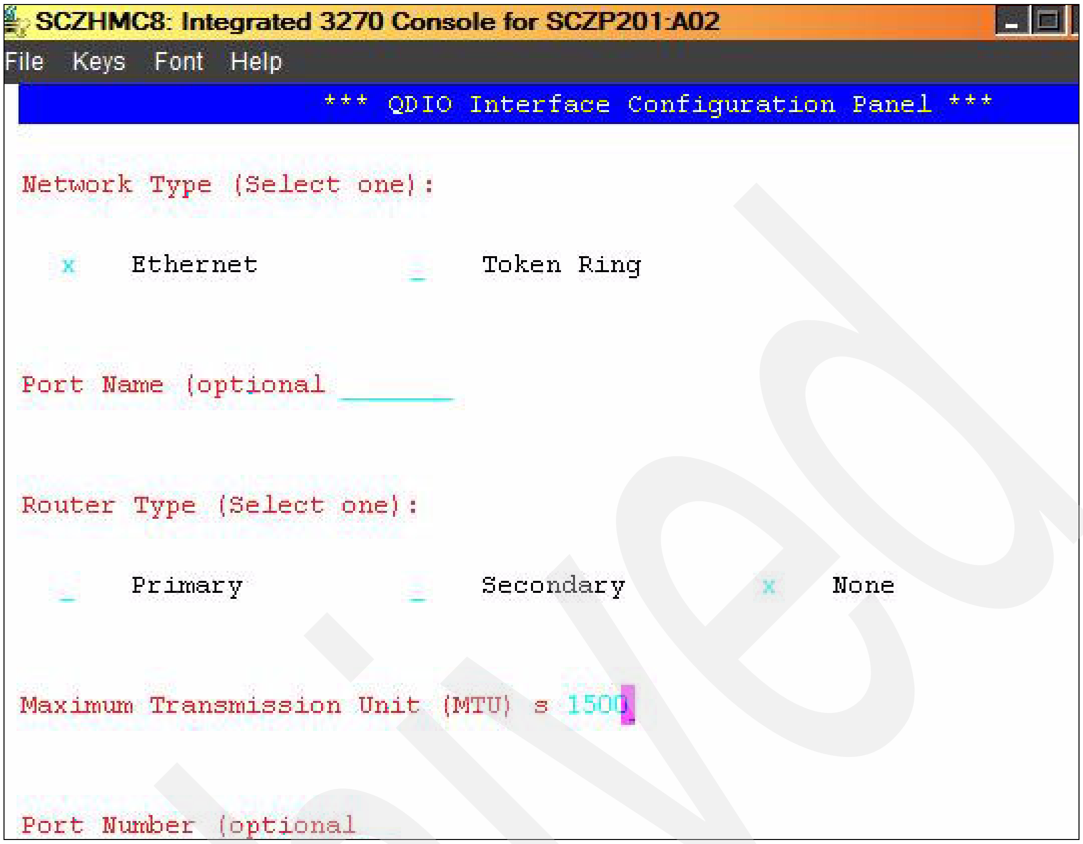


```
                    *** General Interface Configuration Panel ***

Interface Name ETH0_________         Device Number 302(

IP Address:     9.12.5.22____
Subnet Mask:    255.255.252.0

Path MTU Discovery (Optiona _ Enabled       _ Disabled

Interface Type (Select one):

   x    QDIO (layer 3)     _    QDIO (layer 2)    _     LCS
   _    HiperSockets       _    CLAW              _     CTC
```

*Figure 4-11 IPWIZARD window 2*

4. An Interface Name of `ETH0` is arbitrary but recommended. The Device Number will be the starting address of the OSA triplet that the z/VM stack will use. The IP address that must be routed to the OSA card will become the TCP/IP address of the z/VM system. The Interface Type will typically be `QDIO (layer 3)` with modern OSA devices. When you finish, press **F8**.

> **Note:** To utilize QDIO (layer 2), certain prerequisites must be met. Consult with your system administrator regarding these prerequisites.



```
                    *** QDIO Interface Configuration Panel ***

Network Type (Select one):

  x   Ethernet            _   Token Ring


Port Name (optional ________


Router Type (Select one):

  _   Primary             _   Secondary           x   None


Maximum Transmission Unit (MTU) s 1500


Port Number (optional
```

Figure 4-12   IPWIZARD window 3

5. In general, a value for the Port Name is no longer necessary and a Router Type of None is recommended. Press **F5** to complete the wizard.

```
DTCIPW2508I DTCIPWIZ EXEC is attempting to create the necessary
DTCIPW2508I configuration files
```

6. Enter 1 to restart the TCP/IP stack:

```
The TCP/IP stack (TCPIP) must be restarted as part of this procedure
Would you like to restart and continue?
Enter 0 (No), 1 (Yes) 1
USER DSC    LOGOFF AS  TCPIP     USERS = 2     FORCED BY MAINT
...
Successfully PINGed Interface (9.12.5.22)
Successfully PINGed Gateway (9.12.4.1)
Successfully PINGed DNS (9.12.6.7)
DTCIPW2519I Configuration complete; connectivity has been verified
DTCIPW2520I File PROFILE TCPIP created on TCPIP 198
DTCIPW2520I File TCPIP DATA created on TCPIP 592
DTCIPW2520I File SYSTEM DTCPARMS created on TCPIP 198
HCPINP8392I IPWIZARD EXEC ENDED SUCCESSFULLY
DMSVML2061I TCPIP 592 released
```

7. At this point your z/VM TCP/IP stack should be running, and you should be able to ping it from another system.

   If the **IPWIZARD** fails, you must continue debugging it until it succeeds. Double-check all values. Verify that the TCP/IP network and OSA information you were given are properly associated.

**HMC Integrated 3270 Console or 3270 emulator?** At this point, z/VM should be accessible over the network. You can continue working at the HMC, or you can access your new system using a 3270 emulator.

If you want to switch to 3270 emulator, first **LOGOFF** from MAINT or issue **DISConnect** on the Integrated 3270 Console. Note the following points:

- If you log off, the session is ended; it is analogous to shutting and powering down a PC.
- If you disconnect, your session remains where it is and is resumed when you log back on; it is analogous to turning off a PC monitor.

In general, you should **LOGOFF** from system administration user IDs such as MAINT. However, you should always **DISCONNECT** from z/VM service machines such as TCPIP and user IDs running Linux, because logging off from them will terminate the service or crash Linux.

## 4.3 Configuring the XEDIT profile

**Logon** to MAINT (if not logged on already).

The **XEDIT** command looks for the file XEDIT PROFILE configuration file when it is invoked. Many z/VM user IDs do not have such a personal or shared system file, so all **XEDIT** default values are in effect. The MAINT 191 (A) disk has a PROFILE XEDIT so when you are editing files on MAINT, the values in this profile are usually in effect.

If you have never used XEDIT before, refer to , “XEDIT quick reference sheet” on page 212, for more information about this topic. The z/VM 5.4 PDF library is available on the Web at:

http://www-03.ibm.com/systems/z/os/zos/bkserv/zvmpdf/#zvm54

Search for the *XEDIT User’s Guide and Command Reference*. Also there is an old manual available online:

http://ukcc.uky.edu/ukccinfo/391/xeditref.html

One default setting that can be dangerous, especially if you use F12 to retrieve commands, is that PF12 is set to the **FILE** subcommand. Sometimes you may not want to save your changes with the stroke of one key. We recommend that you set PF12 to the **?** subcommand, which has the effect of a retrieve key:

```
==> copy profile xedit a profile xediorig a (oldd
==> x profile xedit a
```

Before:

```
SET PF12 FILE
```

After:

```
SET PF12 ?
```

Save your changes with the **FILE** subcommand.

## 4.4 Customizing the SYSTEM CONFIG file

The first configuration file read when z/VM IPLs is the `SYSTEM CONFIG` file. The following changes are recommended:

- Change the system name.
- Increase retrieve key capacity.
- Allow virtual disks (VDISKs) to be created.
- Turn off the Disconnect Timeout. This will prevent idle disconnected users from being forced off the system.
- Define a virtual switch (VSWITCH) that will be used for Linux networking.

To make these changes, perform the following steps:

1. To edit the `SYSTEM CONFIG` file, the `MAINT CF1` minidisk must be released as a CP disk using the **`CPRELASE`** command. The CP disks are queried using the **`QUERY CPDISK`** command. Note the `MAINT CF1 disk` is accessed as CP disk A before it is released, but not after.

```
==> q cpdisk
Label  Userid   Vdev Mode Stat Vol-ID Rdev Type   StartLoc     EndLoc
MNTCF1 MAINT    0CF1  A   R/O  540RES D850 CKD          39        158
MNTCF2 MAINT    0CF2  B   R/O  540RES D850 CKD         159        278
MNTCF3 MAINT    0CF3  C   R/O  540RES D850 CKD         279        398
==> cprel a
CPRELEASE request for disk A scheduled.
HCPZAC6730I CPRELEASE request for disk A completed.
==> q cpdisk
Label  Userid   Vdev Mode Stat Vol-ID Rdev Type   StartLoc     EndLoc
MNTCF2 MAINT    0CF2  B   R/O  540RES D850 CKD         159        278
MNTCF3 MAINT    0CF3  C   R/O  540RES D850 CKD         279        398
```

2. After it is released, you are able to access the `MAINT CF1` disk read-write. Use the **`LINK`** command with multi-read (**`MR`**) parameter and **`ACCESS`** command to get read-write access as your F disk.

```
==> link * cf1 cf1 mr
==> acc cf1 f
```

3. Make a backup copy of the vanilla `SYSTEM CONFIG` file using the **`COPYFILE`** command with the **`OLDDATE`** parameter so that the time stamp of the file is not modified. Note that because the target file name (`system`) and mode (`f`) are the same, the equal sign (`=`) can be used as a wildcard.

```
==> copy system config f = conforig = (oldd
```

4. Edit the original file:

```
==> x system config f
```

5. The system name is set to `ZVMV5R40` by default in the `System_Identifier_Default` statement. You can search for it using the / subcommand:

```
====> /System_Identifier_D
```

   Modify this to the new name of your system. In this example, `VMLINUXA` is used.

```
System_Identifier_Default <VMLINUXA>
```

6. Next look for the `Features` statement. You can search for it again or you can use **`F8`** to page down. The following changes and additions are recommended:

   – Increase the number of commands that can be retrieved from 20 to 99.
   – Set the Disconnect_Timeout to **`off`** so disconnected users do not get forced off.
   – Allow unlimited VDISKs to be created by users by changing `Userlim` to **`infinite`** and by adding the **`Syslim infinite`** clause, as shown here:

```
Features ,
  Disable ,                     /* Disable the following features */
    Set_Privclass ,             /* Disallow SET PRIVCLASS command */
    Auto_Warm_IPL ,             /* Prompt at IPL always           */
    Clear_TDisk   ,             /* Don't clear TDisks at IPL time */
  Retrieve ,                    /* Retrieve options               */
    Default  99 ,               /* Default.... default is 20      */
    Maximum  255 ,              /* Maximum.... default is 255     */
  MaxUsers noLimit ,            /* No limit on number of users    */
  Passwords_on_Cmds ,           /* What commands allow passwords? */
    Autolog  yes ,              /* ... AUTOLOG does               */
    Link     yes ,              /* ... LINK does                  */
    Logon    yes ,              /* ... and LOGON does, too        */
  Disconnect_Timeout off ,      /* Don't force disconnected users */
  Vdisk ,                       /* Allow VDISKS for Linux swaps   */
    Syslim  infinite  ,
    Userlim  infinite
```

7. Define a VSWITCH:

   Use the **BOTTOM** subcommand to go to the bottom of the file. Add some lines (you can use the **XEDIT** add subcommand **a3**). Define a VSWITCH and set the MAC address prefix. This will set the first three bytes of the MAC address created for each virtual NIC. If you have a multiple z/VM systems, increment this value to avoid having identical MAC addresses created. The last three bytes of the MAC address are automatically incremented by z/VM as they are assigned, so they will be unique on each z/VM system. Modify the two starting addresses of the OSA triplets (3024 and 3028 in this example) to those you specified in 2.7.1, “z/VM resources worksheet” on page 16.

```
====> bot
====> a3
/* define vswitch named vsw1 and set MAC address prefixes to 02-00-01 */
define vswitch vsw1 rdev <3024> <3028>
vmlan macprefix 020001
```

8. Save your changes with the XEDIT **FILE** subcommand:

```
====> file
```

9. Test your changes with the **CPSYNTAX** command, which is on the MAINT 193 disk:

```
==> acc 193 g
==> cpsyntax system config f
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
```

   Pay attention to the output. If you receive any syntax errors, fix them before proceeding.

10. Release and detach the MAINT CF1 disk with the **RELEASE** command and **DETACH** parameter. Then put it back online with the **CPACCESS** command:

```
==> rel f (det
DASD 0CF1 DETACHED
==> cpacc * cf1 a
CPACCESS request for mode A scheduled.
HCPZAC6732I CPACCESS request for MAINT's 0CF1 in mode A completed.
```

11. Verify that the CP disk A has been accessed using the **QUERY CPDISK** command:

```
==> q cpdisk
Label  Userid   Vdev Mode Stat Vol-ID Rdev Type   StartLoc     EndLoc
MNTCF1 MAINT    0CF1  A   R/O  540RES D850 CKD          39        158
MNTCF2 MAINT    0CF2  B   R/O  540RES D850 CKD         159        278
MNTCF3 MAINT    0CF3  C   R/O  540RES D850 CKD         279        398
```

   Note that all three CP disks are now accessed.

## 4.5 Configuring TCP/IP to start at IPL time

Configure the `TCPIP` service machine to be started when z/VM IPLs. This is commonly accomplished from the `AUTOLOG1` **`PROFILE EXEC`**. If the **`noautolog`** parameter is *not* specified when z/VM is IPLed, then the `AUTOLOG1` virtual machine is started.

Because this virtual machine IPLs CMS, the `PROFILE EXEC` that is found on its A disk is run. This is analogous to the `/etc/profile` file on Linux and the `autoexec.bat` on DOS systems.

1. Logoff from `MAINT`.

   ```
   ==> log
   ```

2. You should see a new logon window. Logon to AUTOLOG1. Again, the password is the same as the user ID.
3. At the `VM READ` prompt, enter the command **`ACCESS (NOPROF`** so that the `PROFILE EXEC` is not run.

   ```
   LOGON AUTOLOG1
   z/VM Version 5 Release 4.0, Service Level 0801 (64-bit),
   built on IBM Virtualization Technology
   There is no logmsg data
   FILES:   NO RDR,   NO PRT,   NO PUN
   LOGON AT 10:06:05 EDT THURSDAY 05/15/08
   DMSIND2015W Unable to access the Y-disk. Filemode Y (19E) not accessed
   z/VM V5.4.0    2008-05-13 14:27
   ==> acc (noprof
   ```

4. Copy the `PROFILE XEDIT` from the `MAINT 191` disk so XEDIT sessions will have a common interface among user IDs.
   a. Use the **`VMLINK`** command to both link to the disk read-only and to access it as the highest available file mode. The default read password is **read**:

   ```
   ==> vmlink maint 191
   ENTER READ PASSWORD:
   ==> read
   DMSVML2060I MAINT 191 linked as 0120 file mode Z
   ```

   b. Copy the `PROFILE XEDIT` to your A disk:

   ```
   ==> copy profile xedit z = = a
   ```

5. Make a backup copy of the `PROFILE EXEC` and edit it:

   ```
   ==> copy profile exec a = execorig =
   ==> x profile exec
   ```

6. You should see the text in the top half of the following example. Modify it as follows.
   a. You can safely delete the `Address Command` line.
   b. Add a line to start the TCPIP user ID using the **`XAUTOLOG`** command and keep two statements that start the VSWITCH controllers.
   c. Add a line to **`logoff`** from `AUTOLOG1` when the EXEC is complete. There is no need to keep that virtual machine running, because its sole purpose is to run the `PROFILE EXEC`.

   Before:

   ```
   /***************************/
   /*  Autolog1 Profile Exec  */
   /***************************/

   Address Command
   'CP XAUTOLOG VMSERVS'
   ```

```
'CP XAUTOLOG VMSERVU'
'CP XAUTOLOG VMSERVR'
'CP XAUTOLOG DTCVSW1'
'CP XAUTOLOG DTCVSW2'
```

After:

```
/***************************/
/*  Autolog1 Profile Exec  */
/***************************/
'cp xautolog tcpip'                /* start up TCPIP */
'CP XAUTOLOG VMSERVS'
'CP XAUTOLOG VMSERVU'
'CP XAUTOLOG VMSERVR'
'CP XAUTOLOG DTCVSW1'
'CP XAUTOLOG DTCVSW2'
'cp logoff'                        /* logoff when done */
```

7. Save your changes with the **FILE** subcommand:

```
====> file
```

8. **Logoff** from AUTOLOG1:

```
==> log
```

When your z/VM system IPLs, the TCP/IP stack should now come up automatically (as long as you do *not* specify the **notautolog** parameter at IPL time).

### 4.5.1 Renaming the TCPIP configuration file

It is recommended that you change the name of the main TCPIP configuration file from PROFILE TCPIP to <system_ID> TCPIP, where <system_ID> is the name of your new z/VM system. This is to avoid the possibility that the PROFILE TCPIP file will be overwritten when applying maintenance.

1. **Logoff** from AUTOLOG1 and **logon** to TCPMAINT. The PROFILE TCPIP file is on the TCPMAINT 198 disk, which is accessed as the D disk.
2. Make a backup copy of the original PROFILE TCPIP, then rename it to <SYSTEM_ID> TCPIP (where <SYSTEM_ID> is VMLINUXA in this example). When the TCP/IP service machine starts, it will search for this file before the file PROFILE TCPIP.

```
==> copy profile tcpip d = tcpiorig = (oldd
==> rename profile tcpip d <vmlinuxa> = =
```

3. You have now backed up and renamed your TCP/IP profile. You can verify using the **LISTFILE** command:

```
==> listfile * * d
VMLINUXA TCPIP    D1
SYSTEM   DTCPARMS D1
PROFILE  TCPIORIG D1
```

### 4.5.2 Copy the PROFILE XEDIT file

Again copy the PROFILE XEDIT from the MAINT 191 disk so XEDIT sessions will have a common interface among user IDs.

1. Use the **VMLINK** command to both link to the disk read-only and to access it as the highest available file mode. The default read password is **read**:

```
   ==> vmlink maint 191
```

```
ENTER READ PASSWORD:
read
DMSVML2060I MAINT 191 linked as 0120 file mode Z
```

2. Copy the `PROFILE XEDIT` to your A disk:

```
==> copy profile xedit z = = a
```

Now, XEDIT sessions on `TCPMAINT` will have the same configuration as on `MAINT`.

### 4.5.3 Configuring the FTP server

Turn on the FTP server by editing the renamed configuration file:

1. Add an `AUTOLOG` statement near the top of the file with `FTPSERVE` as the only entry.
2. In the `PORT` statement, remove the semicolons to uncomment the lines with `FTPSERVE` on them (ports 20 and 21). These changes will cause the FTP server to start when TCPIP is started. The important lines before the file is edited and after are shown:

```
==> x <vmlinuxa> tcpip d
```

Before:

```
; ----------------------------------------------------------------------
OBEY
OPERATOR TCPMAINT MAINT MPROUTE DHCPD REXECD SNMPD SNMPQE LDAPSRV
ENDOBEY
; ----------------------------------------------------------------------
PORT
; 20 TCP FTPSERVE  NOAUTOLOG ; FTP Server
; 21 TCP FTPSERVE            ; FTP Server
  23 TCP INTCLIEN            ; TELNET Server
; 25 TCP SMTP                ; SMTP Server
...
```

After:

```
; ----------------------------------------------------------------------
OBEY
OPERATOR TCPMAINT MAINT MPROUTE ROUTED DHCPD REXECD SNMPD SNMPQE
ENDOBEY
; ----------------------------------------------------------------------
AUTOLOG
  FTPSERVE  0
ENDAUTOLOG

PORT
  20   TCP FTPSERVE  NOAUTOLOG ; FTP Server
  21   TCP FTPSERVE            ; FTP Server
  23   TCP INTCLIEN            ; TELNET Server
; 25   TCP SMTP                ; SMTP Server
...
```

3. Save your changes with the **`FILE`** subcommand:

```
====> file
```

You could continue to configure the system, but at this time we recommend that you test your changes by shutting down and reIPLing the system.

### 4.5.4 Shutting down and reIPLing the system

You may want to be able to shut down and reIPL z/VM without having to access the HMC. Often, the HMC will be logged off and thus the Integrated 3270 console (`SYSG`) will not be available.

Because of these factors, it is useful to use the System Console (`SYSC`, which has a title of Operating System Messages on the HMC) in order to shut down z/VM and reIPL it without needing to use the console. This console is always accessible, whether you are logged on to the HMC or not.

z/VM messages during both the shutdown and reIPL process will be written to the system console, but often you will be able to ignore them; you just want your system back in a few minutes over the network.

1. Pass the parameter **`IPLPARMS CONS=SYSC`** to the **`SHUTDOWN REPIL`** command:

   ```
   ==> shutdown reipl iplparms cons=sysc
   ```

   You will lose your session, but it should come back in a few minutes as described. When your system is back up, perform the following commands:

2. Start a 3270 session and logon as `MAINT`. This shows that there is TCP/IP access to z/VM.
3. Query the new VSWITCH:

   ```
   ==> q vswitch
   VSWITCH SYSTEM VSW1     Type: VSWITCH Connected: 0    Maxconn: INFINITE
     PERSISTENT  RESTRICTED    NONROUTER                 Accounting: OFF
     VLAN Unaware
     MAC address: 02-00-01-00-00-01
     State: Ready
     IPTimeout: 5         QueueStorage: 8
     RDEV: 3024.P00 VDEV: 3024 Controller: DTCVSW2
     RDEV: 3028.P00 VDEV: 3028 Controller: DTCVSW1  BACKUP
   ```

   You should see that the VSWITCH `VSW1` exists, that the OSA devices you specified are being used and that there are two built-in VSWITCH controllers, `DTCVSW1` and `DTCVSW2`.

4. Use the **`QUERY VDISK`** and **`QUERY RETRIEVE`** commands to see the changes made to the `Features` statement in the `SYSTEM CONFIG` file:

   ```
   ==> q retrieve
   99 buffers available.  Maximum of 255 buffers may be selected.
   ==> q vdisk userlim
   VDISK USER   LIMIT IS INFINITE
   ==> q vdisk syslim
   VDISK SYSTEM LIMIT IS INFINITE,               0 BLK IN USE
   ```

This shows that the changes to the `SYSTEM CONFIG file` have taken effect.

## 4.6 Adding paging volumes

The z/VM operating system resides on the first three CP volumes (or one volume, if you are installing onto 3390-9s). z/VM 5.4 is installed with one full paging volume and one full spool volume. A single spool volume is probably adequate for Linux needs, but a single paging volume is probably not. We recommend that you add five paging volumes so you will have a total of six (or one more 3390-9 if installing onto 3390-9s, which will give the same total of 20034 cylinders of page space).

If you do not have sufficient DASD, this number can be reduced. Having adequate paging space will give you plenty of headroom to add more Linux virtual machines. A rule of thumb for the amount of paging space is to have twice as much as the total of all memory for all running Linux user IDs combined.

## 4.6.1 Formatting the paging volumes

Before adding paging volumes to the system, the DASD volumes to be used for minidisk space (PERM) and paging space (PAGE) must be formatted. Normally this is done one volume at a time using the **CPFMTXA** command. If you have just a few volumes, that is fine, but when you have many volumes to format, the process of running **CPFMTXA** can become time-consuming and tedious, which can lead to errors.

Therefore, a REXX EXEC named **CPFORMAT** is provided to allow you to format many volumes with a single command. The source code for this EXEC is located in , “The CPFORMAT EXEC” on page 216. It is a wrapper around **CPFMTXA**. To use this EXEC, each DASD to be formatted must first be attached with the virtual device address the same real device address (using **ATTACH <realDev> ***).

**Note:** This EXEC will label the volumes according to the convention described in 2.2.1, “Volume labeling convention” on page 9. If you want different volume labels, you can use the **CPFMTXA** command and manually specify each volume label, or you can modify the REXX EXEC.

### Getting the CPFORMAT EXEC to z/VM

**Logoff** from MAINT so you will be able to get the MAINT 191 disk in read-write mode using FTP.

**Important:** At this point, you will need access to the NFS server described in Chapter 6, “Configuring a Network File System server for RHEL 5.2” on page 85, in order to get the files **CPFORMAT EXEC**.

Start an SSH (putty) session to the NFS server and change to the vm/ directory that was created when you untarred the files associated with this book. Verify that the file CPFORMAT.EXEC exists. Note that the directory name will be one of the following, depending on the Linux distribution you are working with:

```
# cd /nfs/virt-cookbook-S10SP2/vm    // if you are working with SLES 10 SP2
# cd /nfs/virt-cookbook-RH52/vm      // if you are working with RHEL 5.2
# ls CPFORMAT*
CPFORMAT.EXEC
```

Now start an FTP session to z/VM. If you get a reply from the FTP server, it shows that you correctly configured it on the z/VM TCPMAINT user ID. Issue the **ASCII** subcommand to be sure the ASCII characters are converted to EBCDIC, and use the **PUT** subcommand to copy the file.

```
# ftp <9.12.5.22>
Name (9.12.5.22:root): maint
331-Password: maint
230-220-FTPSERVE IBM VM Level 540 at VMLINUXA.ITSO.IBM.COM, 15:47:48 EDT TUESDAY
2008-05-13
230-MAINT logged in; working directory = MAINT 191
...
ftp> ascii
...
```

```
ftp> put CPFORMAT.EXEC
...
ftp> quit
```

## Using the CPFORMAT EXEC

Log back into MAINT. You should now have access to the CPFORMAT EXEC. You can get brief help on **CPFORMAT** by using a parameter of **?**:

```
==> cpformat ?

Synopsis:

  Format one or a range of DASD as page, perm, spool or temp disk space
  The label written to each DASD is M<t><xxxx> where:
    <t> is type - P (page), M (perm), S (spool) or T (Temp disk)
    <xxxx> is the 4 digit address

Syntax is:
                                             .-PAGE-.
   >>--CPFORMAT--.-rdev--------------.--AS---+-PERM-+---------><
                 | <---------------< |       '-SPOL-'
                 '-rdev1-rdev2-------'
```

The following example illustrates how to attach five 3390-3 volumes and use **CPFORMAT** to format them as paging space. Refer to the planning work sheets that you filled out in 2.7.2, “z/VM DASD worksheet” on page 17.

**For 3390-9 volumes:** If you are installing onto 3390-9s, only one more paging volume may be adequate to start. This will give you two full volumes, or the same equivalent of six 3390-3s.

For example, if your z/VM system installed onto DASD at D850-D852 and you plan to put the second paging volume on D853, then only that volume needs to be added.

Rather than using five consecutive DASD addresses, you may consider using DASD from different address ranges in an attempt to locate the paging volumes on different *ranks* in your disk array. This should enable z/VM to page more efficiently.

1. The DASD that will be used for paging volumes are at real addresses **D855, D950, D951, DA50** and **DB50.** First query the DASD devices to see their status:

```
==> q <d855 d950 d951 da50 db50>
DASD D855 NWD855  , DASD D950 NWD950  , DASD D951 NWD951  , DASD DA50 NWDA50
DASD DB50 NWDB50
```

2. Attach the devices to MAINT (the last parameter of “*” means the current user ID) using the **ATTACH** command:

```
==> att <d855 d950 d951 da50 db50> *
D855 D950 D951 DA50 DB50 ATTACHED TO MAINT
```

3. Use the **CPFORMAT** command with the **AS PAGE** parameter:

```
==> cpformat d855 d950 d951 da50 db50 as page

Format the following DASD:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    D855 MAINT    D855 3390  NWD855 D855          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    D950 MAINT    D950 3390  NWD950 D950          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
```

```
MAINT    D951 MAINT    D951 3390  NWD951 D951          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA50 MAINT    DA50 3390  NWDA50 DA50          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DB50 MAINT    DB50 3390  NWDB50 DB50          0       3339

WARNING - this will destroy data!
ARE YOU SURE you want to format the DASD as PAGE space (y/n)?
y
...
ICK00002I ICKDSF PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
D855 D950 D951 DA50 DB50 DETACHED
D855 D950 D951 DA50 DB50 ATTACHED TO MAINT

DASD status after:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    D855 MAINT    D855 3390  MPD855 D855          0       3339
MAINT    D950 MAINT    D950 3390  MPD950 D950          0       3339
MAINT    D951 MAINT    D951 3390  MPD951 D951          0       3339
MAINT    DA50 MAINT    DA50 3390  MPDA50 DA50          0       3339
MAINT    DB50 MAINT    DB50 3390  MPDB50 DB50          0       3339
```

This formatting job should run for about 10 to 50 minutes, depending on many factors. Meanwhile, you can continue to format more volumes for PERM (minidisk) space, as described in the next section.

### 4.6.2 Formatting DASD for minidisks

You could wait until **CPFORMAT** of the five paging volumes completes on MAINT, and then format more volumes for PERM or minidisk space. However, you can also get more format jobs going by using a different user ID.

1. Start a new 3270 session and logon as SYSMAINT. Press **Enter** when you get the VM READ prompt:

```
LOGON SYSMAINT
z/VM Version 5 Release 4.0, Service Level 0801 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 13:30:28 EDT WEDNESDAY 05/14/08
z/VM V5.4.0    2008-05-13 14:27

DMSACP113S A(191) not attached or invalid device address
DMSACP723I D (192) R/O
```

2. Link to the MAINT 191 disk read-only to pick up the CPFORMAT EXEC. This can be done with the **VMLINK** command (**VMLINK** performs the **LINK** and **ACCESS** commands, with a read-only link and accessing the highest free file mode letter).

```
==> vmlink maint 191
DMSVML2060I MAINT 191 linked as 0192 file mode D
```

3. Query the eight devices that will be used for the controller, LNXMAINT (common CMS files), and for the golden image. In this example they are D856-D857 D952-D957.

```
==> q <d856-d857 d952-d957>
DASD D856 NWD856  , DASD D857 NWD857  , DASD D952 NWD952  , DASD D953 NWD953
DASD D954 NWD954  , DASD D955 NWD955  , DASD D956 NWD956  , DASD D957 NWD957
```

4. Attach the eight volumes that will be used for the controller, the common CMS disk, and the golden image. In this example, it is the DASD at the following addresses:

```
==> att <d856-d857 d952-d957> *
DASD D856 ATTACHED TO SYSMAINT D856 WITH DEVCTL HYPERPAV BASE
DASD D857 ATTACHED TO SYSMAINT D857 WITH DEVCTL HYPERPAV BASE
DASD D952 ATTACHED TO SYSMAINT D952 WITH DEVCTL HYPERPAV BASE
DASD D953 ATTACHED TO SYSMAINT D953 WITH DEVCTL HYPERPAV BASE
DASD D954 ATTACHED TO SYSMAINT D954 WITH DEVCTL HYPERPAV BASE
DASD D955 ATTACHED TO SYSMAINT D955 WITH DEVCTL HYPERPAV BASE
DASD D956 ATTACHED TO SYSMAINT D956 WITH DEVCTL HYPERPAV BASE
DASD D957 ATTACHED TO SYSMAINT D957 WITH DEVCTL HYPERPAV BASE
```

5. Invoke the **CPFORMAT** command against these volumes using the parameter **as perm**:

```
==> cpformat <d856-d857 d952-d957> as perm

Format the following DASD:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D856 SYSMAINT D856 3390  NWD856 D856          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D857 SYSMAINT D857 3390  NWD857 D857          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D952 SYSMAINT D952 3390  NWD952 D952          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D953 SYSMAINT D953 3390  NWD953 D953          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D954 SYSMAINT D954 3390  NWD954 D954          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D955 SYSMAINT D955 3390  NWD955 D955          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D956 SYSMAINT D956 3390  NWD956 D956          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D957 SYSMAINT D957 3390  NWD957 D957          0       3339

WARNING - this will destroy data!
ARE YOU SURE you want to format the DASD as PERM space (y/n)?
y
...
DASD status after:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
SYSMAINT D856 SYSMAINT D856 3390  MMD856 D856          0       3339
SYSMAINT D857 SYSMAINT D857 3390  MMD857 D857          0       3339
SYSMAINT D952 SYSMAINT D952 3390  MMD952 D952          0       3339
SYSMAINT D953 SYSMAINT D953 3390  MMD953 D953          0       3339
SYSMAINT D954 SYSMAINT D954 3390  MMD954 D954          0       3339
SYSMAINT D955 SYSMAINT D955 3390  MMD955 D955          0       3339
SYSMAINT D956 SYSMAINT D956 3390  MMD956 D956          0       3339
SYSMAINT D957 SYSMAINT D957 3390  MMD957 D957          0       3339
```

At this point, you should have page volumes being formatted on `MAINT` and PERM or minidisk volumes being formatted on `SYSMAINT`.

When completed, you should have eight newly formatted volumes that can be used as minidisks.

### 4.6.3 Updating the SYSTEM CONFIG file

Now that the PAGE and PERM volumes are ready for use, they must be added to the SYSTEM CONFIG file so that z/VM can use them. Follow these steps to update the SYSTEM CONFIG file:

1. **Logon** to MAINT.
2. The following example uses the same steps to access the MAINT CF1 disk read-write that you used earlier:

   ```
   ==> q cpdisk
   Label  Userid   Vdev Mode Stat Vol-ID Rdev Type   StartLoc     EndLoc
   MNTCF1 MAINT    0CF1  A   R/O  540RES D850 CKD          39        158
   MNTCF2 MAINT    0CF2  B   R/O  540RES D850 CKD         159        278
   MNTCF3 MAINT    0CF3  C   R/O  540RES D850 CKD         279        398
   ==> cprel a
   CPRELEASE request for disk A scheduled.
   HCPZAC6730I CPRELEASE request for disk A completed.
   ==> link * cf1 cf1 mr
   ==> acc cf1 f
   ```

   It is useful to remember this sequence of steps.
3. Make a copy of the working SYSTEM CONFIG file using the WRKS suffix convention:

   ```
   ==> copy system config f = confwrks =
   ```

4. Edit the SYSTEM CONFIG file and specify each of the new page volumes (PAGE) by name as CP_Owned. When you system IPLs, it will pick up these as paging volumes.

   ```
   ==> x system config f
   ====> /cp_owned
   ...
   /*****************************************************************/
   /*                    CP_Owned Volume Statements                 */
   /*****************************************************************/

       CP_Owned   Slot   1  540RES
       CP_Owned   Slot   2  540SPL
       CP_Owned   Slot   3  540PAG
       CP_Owned   Slot   4  540W01
       CP_Owned   Slot   5  540W02
       CP_Owned   Slot   6  <MPD855>
       CP_Owned   Slot   7  <MPD950>
       CP_Owned   Slot   8  <MPD951>
       CP_Owned   Slot   9  <MPDA50>
       CP_Owned   Slot   10 <MPDB50>
       CP_Owned   Slot   11  RESERVED
       CP_Owned   Slot   12  RESERVED
       CP_Owned   Slot   13  RESERVED
   ...
   ```

5. Move down to the User_Volume_List section. User volumes (PERM) can be specified individually with the User_Volume_List statement, or with wild cards using the User_Volume_Include statement. If you are using the labelling convention enforced by the **CPFORMAT EXEC**, then add the following single line to include all PERM space as volume labels all begin with MM (the labeling convention used by the CPFORMAT EXEC).

   ```
   ====> /user_v
   /**********************************************************************/
   /*                            User_Volume_List                         */
   /* These statements are not active at the present time.  They are     */
   /* examples, and can be activated by removing the comment delimeters  */
   /**********************************************************************/
   ```

```
User_Volume_Include MM*
/*  User_Volume_List USRP01    */
/*  User_Volume_List USRP02    */
...
====> file
```

6. Save your changes with the **FILE** subcommand. Verify the integrity of the changes with the **CPSYNTAX** command:

```
==> acc 193 g
==> cpsyntax system config f
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
```

7. After you confirm that there are no syntax errors, put the `MAINT CF1` disk back online. The following example shows how you did this previously:

```
==> rel f (det
DASD 0CF1 DETACHED
==> cpacc * cf1 a
CPACCESS request for mode A scheduled.
HCPZAC6732I CPACCESS request for MAINT's 0CF1 in mode A completed.
==> q cpdisk
Label  Userid   Vdev Mode Stat Vol-ID Rdev Type   StartLoc      EndLoc
MNTCF1 MAINT    0CF1  A   R/O  520RES 0200 CKD          39          83
MNTCF2 MAINT    0CF2  B   R/O  520RES 0200 CKD          84         128
MNTCF3 MAINT    0CF3  C   R/O  520RES 0200 CKD         129         188
```

### 4.6.4 Testing the changes

We recommend that you again shut down and reIPL to test the changes. Before you shut down, note that you have only one page volume (`540PAG`) using the **QUERY ALLOC PAGE** command. Your output should look similar to the following:

```
==> q alloc page
EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
540PAG D852          1       3338 600840      0      0   0%
                                  ------ ------        ----
SUMMARY                           600840      0          0%
USABLE                            600840      0          0%
```

Now shut the system down again with the command **SHUTDOWN REIPL IPLPARMS CONS=SYSC**. This is analogous to the Linux **reboot** command in that the system attempts to come back up after it shuts down. If you are connected using a 3270 emulator, you will lose your session, but if all goes well, your system will be available again in a few minutes.

```
==> shutdown reipl iplparms cons=sysc
```

After the system comes back, logon as `MAINT` and look at the page space again. You should see that you now have six paging volumes:

```
==> q alloc page
EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
540PAG D852          1       3338 600840      0      0   0%
MPD855 D855          0       3338 601020      0      0   0%
MPD950 D950          0       3338 601020      0      0   0%
MPD951 D951          0       3338 601020     12     12   1%
MPDA50 DA50          0       3338 601020      0      0   0%
MPDB50 DB50          0       3338 601020      0      0   0%
```

```
                                  ------ ------        ----
SUMMARY                            3521K     12          1%
USABLE                             3521K     12          1%
```

The output shows there are six paging volumes constituting 3521 KB pages, or about 14 GB of page space (there are 4 KB per page).

## 4.7 Creating a user ID for common files

Now it is time to define your first z/VM user ID, `LNXMAINT`. It will be used to store files that will be shared by Linux user IDs. Before starting, make a copy of the original `USER DIRECT` file:

```
==> copy user direct c = direorig = (oldd
```

### 4.7.1 Define the user in the USER DIRECT file

A small, 20-cylinder minidisk is allocated at virtual address 191. A larger, 300-cylinder minidisk (approximately 225 MB), which is to be shared by many guests, is defined at virtual address 192. Use the next free DASD designated as PERM space on your worksheet (2.7.2, “z/VM DASD worksheet” on page 17). Cylinder 0 should always be reserved for the label; therefore, you start minidisks at cylinder 1.

1. Edit the `USER DIRECT` file and add the following user ID definition to the bottom of the file:

```
==> x user direct c
====> bottom
====> a 6
...
USER LNXMAINT LNXMAINT 64M 128M BEG        1
 INCLUDE TCPCMSU                           2
 LINK TCPMAINT 592 592 RR                  3
 MDISK 0191 3390 0001 0020 MMD857 MR READ     WRITE    MULTIPLE   4
 MDISK 0192 3390 0021 0300 MMD857 MR ALL      WRITE    MULTIPLE   5
*                                                                 6
...
====> file
```

Note the following explanations:

**1** User ID `LNXMAINT`, same password, default size of 64 MB, with class B, E and G privileges

**2** Include the profile named `TCPCMSU` (defined earlier in the `USER DIRECT` file)

**3** Link to the `TCPMAINT 592` disk read-only for access to **`FTP`** and other TCP/IP commands.

**4** Define a 191 minidisk of `size` 20 cylinders from volume `MMD857`.

**5** Define 192 minidisk of `size 300` cylinders (approximately 225 MB) from volume `MMD857` with the special read password of **`ALL`**, which allows read access from any user ID without a disk password.

**6** An empty comment line for better readability.

2. Whenever an `MDISK` statement is added or modified in the `USER DIRECT` file, always check for overlapping cylinders and gaps. (Gaps will only leave empty disk space, but overlaps can occur because unfortunately z/VM will allow you to define multiple minidisks over the *same* disk space). You can perform this check by using the **`DISKMAP`** command:

```
==> diskmap user
```

```
The minidisks with the END option specified in this directory will not be includ
ed in the following DISKMAP file.

File USER DISKMAP A has been created.
```

3. The file created, `USER DISKMAP`, contains a mapping of all minidisk volumes defined in the `USER DIRECT` file. It will list any overlaps or gaps found on the volumes. Edit the file and turn off the prefix area with the **`XEDIT PREFIX OFF`** subcommand to view 80 columns:

```
==> x user diskmap
====> prefix off
```

4. Search for the text **`overlap`** with the / subcommand:

```
====> /overlap
```

You should see the error message `DMSXDC546E Target not found`. This means that no minidisks are overlapping each other.

Now search for gaps. You should also see some gaps:

```
====> /gap
...
                                            0         500         501    GAP
$$$$$$   DATAMOVE   5F0    3380      00501      00501      00001
         DATAMOVE   5FF    3380      00502      00502      00001

------------------------------------------------------------------------

VOLUME   USERID     CUU   DEVTYPE      START         END        SIZE
                                            0           0           1    GAP
$$$LNX   ZVMMAPLX   150    3390      00001      03338      03338

------------------------------------------------------------------------

VOLUME   USERID     CUU   DEVTYPE      START         END        SIZE
                                            0           0           1    GAP
MMD857   LNXMAINT  0191    3390      00001      00020      00020
         LNXMAINT  0192    3390      00021      00320      00300
...
```

Three `GAP`s should be listed on the right side:

- 501 cylinders on the `$$$$$$` volume
- 1 cylinder on the `$$$LNX` volume
- 1 cylinder on volume used for LNXMAINT 191 and 192 disks (`MMD857`, in this example)

There is no need to be concerned about the first two gaps because they are expected, given the layout of the default `USER DIRECT` file. To avoid a 1-cylinder gap being reported on each user volume, we recommend that you use the user ID `$ALLOC$`. This user is set to `NOLOG`, which means it can never be logged onto. Thus, it is not a conventional user ID; instead it is a convenient place to put dummy minidisk definitions for cylinder 0 of all `PERM` volumes.

5. Get out of the file `USER DISKMAP` with the **`QUIT`** command or by pressing **`F3`**.

6. Edit the `USER DIRECT` file again and add a new minidisk definition at virtual address `A04` for the first cylinder of the DASD you added (the label is `MMD857`, in this example):

```
==> x user direct
====> /user $alloc
USER $ALLOC$  NOLOG
 MDISK A01 3390 000 001 540RES R
 MDISK A02 3390 000 001 540W01 R
 MDISK A03 3390 000 001 540W02 R
 MDISK A04 3390 000 001 <MMD857> R
```

7. Save your changes and run **DISKMAP** again. Edit the `USER DISKMAP` file. This time you should see just two gaps for volumes with labels `$$$$$$` and `$$$LNX`. If you search for `$ALLOC$` user ID, you should see the disk map of the volume you added for `LNXMAINT`:

```
==> diskmap user
==> x user diskmap
====> prefix off
====> /$ALLOC
MMD857   $ALLOC$     A04     3390       00000       00000       00001
         LNXMAINT   0191     3390       00001       00020       00020
         LNXMAINT   0192     3390       00021       00320       00300
...
```

8. When you are done, you can quit without saving changes by pressing **F3**.

```
====> F3
```

9. Now that you are sure the minidisk layout is correct, the changes to the `USER DIRECT` file can be brought online using the **DIRECTXA** command:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 5 RELEASE 4.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 43 disk pages
```

If the **DIRECTXA** command fails, correct the problem before proceeding.

At this point, you have now defined your first z/VM user ID named `LNXMAINT`.

## 4.7.2 Logging and customizing the new user ID

Now you should be able to logon to the new user ID and format its two minidisks.

1. **Logoff** from `MAINT` and **logon** to `LNXMAINT`.

```
LOGON LNXMAINT
z/VM Version 5 Release 4.0, Service Level 0801 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 11:32:19 EDT THURSDAY 05/15/08
z/VM V5.4.0    2008-05-13 14:27

DMSACP112S A(191) device error
```

   You should receive an error message ending in `device error`. You receive this error because when CMS is started, it tries to access the user’s 191 minidisk as file mode A. The 191 minidisk has been defined to this user ID; however, it has never been formatted as a CMS file system.

2. To format this disk for CMS, use the **FORMAT** command. It requires a parameter specifying the file mode to access the disk as mode **A**, as shown in the following example:

```
==> format 191 a
DMSFOR603R FORMAT will erase all files on disk A(191). Do you wish to continue?
Enter 1 (YES) or 0 (NO).
1
DMSFOR605R Enter disk label:
lxm191
DMSFOR733I Formatting disk A
DMSFOR732I 20 cylinders formatted on A(191)
```

3. Format the larger 192 disk as the D minidisk, which should take a minute or two:

```
==> format 192 d
DMSFOR603R FORMAT will erase all files on disk D(192). Do you wish to continue?
Enter 1 (YES) or 0 (NO).
1
DMSFOR605R Enter disk label:
lxm192
DMSFOR733I Formatting disk D
DMSFOR732I 300 cylinders formatted on D(192)
```

At this point, you have formatted the two minidisks and accessed them as file modes `A` and `D`. You can confirm this using the **`QUERY DISK`** command:

```
==> q disk
LABEL  VDEV M  STAT   CYL TYPE BLKSZ   FILES  BLKS USED-(%) BLKS LEFT  BLK TOTAL
LXM191 191  A   R/W    20 3390 4096        0           7-00       3593       3600
LXM192 192  D   R/W   300 3390 4096        0          11-00      53989      54000
MNT190 190  S   R/O   100 3390 4096      694      14980-83       3020      18000
MNT19E 19E  Y/S R/O   250 3390 4096     1018      28165-63      16835      45000
```

## 4.7.3 Copying a PROFILE XEDIT

Copy the `PROFILE XEDIT` from the `MAINT 191` disk so XEDIT sessions will have a common interface among user IDs.

1. Use the **`VMLINK`** command to both link to the disk read-only and to access it as the highest available file mode. The default read password is **read**:

```
==> vmlink maint 191
ENTER READ PASSWORD:
==> read
DMSVML2060I MAINT 191 linked as 0120 file mode Z
```

2. Copy the `PROFILE XEDIT` to your A disk:

```
==> copy profile xedit z = = a
```

## 4.7.4 Creating a PROFILE EXEC

Create a simple `PROFILE EXEC` that will be run each time this user ID is logged on.

1. Create the new file using XEDIT and add the following lines (be sure to type the **`A`** file mode so that you do not pick up a `PROFILE EXEC` on another disk). REXX EXECs must always begin with a C language-style comment.

```
==> x profile exec a
====> a 5
/* PROFILE EXEC */
'acc 592 e'
'cp set run on'
'cp set pf11 retrieve forward'
'cp set pf12 retrieve'
====> file
```

This `PROFILE EXEC` accesses the `TCPMAINT 592` disk as file mode `E`, sets CP run on, and sets the retrieve keys per convention.

2. You could test your changes by logging off and logging back on. However, typing the command **`PROFILE`** will produce the same results.

```
==> profile
DMSACP723I E (592) R/O
```

3. By default, CMS tries to access the 191 disk as A and the 192 disk as D. Also, you should have the TCPMAINT 592 disk accessed as E. Again verify using the **QUERY DISK** command:

```
==> q disk
LXM191 191  A   R/W    20 3390 4096        2          9-01       3591       3600
LXM192 192  D   R/W   300 3390 4096        0         11-00      53989      54000
TCM592 592  E   R/O    70 3390 4096      904      10251-81       2349      12600
MNT190 190  S   R/O   100 3390 4096      694      14980-83       3020      18000
MNT19E 19E  Y/S R/O   250 3390 4096     1018      28165-63      16835      45000
MNT191 120  Z   R/O   175 3390 4096       53        391-01      31109      31500
```

4. Verify that your F11 and F12 keys are set to the **RETRIEVE** command using the **QUERY PFKEYS** command:

```
==> q pf
...
PF10 UNDEFINED
PF11 RETRIEVE FORWARD
PF12 RETRIEVE BACKWARD
...
```

### 4.7.5 Copying files associated with this book to LNXMAINT

The z/VM files associated with this book are in the vm/ subdirectory of the NFS server you set up earlier. These files should be stored on the larger 192 disk which is accessed as your D disk. **Logoff** from LNXMAINT so that the 192 disk can be accessed read-write.

Start an SSH session on the NFS server and change directory to the VM files associated with this book. The directory name will be one of the following, depending on the distribution you are working with:

```
# cd /nfs/virt-cookbook-S10S2/vm    // if you are working with SLES 10 SP2
# cd /nfs/virt-cookbook-RH52/vm     // if you are working with RHEL 5.2
```

FTP to z/VM. By default, FTP copies files to your 191 disk, so first change directory to the LNXMAINT 192 disk. The files are in ASCII, so invoke the **ASCII** subcommand to convert them to EBCDIC. Then use the **mput *** subcommand to copy all the files from the vm/ subdirectory to LNXMAINT:

```
# ftp <9.12.5.22>
Connected to 9.12.5.22.
Name (9.12.5.22:root): lnxmaint
331-Password:
Password: lnxmaint
230-LNXMAINT logged in; working directory = LNXMAINT 191
Remote system type is z/VM.
ftp> cd lnxmaint.192
250 Working directory is LNXMAINT 192
ftp> prompt
Interactive mode off
ftp> ascii
...
ftp> mput *
...
ftp> quit
```

**Logon** to LNXMAINT. You should see the following files on your D disk:

```
==> filel * * d
LNXMAINT FILELIST A0  V 169  Trunc=169 Size=8 Line=1 Col=1 Alt=0
Cmd   Filename Filetype Fm Format Lrecl    Records     Blocks   Date     Time
```

```
CHPW540  XEDIT    D1 V        77        194         3  7/01/08  4:30:39
CPFORMAT EXEC     D1 V        79        252         3  7/01/08  4:30:39
LABEL540 EXEC     D1 V        77        116         2  7/01/08  4:30:39
LABEL540 XEDIT    D1 V        77         50         1  7/01/08  4:30:39
LBL540-9 EXEC     D1 V        77        105         2  7/01/08  4:30:39
LBL540-9 XEDIT    D1 V        77         49         1  7/01/08  4:30:39
PROFILE  EXEC     D1 V        63         17         1  7/01/08  4:30:39
SAMPLE   PARM-S10 D1 V        80          9         1  7/01/08  4:30:39
SLES10S2 EXEC     D1 V        69          9         1  7/01/08  4:30:39
SWAPGEN  EXEC     D1 V        72        358         5  7/01/08  4:30:39
```

## 4.8 Customizing system startup and shutdown

When your z/VM system is IPLed, it is often desirable to have important Linux systems also start. Conversely, when you shut down z/VM, it is desirable to have all Linux systems shut down first.

### 4.8.1 Configuring the AUTOLOG1 PROFILE EXEC

We recommend that the following tasks be accomplished by using the AUTOLOG1 PROFILE EXEC.

- Configure Linux to shut down gracefully using the **SET SIGNAL** command.
- Overcommit memory using the **SET SRM STORBUF** command.
- Grant access to the VSWITCH for each Linux user.
- Start user IDs that should be started using the **XAUTOLOG** command.
- Limit minidisk cache in main storage and turn it off in expanded storage.

1. **Logoff** from LNXMAINT and **logon** to AUTOLOG1. At the VM READ prompt you have usually been pressing Enter, which causes the PROFILE EXEC to be run. If you do not want this EXEC to run, enter the command **ACCESS (NOPROF**:

```
LOGON AUTOLOG1
z/VM Version 5 Release 4.0, Service Level 0801 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 10:06:05 EDT THURSDAY 05/15/08
DMSIND2015W Unable to access the Y-disk. Filemode Y (19E) not accessed
z/VM V5.4.0    2008-05-13 14:27
==> acc (noprof
```

2. Make a copy of the working PROFILE EXEC:

```
==> copy profile exec a = execwrks =
```

3. Edit the file and add the bolded text. A **LOGOFF** command is added at the end of the EXEC, so the virtual machine will be logged off when it is complete. This will save a small amount of memory on the system, but does add the requirement that you type **acc (noprof** at the VM READ prompt when you log on interactively.

```
==> x profile exec
/***************************/
/*  Autolog1 Profile Exec  */
/***************************/
'cp xautolog tcpip'                /* start up TCPIP */
'CP XAUTOLOG VMSERVS'
'CP XAUTOLOG VMSERVU'
'CP XAUTOLOG VMSERVR'
```

```
'CP XAUTOLOG DTCVSW1'
'CP XAUTOLOG DTCVSW2'
'cp set pf12 ret'                   /* set the retrieve key */
'cp set mdc stor 0m 128m'           /* Limit minidisk cache in CSTOR */
'cp set mdc xstore 0m 0m'           /* Disable minidisk cache in XSTOR */
'cp set srm storbuf 300% 250% 200%' /* Overcommit memory */
'cp set signal shutdown 300'        /* Allow guests 5 min to shut down */
'cp logoff'                         /* logoff when done */
```

Save your changes with the **FILE** subcommand.

> **Important:** The **set mdc** and **set srm** lines are z/VM tuning values. It is believed that these are useful starts for Linux systems, but will not be optimal for all z/VM systems.
>
> For more information about these values, see the following Web sites:
>
> http://www.vm.ibm.com/perf/tips/linuxper.html
> http://www.vm.ibm.com/perf/tips/prgmdcar.html

You may choose to modify or omit some of these settings. Your system should now be configured to start up and send a signal to shut down Linux user IDs.

### 4.8.2 Testing the changes

To test your changes, you must reIPL z/VM again. Perform the following steps:

1. Shut down and reIPL your system.

```
==> shutdown reipl iplparms cons=sysc
SYSTEM SHUTDOWN STARTED
```

2. When your system comes back, logon as MAINT.
3. Query the SRM values to verify that the new STORBUF settings are in effect and the SIGNAL SHUTDOWN value is set to 300 seconds:

```
==> q srm
IABIAS : INTENSITY=90%; DURATION=2
LDUBUF : Q1=100% Q2=75% Q3=60%
STORBUF: Q1=300% Q2=250% Q3=200%
DSPBUF : Q1=32767 Q2=32767 Q3=32767
...
==> q signal shutdown
System default shutdown signal timeout: 300 seconds
```

This output shows that your changes have taken effect.

## 4.9 Addressing z/VM security issues

This section briefly discusses the following security issues:

- z/VM security products
- High level z/VM security
- Linux user ID privilege classes
- z/VM user ID and minidisk passwords

## 4.9.1 VM security products

You might want to use a z/VM security product such as IBM RACF® or CA VM:Secure. They allow you to address more security issues such as password aging and the auditing of user access attempts.

## 4.9.2 High level z/VM security

The paper *z/VM Security and Integrity* discusses the isolation and integrity of virtual servers under z/VM. It is available on the Web at:

http://www-1.ibm.com/servers/eserver/zseries/library/techpapers/pdf/gm130145.pdf

### Linux user ID privilege classes

Another security issue is the privilege class that Linux user IDs are assigned. The IBM Redpaper publication *Running Linux Guests with less than CP Class G Privilege* addresses this issue. It is on the Web at:

http://www.redbooks.ibm.com/redpapers/pdfs/redp3870.pdf

### z/VM user ID and minidisk passwords

All passwords in a “vanilla” z/VM system are the same as the user ID. This is a large security hole. The *minimum* you should do is to address this issue.

There are two types of passwords in the `USER DIRECT` file:

**User IDs** The password required to logon with

**Minidisks** Separate passwords for read access, write access and multi-write access

Both types of passwords should be modified. This can be done using the `CHPW540 XEDIT` macro defined in the next section.

## 4.9.3 Changing passwords in USER DIRECT

Changing the passwords can be done manually in **`XEDIT`**. However, this is both tedious and error-prone. So a profile named `CHPW540 XEDIT` has been included with this book. The source code is in “The CHPW540 XEDIT macro” on page 220.

This macro will change all z/VM passwords to the same value, which may still not be adequate security, given the different function of the various user IDs. If you want different passwords, you have to modify the `USER DIRECT` file manually, either with or without using the `CHPW52 XEDIT` macro.

To modify all user ID and minidisk passwords to the same value, perform the following steps.

1. **`Logon`** to `MAINT`.
2. Link and access the `LNXMAINT 192` disk to pick up the **`CHPW540 XEDIT`** macro:

```
==> vmlink lnxmaint 192
DMSVML2060I LNXMAINT 192 linked as 0120 file mode Z
```

3. Make a backup copy of the `USER DIRECT` file and first be sure the password that you want to use is not a string in the file. For example, if you want to change all passwords to `lnx4vm`, then do the following:

```
==> copy user direct c = direwrks = (oldd
==> x user direct c
====> /lnx4vm
```

```
DMSXDC546E Target not found
====> quit
```

The `Target not found` message shows that the string `lnx4vm` is not used in the `USER DIRECT` file, so it is a useful candidate for a password.

4. Edit the `USER DIRECT` file with a parameter of **`(profile chpw540)`** followed by the new password. Rather than invoking the default profile of `PROFILE XEDIT`, this command will invoke the **XEDIT** macro named `CHPW540 XEDIT` and pass it the new password. For example, to change all passwords to `lnx4vm`, enter the following command:

   ```
   ==> x user direct c (profile chpw540) lnx4vm

   Changing all passwords to: LNX4VM

   DMSXCG517I 1 occurrence(s) changed on 1 line(s)
   DMSXCG517I 1 occurrence(s) changed on 1 line(s)
   DMSXCG517I 1 occurrence(s) changed on 1 line(s)
   ...
   ```

5. When the profile finishes you are left in the **XEDIT** session with all passwords modified. You may wish to first examine the changes. Then save the changes with the **FILE** subcommand:

   ```
   ====> file
   ```

6. Bring the changes online with the **DIRECTXA** command:

   ```
   ==> directxa user
   z/VM USER DIRECTORY CREATION PROGRAM - VERSION 5 RELEASE 4.0
   EOJ DIRECTORY UPDATED AND ON LINE
   HCPDIR494I User directory occupies 43 disk pages
   ```

Your new directory is online. Remember the new password.

Note that this **XEDIT** macro will only work on a vanilla `USER DIRECT` file because it searches for the original user IDs next to passwords. If you want to change your password again, it should be much easier because you can use the **XEDIT CHANGE** subcommand. For example, to change all passwords from `lnx4vm` to `vm53nx`, invoke the following commands:

```
==> x user direct c
====> c/LNX4VM/VM4LNX/* *
DMSXCG517I 798 occurrence(s) changed on 345 line(s)
```

At this point, your z/VM system is now customized and ready for Linux. We recommend that you back up your system to tape.

## 4.10 Backing up your z/VM system to tape

Your system is now customized with a running TCP/IP stack, a highly available VSWITCH, a startup and shutdown process, and a user ID for shared files. You have changed the passwords. This would be an appropriate time to back up the system to tape.

There are five system volumes that should be backed up: `540RES`, `540SPL`, `540PAG`, `540W01` and `540W02` (or just the first three, if you are using 3390-9s). You also have configured a sixth volume that is important to Linux; that is, the first 320 cylinders of the volume with `LNXMAINT` on it.

To learn how to back up these volumes to tape, refer to Step 11, Store a backup copy of the z/VM system on tape, which you can find in Chapter 8 “Load the system image” in *z/VM Guide for Automated Installation and Service*, GC204-6099.

## 4.11 Relabeling the system volumes

Relabeling the system volumes is optional, but recommended. There are times when you will want to change the volume labels of the five z/VM system volumes (or three, if you installed onto 3390-3s). If there is a possibility that another vanilla z/VM system with the same labels is installed onto volumes accessible by your z/VM system, then one of the systems will not IPL correctly.

When installing z/VM, it is possible to modify all but one volume label, that of the `540RES` volume. This alleviates the problem that is described next, but it does not alleviate the problem of duplicate volume names.

To understand this possibility, refer to Figure 4-13. The z/VM system with the lower device addresses starting at `E340` should IPL without any problem (although you may see a warning at system startup time about duplicate volume labels). However, if the z/VM system starting at device address `F000` is IPLed, the `540RES` volume will be used, but the remaining volumes in the system are searched for by volume label, not by device address. Because the z/VM system 1 addresses are lower than the z/VM system 2 addresses, system 2 will be using the system 1 volumes, which is undesirable for either system.

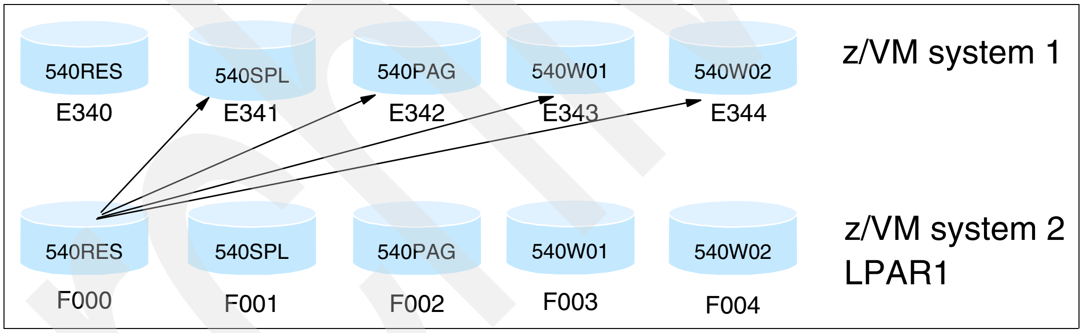


*Figure 4-13 The problem with two z/VM systems with identical volume labels*

If there is a possibility of another z/VM system being installed on DASD that this system will have access to, it is recommended that you perform the following steps. You will need access to the HMC to perform them:

- Modifying labels in the SYSTEM CONFIG file
- Modifying labels in the USER DIRECT file
- Changing the labels on the five volumes
- Shutting down your system and restarting it

**Important:** This process *must* be performed as documented, because making a mistake in one of the steps can easily produce an unusable system. Proceed carefully and your system will come back with no problems. Try to do all steps in succession in a short amount of time.

### 4.11.1 Modifying labels in the SYSTEM CONFIG file

An HMC 3270 session is needed because z/VM will have to be restarted with a `FORCE` option.

1. Start an Integrated 3270 Console session on the HMC from the CPC Recovery (or just Recovery) menu.
2. If you have not already done so, **logon** to `MAINT` and link and access the `LNXMAINT 192` disk to pick up the **LABEL540 EXEC** and **XEDIT** macro:

```
==> vmlink lnxmaint 192
DMSVML2060I LNXMAINT 192 linked as 0120 file mode Z
```

3. Note the first five CP-owned volumes using the **QUERY CPOWNED** command:

```
==> q cpowned
   1  540RES  D850  Own    Online and attached
   2  540SPL  D851  Own    Online and attached
   3  540PAG  D852  Own    Online and attached
   4  540W01  D853  Own    Online and attached
   5  540W02  D854  Own    Online and attached
   6  MPD855  D855  Own    Online and attached
   7  MPD950  D950  Own    Online and attached
   8  MPD951  D951  Own    Online and attached
   9  MPDA50  DA50  Own    Online and attached
  10  MPDB50  DB50  Own    Online and attached
  11  ------  ----  -----  Reserved
  12  ------  ----  -----  Reserved
  ...
```

**For 3390-9s:** If z/VM is installed onto 3390-9s, there should only be three system volumes:

```
==> q cpowned
Slot  Vol-ID  Rdev  Type   Status
   1  540RES  9300  Own    Online and attached
   2  540SPL  9301  Own    Online and attached
   3  540PAG  9302  Own    Online and attached
   ...
```

4. The labeling convention described in 2.2.1, “Volume labeling convention” on page 9 suggests using M in the second character of the label. An **XEDIT** macro, `LABEL540 XEDIT`, is supplied to help make this process more reliable. It can be used on both the `SYSTEM CONFIG` and `USER DIRECT` files.

   To modify the labels in the `SYSTEM CONFIG` file, release the A CP-disk and access it read-write. Back up the `SYSTEM CONFIG` file:

```
==> cprel a
CPRELEASE request for disk A scheduled.
HCPZAC6730I CPRELEASE request for disk A completed.
==> link * cf1 cf1 mr
==> acc cf1 f
==> copy system config f = confwrks = (oldd rep
```

5. Edit the `SYSTEM CONFIG` file with the **LABEL540 XEDIT** macro passing the five addresses of the z/VM system volumes (`d850-d854`, in this example):

```
==> x system config f (profile label540) <d850 d851 d852 d853 d854>
DMSXCG517I 3 occurrence(s) changed on 3 line(s)
DMSXCG517I 1 occurrence(s) changed on 1 line(s)
DMSXCG517I 1 occurrence(s) changed on 1 line(s)
```

```
DMSXCG517I 1 occurrence(s) changed on 1 line(s)
DMSXCG517I 1 occurrence(s) changed on 1 line(s)
```

**For 3390-9s:** If z/VM is installed onto 3390-9s, there is a macro named **LBL540-9 XEDIT** that takes only three parameters. For example, if the system volumes are 9300-9302, the command to edit the SYSTEM CONFIG file would be as follows:

```
==> x system config f (profile lbl540-9) <9300 9301 9302>
DMSXCG517I 3 occurrence(s) changed on 3 line(s)
DMSXCG517I 1 occurrence(s) changed on 1 line(s)
DMSXCG517I 1 occurrence(s) changed on 1 line(s)
```

6. Clear the window and you will be left in XEDIT, editing the file. Search for the string cp_owned and you should see the new labels. Be sure they are correct before saving the file with the **FILE** subcommand:

```
====> /cp_owned
/*                    CP_Owned Volume Statements                    */
/********************************************************************/

 CP_Owned   Slot   1  MVD850
 CP_Owned   Slot   2  MVD851
 CP_Owned   Slot   3  MVD852
 CP_Owned   Slot   4  MVD853
 CP_Owned   Slot   5  MVD854
 CP_Owned   Slot   6  MPD855
 CP_Owned   Slot   7  MPD950
 CP_Owned   Slot   8  MPD951
 CP_Owned   Slot   9  MPDA50
 CP_Owned   Slot   10 MPDB50
 CP_Owned   Slot   11  RESERVED
 CP_Owned   Slot   12  RESERVED
    ...
====> file
```

7. Verify that there are no syntax errors:

```
==> acc 193 g
==> cpsyntax system config f
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
```

8. Release and detach the F disk, **CPACCESS** the A disk, and verify:

```
==> rel f (det
DASD 0CF1 DETACHED
==> cpacc * cf1 a
CPACCESS request for mode A scheduled.
Ready; T=0.01/0.01 09:19:57
HCPZAC6732I CPACCESS request for MAINT's 0CF1 in mode A completed.
==> q cpdisk
Label  Userid   Vdev Mode Stat Vol-ID Rdev Type   StartLoc     EndLoc
MNTCF1 MAINT    0CF1  A   R/O  540RES D850 CKD          39        158
MNTCF2 MAINT    0CF2  B   R/O  540RES D850 CKD         159        278
MNTCF3 MAINT    0CF3  C   R/O  540RES D850 CKD         279        398
```

You have now changed the labels of the system volumes in the SYSTEM CONFIG file.

**Important:** It is critical to continue this process now, because your system is currently in a state where it will not IPL cleanly.

### 4.11.2 Modifying labels in the USER DIRECT file

In this section you will modify the system volume labels in the USER DIRECT file.

1. Modify the USER DIRECT file again using the **LABEL540 XEDIT** macro. You should see many more occurrences of the labels being changed:

```
==> copy user direct c = direwrks = (oldd rep
==> x user direct c (profile label540) <d850 d851 d852 d853 d854>
DMSXCG517I 99 occurrence(s) changed on 99 line(s)
DMSXCG517I 68 occurrence(s) changed on 68 line(s)
DMSXCG517I 124 occurrence(s) changed on 124 line(s)
DMSXCG517I 2 occurrence(s) changed on 2 line(s)
DMSXCG517I 1 occurrence(s) changed on 1 line(s)
```

**For 3390-9s:** Again, the macro named **LBL540-9 XEDIT** should be used for systems installed onto 3390-9s. For example, if the system volumes are 9300-9302, the command to edit the SYSTEM CONFIG file would be as follows:

```
   ==> x user direct c (profile lbl520-9) <9300 9301 9302>
   DMSXCG517I 287 occurrence(s) changed on 287 line(s)
   DMSXCG517I 2 occurrence(s) changed on 2 line(s)

DMSXCG517I 1 occurrence(s) changed on 1 line(s)
```

You may choose to traverse the file before saving the changes with the **FILE** subcommand:

```
====> file
```

You have now changed the labels of the system volumes in the USER DIRECT and SYSTEM CONFIG files. Again, it is critical that you proceed with the remaining steps.

### 4.11.3 Changing the labels on the five volumes

In this section you will change the labels on the five volumes using the **CPFMTXA** command. You could do this one volume at a time with the **CPFMTXA LABEL** command. However, the **LABEL540** EXEC has been written to make this step easier.

1. Use the **LABEL540 EXEC** with the physical addresses of the five system volumes:

```
==> label540 <d850 d851 d852 d853 d854>
The volumes are:
DASD D850 CP OWNED  540RES   59
DASD D851 CP OWNED  540SPL   1
DASD D852 CP OWNED  540PAG   0
DASD D853 CP OWNED  540W01   61
DASD D854 CP OWNED  540W02   25

The system volume labels will become:
MVD850 MVD851 MVD852 MVD853 MVD854

ARE YOU SURE you want to relabel the DASD (y/n)?
y
HCPCCF6209I INVOKING ICKDSF.
...
```

**For 3390-9s:** The `LBL540-9 EXEC` should be used for systems installed onto 3390-9s. For example, if the system volumes are 9300-9302, the command to relabel the system volumes would be as follows:

```
==> lbl540-9 <9300 9301 9302>
The volumes are:
DASD 9300 CP OWNED  540RES   142
DASD 9301 CP OWNED  540SPL   2
DASD 9302 CP OWNED  540PAG   0

The system volume labels will become:
MV9300 MV9301 MV9302

ARE YOU SURE you want to relabel the DASD (y/n)?
y
...
```

2. Now that the five volumes have been relabeled (this is sometimes called “clipping the volumes”, which is derived from a contraction (clp) of the z/OS term change label program), you can run the **`DIRECTXA`** command to update the directory:

   ```
   ==> directxa user
   z/VM USER DIRECTORY CREATION PROGRAM - VERSION 5 RELEASE 4.0
   EOJ DIRECTORY UPDATED
   HCPDIR494I User directory occupies 43 disk pages
   Ready(00005); T=0.01/0.01 15:45:51
   ```

   A return code of 5 is expected because the labels in the `USER DIRECT` file are different from the spool data in the currently running system.

Finally, you are ready to issue a **`SHUTDOWN`** command.

## 4.11.4 Shutting down your system and restarting it

You will need an HMC console session for this step, if you are not already running from there. To test the changes, you must shut your system down and then restart it. You cannot do a `SHUTDOWN REIPL` in this situation because you will have to do a **`FORCE`** start

```
==> shutdown
SYSTEM SHUTDOWN STARTED
HCPSHU960I System shutdown may be delayed for up to 210 seconds
```

Perform the following steps to bring the system back up:

1. Open an HMC session.
2. Select your LPAR.
3. Use the circular arrow racetrack buttons to get to the CPC Recovery (or just Recovery) menu.
4. Double-click the **Integrated 3270 Console** menu item. A new window should appear.
5. Double-click the **LOAD** menu item. The Load Address (`D850`, in this example) and Load Parameter (`SYSG`) fields should be correct from the previous IPL.
6. Select the **Clear** radio button. The Load Address and Load Parameter fields should be correct from the previous IPL. Click **OK**.
7. Click **Yes** on the Load Task Confirmation window.

8. Go back to the Integrated 3270 console. After a few minutes, the Standalone Program Loader window should appear. Use the **Tab** key to traverse to the section IPL Parameters and enter the value **cons=sysg**.
9. Press the **F10** key to continue the IPL of your z/VM system. This should take 1 to 3 minutes.
10. At the Start prompt you have to specify a **FORCE** start, again because the spool volume label has changed:

    ```
    ==> force drain
    ```

11. Do not change the time-of-day clock:

    ```
    ==> no
    ```

12. When the IPL completes, **DISCONNECT** from the OPERATOR user ID:

    ```
    ==> disc
    ```

13. Close the HMC windows.
14. Start a 3270 emulator session as the TCPIP service machine should be up. Logon as MAINT.

    Get a 3270 session as MAINT and verify the volume labels have changed with the **QUERY CPOWNED** command:

    ```
    ==> q cpowned
    Slot  Vol-ID  Rdev  Type   Status
       1  MVD850  D850  Own    Online and attached
       2  MVD851  D851  Own    Online and attached
       3  MVD852  D852  Own    Online and attached
       4  MVD853  D853  Own    Online and attached
       5  MVD854  D854  Own    Online and attached
       6  MPD855  D855  Own    Online and attached
      ...
    ```

**Important:** In the event that you IPLed a system with duplicate system volumes, it is possible that you may have destroyed your saved segments. You will know this is the case when you cannot **IPL CMS**. Rather, you will have to **IPL 190**.

To rebuild saved segments, use the following commands (only do so if your saved segments have been destroyed):

```
==> vmfsetup zvm cms
==> sampnss cms
==> ipl 190 clear parm nosprof instseg no
==> acc (noprof
==> acc 5e6 b
==> acc 51d d
==> vmfbld ppf segbld esasegs segblist ( all
```

## 4.12 Restoring your z/VM system from tape

It is a good idea to *practice* restoring a system before you need to do it under pressure in a real production environment.

Restoring a z/VM system from tape that has the same set of volume labels as the system that is running is problematic. If there are two z/VM systems on the same LPAR with the same volume labels, both systems cannot be IPLed cleanly.

Perform this step only if you successfully completed 4.10, “Backing up your z/VM system to tape” on page 62, and 4.11, “Relabeling the system volumes” on page 63. If you have done both, then the system on tape has volume labels of `540`*xxx* and the system on DASD has volume labels `MV`*yyyy*.

You can restore this system to five other 3390-3s, as explained in Appendix E, “Restore the z/VM system backup copy from tape” in *z/VM Guide for Automated Installation and Service*, GC204-6099.

# 5

# Servicing z/VM

This chapter describes how to apply:

- A Programming Temporary Fix (PTF)
- A Recommended Service Upgrade (RSU) from “envelope files”

Both processes are basically the same.

Note that this chapter is based on z/VM 5.2, and not on 5.4, due to the timing of the writing of this book. However, the process has not changed significantly, so it should be just as useful. Keep in mind, however, that your output messages and so on will be slightly different.

**Important:** When applying service, there is always a chance that you may want to back it out. We recommend that you have an up-to-date backup of your system available before starting this section.

The application of corrective service to z/VM is covered in *z/VM V5R3 Service Guide* and in *VMSES/E Introduction and Reference*. Both of these documents can be downloaded in PDF format from the following URL:

`http://www.vm.ibm.com/library`

VMSES/E is a component of z/VM that provides the **`SERVICE`** and **`PUT2PROD`** EXECs. The **`SERVICE`** EXEC performs the following tasks:

- It installs an RSU or applies CORrective service for z/VM components, features, or products.
- It displays either the RSU level of the component specified or whether a particular PTF or APAR has been applied (when used with STATUS).
- It creates PTF bitmap files (when used with BITMAP).

When **`SERVICE`** is successfully completed, the **`PUT2PROD`** EXEC places the z/VM components, features, or products that are installed on the z/VM System DDR into production.

## 5.1 Applying a PTF

You may determine that you need to apply a specific fix or PTF to your system.

For example, an Authorized Program Analysis Report (APAR), VM63895, was opened to address the problems reported with virtual NIC support. There are three known symptoms addressed by this APAR:

- Linux guests may lose connectivity after **shutdown -r now** (or any device reset).
- Using an External Security Manager (ESM) to authorize a VLAN list may lead to an FRF002 abend.
- Virtual Hipersockets NIC configured with "VLAN nnn" (exploiting Set Global VLAN ID) did not really filter inbound frames.

The APAR was assigned the following PTF numbers:

- z/VM 5.1.0 VM63895 UM31612
- z/VM 5.2.0 VM63895 UM31613

There are more details on:

http://www-1.ibm.com/support/docview.wss?uid=isg1VM63895

So, for z/VM 5.2, apply PTF UM31613 as described in the next section.

### 5.1.1 Determining whether a PTF has been applied

Check to make sure the PTF has not previously been applied.

1. **Logon** to MAINT and issue the **VMFSETUP** command to set up minidisks for TCP/IP and link to them:

```
==> vmfsetup zvm cp (link
VMFSET2760I VMFSETUP processing started for ZVM CP
VMFUTL2205I Minidisk|Directory Assignments:
            String    Mode  Stat  Vdev  Label/Directory
VMFUTL2205I LOCALMOD  E      R/W  2C4   MNT2C4
VMFUTL2205I LOCALSAM  F      R/W  2C2   MNT2C2
...
VMFSET2760I VMFSETUP processing completed successfully
```

2. Use the **VMFINFO** command to query the Software Inventory files. Move the Tab key to ZVM and type s to select it on the PPF Fileid window:

```
==> vmfinfo

PPF Fileid - Help

  Product parameter files (PPFs) define the environment and key variables
  required to process the queries.  The following is a list of all PPFs
  found on all accessed disks.  Select one to continue.  The View function
  can be used to examine one or more PPFs.

  Type a "V" next to one or more PPFs to view their contents, or type an
  "S" next to one PPF to select.

       Options: S - select  V - view

  Option   PPF Fileid
    _      $5654260 PPF      D1
```

```
_       SEGBLD   PPF      D2
_       SERVP2P  PPF      D1
_       UCENG    PPF      D2
s       ZVM      PPF      D2
_       4OSASF40 PPF      D1
```

3. Because the description of the PTF cites a component name of “VM CP”, select **CP** on the Component Name window.
4. Select **PTFs/APARs** on the VMFINFO Main window.
5. Type in the PTF number `UM31613` in the PTF number field, then select **Status of PTF** on the PTF/APAR Queries window:

```
PTF/APAR Queries

 Enter a PTF or APAR number and type an option code.  Then press Enter.
  PPF fileid ...... ZVM       PPF  D
  Component name .. CP                Setup ... NO
  Product ID .....: 5VMCPR20          System .. VM
  PTF number ...... UM31613
  APAR number .....

     Options: S - select
  Option   Query
    s      Status of PTF
    _      Requisites/supersedes of PTF
    _      Dependencies/superseding of PTF
    _      User memo of PTF
    _      Serviceable parts included by PTF

    _      Abstract of APAR(s)
```

6. If the PTF has not been successfully applied, you should see the message `No data found`:

```
Query Output - PTF Status

PPF fileid .....: ZVM       PPF  D
  Component name .: CP                Setup ..: NO
  Product ID .....: 5VMCPR10          System .: VM
 ---------------------------------------------------
WN:VMFSIP2481W No entries match search arguments
WN:            TDATA :PTF UM31198
WN:            in table 5VMCPR10 SRVRECS J

No data found
```

This shows that PTF UM31613 has *not* been successfully applied. The sections that follow describe how to obtain and apply it.

## 5.1.2 Getting service using Internet FTP

You may obtain service for z/VM using tapes. However, you may also want to obtain service over the Internet. If so, point a Web browser to:

https://techsupport.services.ibm.com/server/login

If you have an IBM user ID and password, use that. If you do not, you can fill out the form to create an IBM ID and password. You should then be at the following Web site:

https://www.ibm.com/account/profile/us

1. Click **Support and Downloads** at the top menu.
2. Click **Downloads and Drivers** on the left frame.
3. Under Category, select **zSeries (mainframe)**
4. Under Operating Systems, select **z/VM** and click **Software Only**. This should take you to a page entitled Support for VM.
5. Click on **Download selective fixes by PTF**. You may be prompted for your IBM ID and password.
6. In the text box Enter PTF numbers below [e.g: U412345, U467890], enter `UM31613`. All other defaults should be correct. Click **Continue**.
7. In the Verify Order page, click **Submit**. You should get a message similar to the following

```
Your order has been submitted for processing. Email will be sent to nospam@us.ibm.com.

COER NUMBER is <390473266>. This number is used to submit your request. You will receive
a confirmation email that contains your ORDER NUMBER.
...
```

### 5.1.3 Downloading the service to z/VM

You should receive two e-mails. The first e-mail contains your order number. The second e-mail contains instructions about how to download the service files. Make sure you have access to these. Following is an example.

```
TEXT    = Data sent via "INET". To retrieve your service:
TEXT    =   FTP to: ptf.boulder.ibm.com
TEXT    =   Log on using userid "owte8a" and password "h2q9nep9"
TEXT    =   Enter the following FTP commands:
TEXT    =     cd /390268476/c568411202
TEXT    =     ascii
TEXT    =     get ftp8476.txt
TEXT    =     binary f 1024
TEXT    =     get rlst1585.bin
TEXT    =     get rptf1585.bin
```

1. `Logon` to MAINT.
2. The `MAINT 500` disk should have a large amount of free space, so it is a useful minidisk on which to download the files. By default, the FTP client saves files on the A disk, so access the 500 disk as A:

```
==> vmlink tcpmaint 592
DMSVML2060I TCPMAINT 592 linked as 0120 file mode Z
==> acc 500 a
DMSACC724I 500 replaces A (191)
```

3. Use the FTP client to get the PTF *envelope files* from the Internet. The envelope files can be large, so this may take some time. As you are downloading the files, note the file sizes. Following is an example:

```
==> ftp ptf.boulder.ibm.com
ftp> <owte8a>
ftp> <h2q9nep9>
ftp> cd </390268476/c568411202>
ftp> ascii
ftp> get ftp1585.txt
...
ftp> binary f 1024
ftp> get vlst1585.bin
...
```

```
150 Opening BINARY mode data connection for vlst1585.bin (7168 bytes).
7168 bytes transferred in 0.231 seconds. Transfer rate 31.03 Kbytes/sec.
ftp> get vptf1585.bin
...
551936 bytes transferred in 22.272 seconds. Transfer rate 24.78 Kbytes/sec.
ftp> quit
```

4. Use the **BROWSE** command to view the first text file and verify that the correct number of bytes were downloaded for each file. Press the **F3** key to quit.

```
==> browse ftp1585 txt
VM PTF Package Information
--------------------------
This file contains byte counts of files to receive and instructions
for preparing the files for installation. The byte counts listed
below should match the byte counts of the files when they are received
using FTP.

FILE BYTE COUNTS
----------------

The vptf1585.bin byte count is: 551936.
The vlst1585.bin byte count is: 7168.

Match these byte counts to that reported during the FTP get.
...
====> F3
```

5. You should now have the service or envelope files on your z/VM system. Rename the file type from `BIN` to `SERVLINK`, because this is the file type that the **SERVICE** command expects.

```
==> rename vlst1585 bin a = servlink =
==> rename vptf1585 bin a = servlink =
```

6. The envelope files arrive in a compressed format to speed downloads. To use them, they must first be uncompressed with the **DETERSE** command. Use the **(REPLACE** parameter to uncompress them in place and save disk space:

```
==> deterse vlst1585 servlink a = = = (replace
==> deterse vptf1585 servlink a = = = (replace
```

### 5.1.4 Receiving, applying, and building service

You must receive, apply, and build the PTF. Then it can be put into production. This can be achieved using a process that is much easier now with the **SERVICE** command.

To prepare to use the **SERVICE** command, you must have a 256 MB virtual machine and you must have the minidisk with a large amount of free space; that is what the `MAINT 500` minidisk is for.

1. Increase the size of the `MAINT` virtual machine with the **DEFINE STORAGE** command:

```
==> def stor 256M
STORAGE = 256M
Storage cleared - system reset.
```

2. ReIPL CMS:

```
==> ipl cms
IPL CMS
z/VM V5.4.0    2008-05-13 14:27
==> Press Enter
```

3. The **SERVICE** command will write to the current A disk. Again access minidisk 500 as A:

```
==> acc 500 a
DMSACC724I 500 replaces A (191)
```

4. Use the **SERVICE ALL** command, specifying the envelope files you downloaded. Many, many windows of output will scroll by and the windows will automatically be cleared. Important messages will be saved to the A (500) disk. This process may take many minutes. Following is an example:

```
   ==> service all vptf1585
   ...
   VMFSUT2760I VMFSUFTB processing started
   VMFSUT2760I VMFSUFTB processing completed successfully
VMFSRV2760I SERVICE processing completed successfully
```

A return code of zero (0) is ideal. In general a return code of 4 is acceptable. That means that only warnings were issued. A return code of 8 or greater generally means that errors were encountered.

5. The output files written to the A disk are of the form $VMF* $MSGNUM. You may wish to inspect these files.

```
==> filel $VMF* $MSGLOG
MAINT    FILELIST A0  V 169  Trunc=169 Size=5 Line=1 Col=1 Alt=0
Cmd   Filename Filetype Fm Format Lrecl    Records     Blocks   Date     Time
      $VMFSRV  $MSGLOG  A1 V         80        132          3  1/31/06 12:58:09
      $VMFBLD  $MSGLOG  A1 V         80         76          2  1/31/06 12:57:34
      $VMFAPP  $MSGLOG  A1 V         80         70          1  1/31/06 12:57:13
      $VMFREC  $MSGLOG  A1 V         80         55          1  1/31/06 12:57:12
      $VMFMRD  $MSGLOG  A1 V         80         30          1  1/31/06 12:57:10
```

6. Invoke the **VMFVIEW SERVICE** command to review the results of the previous **SERVICE** command. Press the **F3** key to quit. Following is an example:

```
==> vmfview service
************************************************************************
****             SERVICE              USERID: MAINT                ****
************************************************************************
****            Date: 11/31/07 Time: 12:57:09           ****
************************************************************************
====> F3
```

Ideally there will be no output (as in this example); that means the service applied perfectly.

### 5.1.5 Putting the service into production

Use the **PUT2PROD** command to put the service into production.

> **Important:** If you run **PUT2PROD** from a 3270 emulator session, you may lose your connection because the TCP/IP service machine may be recycled. Therefore, you may want to run this command from a console.
>
> In this example, applying PTF UM31613 did *not* affect the emulator session.

```
==> put2prod
RDR FILE 0016 SENT FROM MAINT    CON WAS 0016 RECS 0004 CPY  001 T NOHOLD NOKEEP
VMFP2P2760I PUT2PROD processing started
VMFP2P2760I PUT2PROD processing started for VMSES
VMFSET2760I VMFSETUP processing started for SERVP2P VMSESP2P
...
```

```
USER DSC   LOGOFF AS  BLDCMS   USERS = 7     FORCED BY MAINT
VMFP2P2760I PUT2PROD processing completed successfully for SAVECMS
VMFP2P2760I PUT2PROD processing completed successfully
```

Your PTF should now be “put into production”. You may or may not have to reIPL the system, depending on the nature of the PTF applied. It is safest to reIPL using the **SHUTDOWN REIPL** command to completely test the changes:

```
==> shutdown reipl iplparms cons=sysc
SYSTEM SHUTDOWN STARTED
...
```

Your z/VM system should come back in a few minutes. You may want to perform the steps described in 5.1.1, “Determining whether a PTF has been applied” on page 72, to verify that this PTF has been applied.

## 5.2 Applying a Recommended Service Upgrade

Applying a Recommended Service Upgrade (RSU) is very similar to applying a PTF, as described in 5.1, “Applying a PTF” on page 72. z/VM service can be preventive (RSU) or corrective (COR).

Part 4, “Service Procedure”, in *Guide for Automated Installation and Service,* provides a complete description of how to apply service to z/VM; however, it assumes you are starting with the RSU tape. Following is an example of upgrading to a z/VM 5.3 RSU with the medium being files, not tape.

The following sections summarize how to apply service and also describe how to obtain service using envelope files over the Internet.

You must first determine if your system needs service. Use the **QUERY CPLEVEL** command:

```
==> q cplevel
z/VM Version 5 Release 3.0, service level 0701 (64-bit)
Generated at 11/18/07 23:44:09 EST
IPL at 11/19/07 13:44:37 EST
```

The service level four digits are split in half, where the first half is the last two digits of the year and the second half is the service level for that year. Therefore, a service level of 0701 means the first service level for the year 2007 has been applied. If you installed from DVD, that was on the second RSU DVD disc.

The overall steps in applying a service level are as follows:

- Getting service from the Internet
- Download the service to z/VM
- Receive, apply, and build the service
- Put the service into production

### 5.2.1 Getting service from the Internet

When getting service from the Internet, note the following information:

- The PTF number for the most current RSU for z/VM 5.4.0 is UM97540.
- The PTF number for the most current RSU for z/VM 5.3.0 is UM97530.
- The PTF number for the most current RSU for z/VM 5.2.0 is UM97520.

Point a Web browser to:

`https://techsupport.services.ibm.com/server/login`

1. If you have an IBM user ID and password, use that. If you do not, you can fill out the form to create an IBM ID and password. You should then be at the following Web site:

   `https://www.ibm.com/account/profile/us`

2. Click **Support and Downloads** at the top menu.
3. In the Choose support type field, choose **System z**.
4. On the Support for mainframes page, choose **z/VM** in the Hardware or operating system field.
5. Click the **Download** tab, then click **Download specific fixes** (these are ordered by PTF number). You may be prompted for your IBM ID and password.
6. In the text box Enter PTF numbers below [e.g: U412345, U467890], enter `UM97530` for the latest z/VM 5.3 service level, or enter `UM97520` for the latest z/VM 5.2 service level, or enter the appropriate PTF number. All other defaults should be correct.
7. Click **Continue**.
8. In the Verify Order page, click **Submit**. You should receive a message similar to the following:

```
Your order has been submitted for processing. Email will be sent to
nospam@us.ibm.com.

COER NUMBER is <390777166>. This number is used to submit your request. You will
receive a confirmation email that contains your ORDER NUMBER.
```

## 5.2.2 Downloading the service to z/VM

You should receive two e-mails. The first e-mail contains your order number. The second e-mail contains instructions about how to download the service files. Make sure you have access to these. Following is an example.

```
TEXT    = Data sent via "INET". To retrieve your service:
TEXT    =   FTP to: ptf.boulder.ibm.com
TEXT    =   Log on using userid "anonymous" and password "h5q7nep9"
TEXT    =   Enter the following FTP commands:
TEXT    =     cd /390777166/c568411202
TEXT    =     ascii
TEXT    =     get ftp7166.txt
TEXT    =     binary f 1024
TEXT    =     get rlst7166.bin
TEXT    =     get rptf0166.bin
TEXT    =     get rptf0266.bin
TEXT    =     get rptf0366.bin
```

1. **Logon** to `MAINT`.
2. The `MAINT 500` disk should have a large amount of free space, so it is a useful minidisk on which to download the files. By default, the FTP client saves files on the A disk, so access the 500 disk as A:

```
==> acc 500 a
DMSACC724I 500 replaces A (191)
```

3. Link to the TCPMAINT 592 disk to obtain access to the FTP client command:

```
==> vmlink tcpmaint 592
DMSVML2060I TCPMAINT 592 linked as 0120 file mode Z
```

4. Use the FTP client to obtain the RSU envelopes from the Internet. The envelope files can be large, so this may take some time. We recommend that you rename the file type from `BIN` to `SERVLINK` using FTP, because this is the file type that the **SERVICE** command expects. As you are downloading the files, note the file sizes. Following is an example.

```
==> ftp ptf.boulder.ibm.com
ftp> <owte8a>
ftp> <h2q9nep9>
ftp> cd </390777166/c568411202>
ftp> ascii
ftp> get ftp7166.txt
ftp> binary f 1024
ftp> get rlst7166.bin rlst8476.servlink
...
10240 bytes transferred in 0.523 seconds. Transfer rate 19.58 Kbytes/sec
ftp> get rptf0166.bin rptf0176.servlink
...
36944896 bytes transferred in 191.632 seconds. Transfer rate 192.79 Kbytes/sec.
ftp> get rptf0266.bin rptf0276.servlink
...
26028032 bytes transferred in 132.353 seconds. Transfer rate 196.66 Kbytes/sec.
ftp> get rptf0366.bin rptf0376.servlink
...
52193280 bytes transferred in 269.094 seconds. Transfer rate 193.96 Kbytes/sec.
ftp> quit
```

You should now have the service or envelope files on your z/VM system.

5. The envelope files arrive in a compressed format to speed downloads. To use them, they must first be uncompressed with the **DETERSE** command. Use the **(REPLACE** parameter to uncompress them in place and save disk space:

```
==> deterse rlst7166 servlink a = = = (replace
==> deterse rptf0166 servlink a = = = (replace
==> deterse rptf0266 servlink a = = = (replace
==> deterse rptf0366 servlink a = = = (replace
```

> **Note:** If you did not rename the file names during the FTP session, you can use the CMS **RENAME** command. Type **HELP RENAME** for help.

6. Use the **BROWSE** command to read the RSU information. Compare the byte count that you recorded earlier with the values in this file.

```
==> browse ftp7166 txt
VM RSU Package Information
--------------------------
This file contains byte counts of files to receive and instructions
for preparing the files for installation. The byte counts listed
below should match the byte counts of the files when they are received
using FTP.

FILE BYTE COUNTS
----------------

The rlst7166.bin byte count is: 10240.
The rptf166.bin byte count is: 36944896.
The rptf266.bin byte count is: 26028032.
The rptf366.bin byte count is: 52193280.
...
```

### 5.2.3 Receiving, applying, and building the service

You must receive, apply, and build the service. Then it can be put into production. In the past, this was a cumbersome procedure. For example, to receive, apply and build the CP component, the following steps were needed:

```
vmfmrdsk zvm cp apply (setup
vmfsetup zvm cp
vmfpsu zvm cp
vmfins install ppf zvm cp (nomemo env {filename} nolink override no
vmfapply ppf zvm cp (setup
vmfbld ppf zvm cp (status
vmfbld ppf zvm cp (serviced
```

Then the same steps were needed for many other components. The process is much easier now with the **SERVICE** command. On the other hand, the previous method is more granular and better enables the system administrator to know which pieces of service have been applied.

1. To prepare to use the **SERVICE** command, you must have a 256 MB virtual machine and you must have the minidisk with a large amount of free space; that is what the MAINT 500 minidisk is for. Increase the size of the MAINT virtual machine with the **DEFINE STORAGE** command:

   ```
   ==> def stor 256M
   STORAGE = 256M
   Storage cleared - system reset.
   ==> ipl cms
   IPL CMS
   z/VM V5.4.0    2008-05-13 14:27
   ==> Press Enter
   ```

2. The **SERVICE** command will write to the current A disk, so you again want to access 500 as A:

   ```
   ==> acc 500 a
   DMSACC724I 500 replaces A (191)
   ```

3. Use the **SERVICE ALL** command, specifying the envelope files you downloaded. Many, many windows of output will scroll by and the windows will automatically be cleared. Important messages will be saved to the A (500) disk. This process may take many minutes or tens or tens of minutes. Following is an example:

   ```
   ==> service all rptf0176 rptf0276 rptf0376
   ...
   VMFSUT2760I VMFSUFTB processing completed successfully
   VMFSRV2760I SERVICE processing completed with warnings
   Ready(00004); T=185.66/191.79 11:56:47
   ```

   A return code of zero (0) is ideal. Note in the last Ready line that this command returned a code of 4. In general a return code of 4 is acceptable. That means that only warnings were issued. A return code of 8 or greater generally means that errors were encountered.

4. The output files written to the A disk are of the form $VMF* $MSGNUM:

   ```
   ==> filel $VMF* $MSGLOG
   $VMFP2P  $MSGLOG  A1 V         80      1520         28 11/10/05 13:35:43
   $VMFBLD  $MSGLOG  A1 V         80       639          9 11/10/05 13:28:42
   $VMFMRD  $MSGLOG  A1 V         80       499          7 11/10/05 13:28:39
   $VMFSRV  $MSGLOG  A1 V         80      1369         25 11/10/05 11:56:47
   $VMFAPP  $MSGLOG  A1 V         80       682          9 11/10/05 11:54:07
   $VMFINS  $MSGLOG  A1 V         80       381          6 11/10/05 11:54:05
   ```

5. Invoke the **VMFVIEW SERVICE** command to review the results of the previous **SERVICE** command. Following is an example:

```
==> vmfview service
************************************************************************
****              SERVICE              USERID: MAINT               ****
************************************************************************
****             Date: 11/10/05           Time: 11:43:15            ****
************************************************************************
CK:VMFSUI2104I PTF UM30896 contains user information. Review the :UMEMO
CK:            section in file UM30896 $PTFPART
CK:VMFSUI2104I PTF UM31044 contains user information. Review the :UMEMO
CK:            section in file UM31044 $PTFPART
CK:VMFSUI2104I PTF UM31233 contains user information. Review the :UMEMO
CK:            section in file UM31233 $PTFPART
CK:VMFSUI2104I PTF UM31275 contains user information. Review the :UMEMO
CK:            section in file UM31275 $PTFPART
WN:VMFBDC2250W The following VMHCD objects have been built on BUILD0 300
WN:            (I) and should be copied to your workstation:
WN:VMFBDC2250W EEQINSTX EXEBIN
WN:VMFSRV1221W The CP Stand-Alone Dump Utility must be rebuilt. Follow
WN:            the instructions in the z/VM Service Guide.
```

   Ideally there will be no output, which means the service applied perfectly. In this example, messages were generated. The first four `VMFSUI2104I` messages are informational. The `VMFBDC2250W` message is pertinent if you are using the VM HCD tool. The `VMFSRV1221W` is pertinent if you are using the CP Stand-alone Dump Utility.

You should see that the service was installed successfully.

### 5.2.4 Putting the service into production

This section describes how to use the **PUT2PROD** command to put the service into production.

**Important:** If you run **PUT2PROD** from a 3270 emulator session, you may lose your connection because the TCP/IP service machine may be recycled. Therefore, you may want to run this command from a console. If you do switch to a new session, remember to set the machine size to 256 M and access the 500 disk as A.

1. Use the **PUT2PROD** command to put the service into production. Many windows will scroll by. This command can take quite a number of minutes to complete:

```
==> put2prod
RDR FILE 0016 SENT FROM MAINT    CON WAS 0016 RECS 0004 CPY  001 T NOHOLD NOKEEP
VMFP2P2760I PUT2PROD processing started
...
USER DSC   LOGOFF AS  BLDCMS   USERS = 7     FORCED BY MAINT
VMFP2P2760I PUT2PROD processing completed successfully for SAVECMS
VMFP2P2760I PUT2PROD processing completed successfully
```

Even though the service has been “put into production”, the **QUERY CPLEVEL** command should still return the current service level; in this example, 0501. This is because the new CP load module (nucleus) has not been invoked:

```
==> q cplevel
z/VM Version 5 Release 3.0, service level 0701 (64-bit)
Generated at 05/28/07 23:44:09 EST
IPL at 11/11/07 13:44:37 EST
```

To invoke the new CP load module, use the **SHUTDOWN REIPL** command. When your system comes back up, it should be at the new CP service level; in this example, 0702:

```
==> shutdown reipl iplparms cons=sysc
...
==> q cplevel
z/VM Version 5 Release 3.0, service level 0702 (64-bit)
Generated at 05/29/07 18:39:52 EST
IPL at 11/12/07 14:53:28 EST
```

This shows that the new CP load module is now being used.

## 5.3 Determining the z/VM service level

Often you will want to be able to query more than just the service level. The following steps were taken from the links CP Maintenance Levels and Virtual Switch TCP/IP Maintenance Levels starting at the Web site:

http://www.vm.ibm.com/virtualnetwork/

Logon as TCPMAINT. Use the **QUERY VMLAN** command to determine the latest APAR applied:

```
==> cp query vmlan
VMLAN maintenance level:
  Latest Service: VM63895
VMLAN MAC address assignment:
  MACADDR Prefix: 020001
  MACIDRANGE SYSTEM: 000001-FFFFFF
             USER:   000000-000000
VMLAN default accounting status:
  SYSTEM Accounting: OFF       USER Accounting: OFF
VMLAN general activity:
  PERSISTENT Limit: INFINITE   Current: 1
  TRANSIENT  Limit: INFINITE   Current: 0
```

This shows that the latest APAR applied is VM63895.

The maintenance level of the TCP/IP stack is important to virtual networking. To determine this, first get the active VSWITCH controller:

```
==> q vswitch
VSWITCH SYSTEM VSW1     Type: VSWITCH Connected: 8    Maxconn: INFINITE
  PERSISTENT  RESTRICTED    NONROUTER                 Accounting: OFF
  VLAN Unaware
  State: Ready
  IPTimeout: 5         QueueStorage: 8
  Portname: UNASSIGNED RDEV: 3004 Controller: DTCVSW1  VDEV:  3004
  Portname: UNASSIGNED RDEV: 3008 Controller: DTCVSW2  VDEV:  3008 BACKUP
```

This shows the controller is named `DTCVSW1`. Then use the **NETSTAT** command with the controller name to determine the maintenance of the TCPIP MODULE:

```
==> netstat tcp <dtcvsw1> level
VM TCP/IP Netstat Level 530
IBM 2094; z/VM Version 5 Release 3.0, service level 0701 (64-bit), VM TCP/IP Lev
el 530; RSU 0701 running TCPIP MODULE E2 dated 11/10/07 at 11:34
TCP/IP Module Load Address:  00C44000
```

This shows information about the TCPIP MODULE. Use the **TCPSLVL** command and the complete file specification (TCPIP MODULE E, in this example) to get more information. Of particular interest is the latest APAR applied to TCTOOSD:

```
==> tcpslvl tcpip module e
SLVL  TCPIP     PQ22678
...
SLVL  TCTOOSD   PK00905
...
```

## Summary

At this point, you have completed installing, configuring, and servicing z/VM. A valuable attribute of z/VM is that it normally runs with little maintenance required. It is now time to change your focus to Linux.

6

# Configuring a Network File System server for RHEL 5.2

A common method of installing Linux on z/VM is over the network from another server using the Network File System (NFS). To accomplish this, we recommend using a PC Linux system. This server supplies both the RHEL 5.2 distribution and the files associated with this book.

The server must have at least 4 GB of free disk space. It can be a Linux PC, but it can also be a UNIX machine (Sun™ Solaris™, Hewlett Packard HP-UX, IBM AIX® or other). You can also choose to use a Windows workstation using FTP or HTTP. There are often more issues encountered when using a Windows workstation than a Linux or UNIX workstation to serve the RHEL 5.2 install tree.

The steps in this chapter explain how to configure a PC Linux box as the NFS server. The *Red Hat Installation Guide for the IBM S/390 and IBM System z Architectures* publication provides additional information about the installation options on the Web at:

`http://www.redhat.com/docs/manuals/enterprise/`

Performing the following tasks will set up a Linux NFS server:

- Installing Linux on the PC
- Downloading files associated with this book
- Setting up a RHEL 5.2 install tree
- Enabling the NFS server

## 6.1 Installing Linux on the PC

If you do not have a Linux PC, then you must obtain access to one in the network and install Linux onto it. Describing that process is beyond the scope of this book. However, installing the same distribution onto a PC server that you plan to install on System z is recommended. Doing so will give you practice with the installation process and will give you a reference system that may be helpful in understanding the differences between the Intel® (i386, i686) and System z (s390x) architectures. In this chapter, a PC running RHEL 5.2 is used.

## 6.2 Downloading files associated with this book

This book has files associated with it to make the task of customizing and cloning your virtual servers easier. The tar file is available on the Web at:

`ftp://www.redbooks.ibm.com/redbooks/SG247492/SG24-7492.tgz`

The tar file `SG24-7492.tgz` is about 23 KB in size. Download the file and untar it. The following example shows this being done from a newly created directory `/nfs/`:

```
# mkdir /nfs
# cd /nfs
... download or copy the file SG24-7492.tgz to /nfs/ ...
# tar xzf SG24-7492.tgz
```

List the files in the new directory `virt-cookbook-RH5.2/`:

```
# cd virt-cookbook-RH5.2
# ls
clone-1.0-5.s390x.rpm  README.txt  vm/
```

The `README.txt` file briefly describes each of the files and the one directory; you may want to briefly review that file. You now have downloaded and uncompressed the files associated with this book.

## 6.3 Setting up a RHEL 5.2 install tree

You must have a valid Red Hat entitlement for Linux on IBM System z to access the Red Hat Enterprise Linux 5.2 ISO images. If you do not have one, you can request a free 180-day evaluation copy at:

`http://www.redhat.com/z`

Follow the link named Free Evaluation on the left, then fill out the online form. If you do not have a Red Hat login, you will need to create one by clicking the **Register and Continue** button. Otherwise, enter your Red Hat login and password, then click **Log In** to continue. After completing the form, you will automatically receive an e-mail notification with instructions about how to access the Red Hat Network (RHN), where you can download the installation discs at:

`https://rhn.redhat.com`

You can also click the **Contact Sales** link on the left of the page or call 1-888-733-4281.

## 6.3.1 Copying from physical DVD

RHEL 5.2 is distributed on physical CDs or files that are ISO images of CDs. RHEL 5.2 is also distributed on a single physical DVD disc or a single ISO image. It is easier to work with a single DVD ISO image than to work with multiple CD ISO images, so this approach is recommended.

If you have a physical DVD, but not an ISO image, we recommend that you create an ISO image. You could avoid creating the ISO image and copy the data directly from the DVD to the install tree, but we recommend that you create the ISO image so that you will have a reference file.

Be sure your PC has a DVD drive, not just a CD drive (if you have a PC that only has a CD drive, you can create ISO images of the CDs, but this is not described). Put the DVD in the tray and use the **dd** command to create the ISO image. The device file named `/dev/cdrom` is often associated with the CD/DVD drive, but your device file name may be different. If so, you must determine the correct name.

**Perform these steps only if you are starting with a physical DVD disc:**

```
# cd /nfs
# dd if=/dev/cdrom of=rhel-5.2-server-s390x-dvd.iso
# umount /mnt/cdrom
```

You should now have an ISO image of the DVD.

## 6.3.2 Verifying the ISO image

An important early step is to verify the integrity of DVD ISO image. You perform this verification by comparing a checksum value that was calculated when the DVD was created against a checksum value calculated against your ISO image. If the two checksum values differ, then it means there was an error somewhere in the copying process.

The **md5sum** command allows you to compare checksum files. The checksum value for RHEL 5.2 for the s390x architecture is as follows:

```
# cat MD5SUM
8cdb5ae0d74cbfcd2e8d44a23f69ff70 rhel-5-server-s390x-dvd.iso
```

Run the **md5sum** command against the `MD5SUM` file:

```
# md5sum -c MD5SUM
rhel-5-server-s390x-dvd.iso: OK
```

**Important:** Your MD5SUM file may have checksum values for the DVD *and* the CD ISO images. If this is the case and you only have one DVD ISO image, the md5sum will generate errors of the following form:

```
md5sum: rhel-5-server-s390x-disc1.iso: No such file or directory
rhel-5-server-s390x-disc1.iso: FAILED open or read
```

This is not a problem if the DVD ISO image is reported as `OK`.

If the ISO image does not report `OK`, it must be downloaded or copied again until it does.

## 6.3.3 Copying the DVD contents

Copy the contents of the ISO image to the file system. Temporarily mount it over a new directory `tmp/` using a loopback device:

```
# cd /nfs
# mkdir tmp
# mount -o loop rhel-5-server-s390x-dvd.iso tmp
```

List the contents of the mounted ISO image:

```
# ls tmp
EULA            README-pa.html         RELEASE-NOTES-ml.html
eula.en_US      README-pt_BR.html      RELEASE-NOTES-mr.html
generic.ins     README-ru.html         RELEASE-NOTES-or.html
...
```

Make a new directory, `/nfs/rhel5.2/`, and recursively copy the contents of the DVD to it using the **`cp -a`** command. This will take a number of minutes to complete. Then unmount `tmp/`, as shown here:

```
# cp -a tmp/* rhel5.2/
# umount tmp
```

## 6.3.4 Building the repository directory

For the **`yum`** command to work, a common metadata repository must be built using the **`createrepo`** command. There is a sample repository in the directory `Server/repodata/`. The group XML file named `comps-rhel5-server-core.xml` should be used to create group information:

```
# cd /nfs/rhel5.2/Server/
# mv repodata/ repodata.orig
# createrepo -g repodata.orig/comps-rhel5-server-core.xml .
2495/2495 - junit-javadoc-3.8.2-3jpp.1.s390x.rpm
Saving Primary metadata
Saving file lists metadata
Saving other metadata
```

Now the newly created `repodata/` directory contains the correct common metadata:

```
# cd repodata
# ls
comps-rhel5-server-core.xml  other.xml.gz    repomd.xml
filelists.xml.gz             primary.xml.gz
```

# 6.4 Enabling the NFS server

The method of enabling an NFS server will differ, depending upon the operating system. However, the steps are basically the same:

- Export the appropriate directories.
- Start the NFS server in the current run level.

The directories to export using NFS are set in the `/etc/exports` configuration file. Export the directory `/nfs/rhel5.2/` to make the install tree available and `/nfs/virt-cookbook-RH5.2/` to make the files associated with this book available. First make a backup copy of the file. Then edit the original copy and add the two directories as follows:

```
# cd /etc
# cp exports exports.orig
# vi exports      // add two lines
/nfs/rhel5.2             *(ro,sync)
/nfs/virt-cookbook-RH5.2 *(ro,sync)
```

The `*(ro,sync)` parameter specifies that any client with access to this server can get the NFS mount read-only. You may want to be more restrictive than allowing any client (using the `*`) for security reasons. Type **`man exports`** for more details.

Set the NFS server to start using the **`chkconfig`** command and start it on for the current session with the **`service nfs start`** command:

```
# chkconfig nfs on
# chkconfig --list nfs
nfs             0:off   1:off   2:on    3:on    4:on    5:on    6:off
# service nfs start
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
```

Your NFS server should now be running with the directory exported. We recommend that you test this by mounting the exported directory locally. The following example shows that the `/mnt/` directory is empty. Then the newly exported `/nfs/` directory is mounted and the files are listed.

```
# mkdir /mnt/tmp
# mount localhost:/nfs/rhel5.2/ /mnt/tmp
# ls -F /mnt/tmp
EULA            README-or.html         RELEASE-NOTES-ja.html
eula.en_US      README-pa.html         RELEASE-NOTES-ko.html
generic.ins     README-pt_BR.html      RELEASE-NOTES-ml.html
GPL             README-ru.html         RELEASE-NOTES-mr.html
images/         README-si.html         RELEASE-NOTES-or.html
...
```

This shows that the RHEL 5.2 install tree is accessible using NFS. Now unmount it and test the `virt-cookbook-RH5.2/` directory:

```
# umount /mnt/tmp
# mount localhost:/nfs/virt-cookbook-RH5.2 /mnt/tmp
# ls -F /mnt/tmp
clone-1.0-5.s390x.rpm  README.txt  vm/
# umount /mnt/tmp
```

You should now be able to use this server as the source of a RHEL 5.2 mainframe Linux installation. Later you will be able to copy the install tree to a System z Linux virtual server.

7

# Installing RHEL 5.2 on the controller

**Note:** Before proceeding with this chapter, you must complete the tasks described in Chapter 4, “Installing and configuring z/VM” on page 27, Chapter 5, “Servicing z/VM” on page 71, and Chapter 6, “Configuring a Network File System server for RHEL 5.2” on page 85.

At this point, you must have created a new z/VM user ID, `LNXMAINT`. Now it is time to create the first Linux user ID, `LNXINST`. This Linux ID is the controller installation server, and serves as the administration point for future Linux IDs. This server is also referred to as the controller. LNXINST serves the following purposes:

- Red Hat Enterprise Linux 5.2 installation server: This is a tree of Red Hat packages (RPMs) and other files required for installation.
- Network File System (NFS) server: This exports the installation tree and possibly other useful files.
- Clone server: This is for cloning an existing installation to a new Linux ID. See Chapter 9, “Configuring RHEL 5.2 for cloning” on page 129.
- Kickstart server: This hosts files necessary for automated installations. See Chapter 10, “Installing Linux with kickstart” on page 147.

In this chapter, you will complete the following tasks:

- Installing the controller
- Configuring the controller

## 7.1 Installing the controller

In this section you will install the RHEL 5.2 controller under the user LNXINST. This is the guest that will serve as the installation and file server for future Linux guests.

### 7.1.1 Creating the user ID LNXINST

In this section you will define the LNXINST user ID to z/VM.

1. Logon to MAINT and edit the USER DIRECT file:

   ```
   ==> x user direct c
   ```

   In the USER DIRECT file, you can group statements that will be common to many user definitions in a construct called a *profile*. This profile can then become part of the user definitions using the INCLUDE statement. You used the existing profile TCPCMSU when you defined the LNXMAINT user.

2. Create a new profile named LNXDFLT. This will contain the user directory statements that will be common to all Linux user IDs. To save typing, you can use the **""** prefix commands to duplicate the IBMDFLT profile that should be on lines 37-50:

   ```
   ""037 **************************************************************
   00038 *
   00039 PROFILE IBMDFLT
   00040   SPOOL 000C 2540 READER *
   00041   SPOOL 000D 2540 PUNCH A
   00042   SPOOL 000E 1403 A
   00043   CONSOLE 009 3215 T
   00044   LINK MAINT 0190 0190 RR
   00045   LINK MAINT 019D 019D RR
   00046   LINK MAINT 019E 019E RR
   00047   LINK MAINT 0402 0402 RR
   00048   LINK MAINT 0401 0401 RR
   00049   LINK MAINT 0405 0405 RR
   ""050 ****************************
   ```

3. Edit the duplicated profile by deleting the three LINK MAINT 040x lines, and inserting the lines that are shown in bold font:

   ```
   PROFILE LNXDFLT
     IPL CMS                                 1
     MACHINE ESA 4                           2
     CPU 00 BASE                             3
     NICDEF 600 TYPE QDIO LAN SYSTEM VSW1    4
     SPOOL 000C 2540 READER *
     SPOOL 000D 2540 PUNCH A
     SPOOL 000E 1403 A
     CONSOLE 009 3215 T
     LINK MAINT 0190 0190 RR
     LINK MAINT 019D 019D RR
     LINK MAINT 019E 019E RR
     LINK LNXMAINT 192 191 RR                5
     LINK TCPMAINT 592 592 RR                6
   ```

   The numbers in bold font on the command lines refer to the descriptions that follow.

   **1** CMS will be IPLed when the user ID is logged onto.
   **2** The machine will of type ESA with a maximum of 4 CPUs that can be defined.
   **3** This defines the base CPU.
   **4** This defines a virtual NIC connected to the VSWITCH starting at virtual address 600.

5 This provides read access to the LNXMAINT 192 disk as the user's 191 disk.
6 This provides read access to the TCPMAINT 592 disk, so that the user has access to TCP/IP services such as FTP.

4. Go to the bottom of the file and add the definition for a new user ID named RHEL52. This user ID is given the class B privilege (aside from the typical class G) in order to run the **FLASHCOPY** command. It is also given the class D privilege to run the **QUERY ALLOC MAP** command, and the class E privilege to run the **QUERY NSS** command. Be sure to replace the volume labels in that are in brackets (for example, <MMD856>) with the labels of your DASD:

```
USER LNXINST LNX4VM 256M 1G BDEG
  INCLUDE LNXDFLT
  OPTION LNKNOPAS APPLMON
  MDISK 100 3390 0001 3338 <MMD856> MR LNX4VM LNX4VM LNX4VM
  MDISK 101 3390 0321 3018 <MMD857> MR LNX4VM LNX4VM LNX4VM
  MDISK 102 3390 0001 3338 <MMD952> MR LNX4VM LNX4VM LNX4VM
  MDISK 103 3390 0001 3338 <MMD953> MR LNX4VM LNX4VM LNX4VM
  MDISK 104 3390 0001 3338 <MMD954> MR LNX4VM LNX4VM LNX4VM
*
```

This Linux user ID will have the minidisks and virtual disks (VDISKs) described in Table 7-1.

*Table 7-1 Minidisks to be defined*

| Minidisk | Description |
|---|---|
| 100 | The root file system of the Linux controller. This will serve as the administration point for all your Linux virtual servers. |
| 101-104 | Minidisks used to create a logical volume mounted over /nfs/ for making the RHEL 5.2 installation tree and the files associated with this book available using NFS. |
| 300-301 | These are virtual disk (VDISK) swap spaces that are not defined in the USER DIRECT file, but instead are defined by calls to the **SWAPGEN EXEC** in the user's **PROFILE EXEC** so that when the user ID logs on, the VDISKs are created. |

5. Go back to the top of the file and search for string USER $ALLOC$. Add cylinder 0 of each of the five new volumes to this dummy user ID so that they do not show up as gaps in the USER DISKMAP report file:

```
====> top
====> /user $alloc$
USER $ALLOC$  NOLOG
 MDISK A01 3390 000 001 540RES R
 MDISK A02 3390 000 001 540W01 R
 MDISK A03 3390 000 001 540W02 R
 MDISK A04 3390 000 001 <MMD857> R
 MDISK A05 3390 000 001 <MMD856> R
 MDISK A06 3390 000 001 <MMD952> R
 MDISK A07 3390 000 001 <MMD953> R
 MDISK A08 3390 000 001 <MMD954> R
...
====> file
```

6. Run **DISKMAP** to check for overlaps and gaps. You should only see the single 501 cylinder gap.

```
==> diskmap user
==> x user diskmap
```

```
====> all /gap/|/overlap/
--------------------  4  line(s) not displayed --------------------
                                             0         500        501    GAP
--------------------  6  line(s) not displayed --------------------
                                             0           0          1    GAP
--------------------  355  line(s) not displayed --------------------
====> quit
```

7. When the disk layout is correct, run **DIRECTXA** to bring the changes online:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 5 RELEASE 3.0
EOJ DIRECTORY UPDATED AND ON LINE
```

You have now defined the user ID that will be both the master Linux image and the controller.

### 7.1.2 Adding LNXINST to AUTOLOG1's PROFILE EXEC

The new Linux ID you defined needs access to the VSWITCH. A **SET VSWITCH** command with the **GRANT** parameter can be added to the AUTOLOG1 **PROFILE EXEC** to do this. Also, an **XAUTOLOG** statement can be added if the user ID is automatically logged on at z/VM IPL time.

Link and access the AUTOLOG1 191 disk as read/write and edit the file **PROFILE EXEC**. Add the RHEL52 user ID to the sections that grant access to the VSWITCH and that **XAUTOLOG** the Linux user IDs:

```
==> link autolog1 191 1191 mr
==> acc 1191 f
==> x profile exec f   // add two lines
/***************************/
/*  Autolog1 Profile Exec  */
/***************************/
'cp xautolog tcpip'                 /* start up TCPIP */
'CP XAUTOLOG DTCVSW1'               /* start VSWITCH controller 1 */
'CP XAUTOLOG DTCVSW2'               /* start VSWITCH controller 2 */
'cp set pf12 ret'                   /* set the retrieve key */
'cp set mdc stor 0m 128m'           /* Limit minidisk cache in CSTOR */
'cp set mdc xstore 0m 0m'           /* Disable minidisk cache in XSTOR */
'cp set srm storbuf 300% 250% 200%' /* Overcommit memory */
'cp set signal shutdown 300'        /* Allow guests 5 min to shut down */

/* Grant access to VSWITCH for each Linux user */
'cp set vswitch vsw1 grant lnxinst'

/* XAUTOLOG each Linux user that should be started */
'cp xautolog lnxinst'

'cp logoff'                          /* logoff when done */
====> file
```

These changes will not take effect until the next IPL, so you must grant this user ID access to the VSWITCH for this z/VM session. This is done as follows:

```
==> set vswitch vsw1 grant lnxinst
Command complete
```

### 7.1.3 Preparing LNXINST bootstrap files

To IPL a RHEL 5.2 installation system, four bootstrap files must be prepared. Three are punched to z/VM reader and then IPLed. These three files IPLed are a kernel, a parameter file, and an initial RAMdisk. The fourth file is a configuration file stored on a CMS disk that the parameter file points to.

Think of these as the files that are on as a PC Linux boot CD (or floppy disk). Also, a small REXX EXEC is commonly used to clean out the reader, punch the three files, and IPL the reader.

A sample RHEL52 parameter file, configuration file, and RHEL52 EXEC are supplied and should be on the LNXMAINT 192 disk. This is described in 7.1.1, “Creating the user ID LNXINST” on page 92. Therefore, only the kernel and RAMdisk should need to be copied.

1. Start an SSH session as root on the NFS server.
2. Use the **ftp** command to copy the RHEL 5.2 kernel and initial RAMdisk to the LNXMAINT D disk. These files must have a record format of fixed 80 byte records. This format can be set with the **site fix 80** FTP subcommand (if this subcommand fails, try **quote site fix 80**). Following is an example:

```
# cd /nfs/rhel5.2/images
# ftp <9.12.5.22>
Name (9.12.5.22:root): lnxmaint
Password:
230 LNXMAINT logged in; working directory = LNXMAINT 191
Remote system type is z/VM.
ftp> cd lnxmaint.192
250 Working directory is LNXMAINT 192
ftp> site fix 80
200 Site command was accepted.
ftp> bin
200 Representation type is IMAGE.
ftp> put initrd.img rhel52.initrd
local: initrd.img remote: rhel52.initrd
12237790 bytes sent in 01:28 (135.35 KB/s)
ftp> put kernel.img rhel52.kernel
local: kernel.img remote: rhel52.kernel
3629384 bytes sent in 00:25 (138.96 KB/s)
ftp> quit
```

3. Go back to your 3270 session. Logoff from MAINT and logon to LNXMAINT.
4. The files LNXINST PARM-RH5, LNXINST CONF-RH5, and RHEL52 EXEC should exist on the LNXMAINT 192 (D) disk because they were copied in 4.7.5, “Copying files associated with this book to LNXMAINT” on page 58.

   Use the **FILELIST** command to verify that the files were copied, and that the kernel and initial RAMdisk were copied in Fixed 80 byte record format. You should see the following files (the number of records and blocks may vary):

```
==> filel * * d
LNXMAINT FILELIST A0  V 169  Trunc=169 Size=5 Line=1 Col=1 Alt=0
Cmd   Filename Filetype Fm Format Lrecl    Records     Blocks   Date
      RHEL52   KERNEL   D1 F         80      47182        864  7/15/08 12:04:20
      RHEL52   INITRD   D1 F         80     164400       3211  7/15/08 12:04:07
      CHPW540  XEDIT    D1 V         77        194          3  7/15/08  9:43:52
      CPFORMAT EXEC     D1 V         79        252          3  7/15/08  9:43:52
      LABEL540 EXEC     D1 V         77        116          2  7/15/08  9:43:52
      LABEL540 XEDIT    D1 V         77         50          1  7/15/08  9:43:52
      LBL540-9 EXEC     D1 V         77        105          2  7/15/08  9:43:52
      LBL540-9 XEDIT    D1 V         77         49          1  7/15/08  9:43:52
      LNXINST  CONF-RH5 D1 V         38         15          1  7/15/08  9:43:52
      LNXINST  PARM-RH5 D1 V         43          3          1  7/15/08  9:43:52
      PROFILE  EXEC     D1 V         63         17          1  7/15/08  9:43:52
```

```
RHEL52   EXEC     D1 V         69         9          1  7/15/08  9:43:52
SWAPGEN  EXEC     D1 V         72       358          5  7/15/08  9:43:52
```

5. Quit by pressing **F3**.
6. Verify that the file **RHEL52 EXEC** has the correct information. Note the kernel and RAMdisk have hardcoded file names (RHEL52), but the file name of the parameter file will be the user ID (**userid()** function) of the user running the EXEC:

```
==> type rhel52 exec d
/* EXEC to punch RHEL 5.2 install system to reader and IPL from it */
'CP SPOOL PUN *'
'CP CLOSE RDR'
'PUR RDR ALL'
'PUN RHEL52       KERNEL * (NOH'
'PUN' userid() 'PARM-RH5 * (NOH'
'PUN RHEL52       INITRD * (NOH'
'CH RDR ALL KEEP'
'IPL 00C CLEAR'
```

7. Two text files are needed to install RHEL 5.2: a parameter file, and a configuration file. The parameter file is named RHEL52 PARM-RH5, and is punched to the reader. This file has some values. The most important value, the CMSCONFFILE variable, points to the configuration file, which remains on a CMS minidisk. The sample parameter file should not need to be modified:

```
==> type lnxinst parm-rh5 d
ramdisk_size=40000 root=/dev/ram0 ro ip=off
CMSDASD=191 CMSCONFFILE=LNXINST.CONF-RH5
vnc vncpassword=lnx4vm
```

8. Enter the appropriate networking values in the configuration file, in this case LNXINST CONF-RH5. The sample configuration file contains some fields that should be correct, and many other fields that will have to be modified.

   Modify at least the fields shown in bold font and perhaps others; refer to the worksheet in 2.7.4, “Linux user ID worksheet” on page 18, for guidance.

> **Note:** In the following examples, the DASD range includes disks that do not yet exist. This is done so that disks can be added in the future without affecting the device naming convention used in Linux. Think of this as “reserving” slots 100-10f, for example, so that disk 300 always gets assigned the same device in Linux.

Before:

```
DASD=100-10f,300-30f
HOSTNAME=hostName.DNSname.com
NETTYPE=qeth
IPADDR=n.n.n.n
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
NETWORK=n.n.n.n
NETMASK=255.255.255.0
SEARCHDNS=DNSname.com
BROADCAST=n.n.n.n
GATEWAY=n.n.n.n
DNS=n.n.n.n
MTU=1500
PORTNAME=DONTCARE
LAYER2=0
VSWITCH=1
```

After:

```
==> x lnxinst conf-rh5
DASD=100-10f,300-30f
HOSTNAME=<virtc530.itso.ibm.com>
NETTYPE=qeth
IPADDR=<9.12.5.30>
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
NETWORK=<9.12.5.255>
NETMASK=<255.255.252.0>
SEARCHDNS=<itso.ibm.com>
BROADCAST=<9.12.4.255>
GATEWAY=<9.12.4.1>
DNS=<9.12.6.7>
MTU=1500
PORTNAME=DONTCARE
LAYER2=0
VSWITCH=1
```

> **Note:** The RHEL 5.2 installer supports OSA/NIC in layer 2 (Ethernet) mode. In the preceding example, we are connecting to a layer 3 VSWITCH, so we set the parameter LAYER2=0. When connecting in layer 2 mode, set `LAYER2=1`. Then, if this guest is connected to a VSWITCH, set `VSWITCH=1`, thus signifying that the VSWITCH will provide the MAC address. If this guest is not connected to a VSWITCH, set `VSWITCH=0` and add the parameter `MACADDR=` followed by the MAC address for this guest.

9. Save your changes with the **`FILE`** subcommand.

   You should still be logged on to `LNXMAINT`. When you log on to the new `LNXINST` user ID, you will pick up a `PROFILE EXEC` from `LNXMAINT 192`. This file runs when you press **Enter** at the `VM READ` prompt, and it creates two VDISKs using the `SWAPGEN EXEC`, to be used later as swap spaces. View the contents of the `PROFILE EXEC` using the CMS **`TYPE`** command:

```
==> type profile exec d
/* PROFILE EXEC for Linux virtual servers */
'CP SET RUN ON'
'CP SET PF11 RETRIEVE FORWARD'
'CP SET PF12 RETRIEVE'
'ACC 592 C'
'SWAPGEN 300 524288' /* create a 256M VDISK disk swap space */
'SWAPGEN 301 1048576' /* create a 512M VDISK disk swap space */
'PIPE CP QUERY' userid() '| var user'
parse value user with id . dsc .
if (dsc = 'DSC') then /* user is disconnected */
  'CP IPL 100'
else  /* user is interactive -> prompt */
 do
  say 'Do you want to IPL Linux from minidisk 100? y/n'
  parse upper pull answer .
  if (answer = 'Y') then 'CP IPL 100'
 end /* else */
```

## 7.1.4 Beginning the Linux installation

Perform the following steps to prepare for installing Linux:

1. Logoff from LNXMAINT and logon to LNXINST. When you log on, you should see a message indicating that a virtual NIC has been created at addresses `600-602` (using the

**NICDEF** statement in the USER DIRECT file), and that two VDISKs have been created at virtual addresses 300 and 301 (using the **SWAPGEN EXEC** called from the **PROFILE EXEC**):

```
LOGON LNXINST
00: NIC 0600 is created; devices 0600-0602 defined
00: z/VM Version 5 Release 3.0, Service Level 0702 (64-bit),
00: built on IBM Virtualization Technology
00: There is no logmsg data
00: FILES:   NO RDR,   NO PRT,   NO PUN
00: LOGON AT 11:10:42 EST WEDNESDAY 12/12/07
z/VM V5.3.0    2007-11-18 11:27

DMSACP723I A (191) R/O
DMSACP723I C (592) R/O
DIAG swap disk defined at virtual address 300 (64989 4K pages of swap space)
DIAG swap disk defined at virtual address 301 (129981 4K pages of swap space)
```

2. You are prompted to IPL Linux, but because you have not installed Linux yet, reply n:

```
Do you want to IPL Linux from minidisk 100? y/n
==> n
```

3. Before you install Linux, it is useful to verify the resources. Verify that you have DASD (minidisks) at virtual addresses 100-104 and 300-301 with the **QUERY VIRTUAL** command (other Linux IDs without class B privilege will just use the **QUERY** command):

```
==> q v 100-104
00: DASD 0100 3390 MMD856 R/W       3338 CYL ON DASD  AD24 SUBCHANNEL = 0002
00: DASD 0101 3390 MMD857 R/W       3338 CYL ON DASD  AD25 SUBCHANNEL = 0003
00: DASD 0102 3390 MMD952 R/W       3338 CYL ON DASD  AD26 SUBCHANNEL = 0004
00: DASD 0103 3390 MMD953 R/W       3338 CYL ON DASD  AD27 SUBCHANNEL = 0005
00: DASD 0104 3390 MMD954 R/W       3338 CYL ON DASD  AE22 SUBCHANNEL = 0006
==> q v 300-301
00: DASD 0300 9336 (VDSK) R/W     524288 BLK ON DASD  VDSK SUBCHANNEL = 0013
00: DASD 0301 9336 (VDSK) R/W    1048576 BLK ON DASD  VDSK SUBCHANNEL = 0014
```

4. Verify that you have a virtual OSA at addresses 600-602 by using the **QUERY VIRTUAL OSA** command:

```
==> q v osa
00: OSA  0600 ON NIC  0600  UNIT 000 SUBCHANNEL = 0007
00:      0600 DEVTYPE OSA         CHPID OE OSD
00:      0600 QDIO-ELIGIBLE       QIOASSIST-ELIGIBLE
00: OSA  0601 ON NIC  0600  UNIT 001 SUBCHANNEL = 0008
00:      0601 DEVTYPE OSA         CHPID OE OSD
00:      0601 QDIO-ELIGIBLE       QIOASSIST-ELIGIBLE
00: OSA  0602 ON NIC  0600  UNIT 002 SUBCHANNEL = 0009
00:      0602 DEVTYPE OSA         CHPID OE OSD
00:      0602 QDIO-ELIGIBLE       QIOASSIST-ELIGIBLE
```

5. Use the **QUERY VIRTUAL STORAGE** command to show that you have a 256 MB machine:

```
==> q v stor
00: STORAGE = 256M
```

6. This is adequate memory to run a RHEL 5.2 Linux image. However, it is often too small in which to install Linux. Temporarily modify the storage up to 512 MB with the **DEFINE STORAGE** command. Then **IPL CMS** and again reply n to the question of IPLing Linux:

```
==> def stor 512m
00: STORAGE = 512M
00: Storage cleared - system reset.
==> ipl cms
z/VM V5.3.0    2007-117-18 09:53
```

```
DMSACP723I A (191) R/O
DMSACP723I C (592) R/O
DIAG swap disk defined at virtual address 300 (64989 4K pages of swap space)
DIAG swap disk defined at virtual address 301 (129981 4K pages of swap space)
Do you want to IPL Linux from minidisk 100? y/n
n
```

7. Verify that you have a 512 MB virtual machine:

   ```
   ==> q v stor
   00: STORAGE = 512M
   ```

   This change is for the duration of the user ID session. When you log off and log back on this user ID, the storage will revert to 256 MB.

Now you are ready to begin the Linux installation.

### 7.1.5 Stage 1 of the RHEL 5.2 installation

In this section you will install Linux to the controller. Follow these steps to begin the installation of RHEL 5.2.

1. Run the **RHEL52 EXEC**. You should see many windows of questions and answers scrolling by. If you had used the default parameter file shipped with RHEL 5.2, you would have had to answer all the networking questions manually. With the proper parameters set in LNXINST CONF-RH5, the install process should proceed to where you have to use a browser to VNC client get into the installation program:

   ```
   ==> rhel52
      00: RDR FILE 0019 SENT FROM RHEL52    PUN WAS 0019 RECS 045K CPY  001 A NOHOLD NO
      KEEP
      00: RDR FILE 0020 SENT FROM RHEL52    PUN WAS 0020 RECS 0003 CPY  001 A NOHOLD NO
      KEEP
      00: RDR FILE 0021 SENT FROM RHEL52    PUN WAS 0021 RECS 153K CPY  001 A NOHOLD NO
      KEEP
      00: 0000003 FILES CHANGED
      Linux version 2.6.18-92.el5 (brewbuilder@btest1.z900.redhat.com) (gcc version 4.
      1.2 20071124 (Red Hat 4.1.2-41)) #1 SMP Tue Apr 1 19:03:24 EDT 2008
      We are running under VM (64 bit mode)
      Detected 2 CPU's
      Boot cpu address  0
      Built 1 zonelists.  Total pages: 131072
      Kernel command line: ramdisk_size=40000 root=/dev/ram0 ro ip=off
                              CMSDASD=191 CMSCONFFILE=LNXINST.CONF-RH5
                           vnc vncpassword=lnx4vm
      ...
   ```

   The kernel should continue to boot until you see the following messages:

   ```
   ...
   Starting telnetd and sshd to allow login over the network.

   Connect now to 9.12.5.30 to start the installation.
   ```

2. From your workstation, use your SSH client (for example, PuTTY) to connect to the IP address and begin the installation. When prompted for a user name, enter root. A password will not be required.

   Figure 7-1 on page 100 shows the initial window of the installer. Use the **Tab** key to move between fields. Use the arrow keys to move among choices, and press **Enter** to select a choice.

**Important:** If the DASD you are using has never been formatted for Linux, you may get many windows of warning messages similar to the following on your 3270 session:

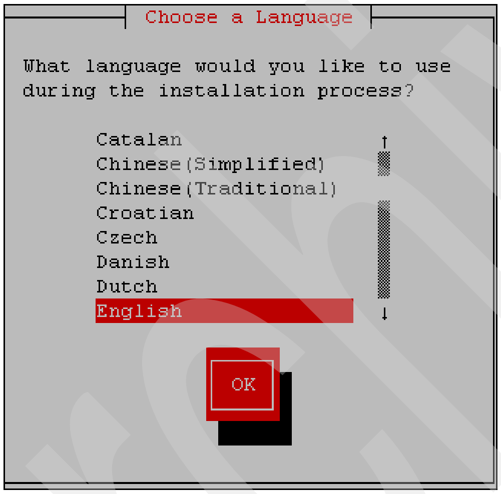
```
dasd(eckd): I/O status report for device 0.0.0100:
dasd(eckd): in req: 000000000e027ee8 CS: 0x40 DS: 0x0E
dasd(eckd): device 0.0.0100: Failing CCW: 000000000e027fd0
dasd(eckd): Sense(hex)  0- 7: 00 08 00 00 04 ff ff 00
```

This is not a problem, you simply have to clear the window many times or else the install process will freeze. An alternative to clearing the window many times is to issue the following CP **TERM** command:

```
#cp term more 0 0
```

Press **Enter** and the window should scroll freely. The downside of this option is that you may miss some messages that are important. You may later want to set the value back to the default of waiting 50 seconds to beep, and then waiting another 10 seconds to clear the window, using the following command:

```
#cp term more 50 10
```

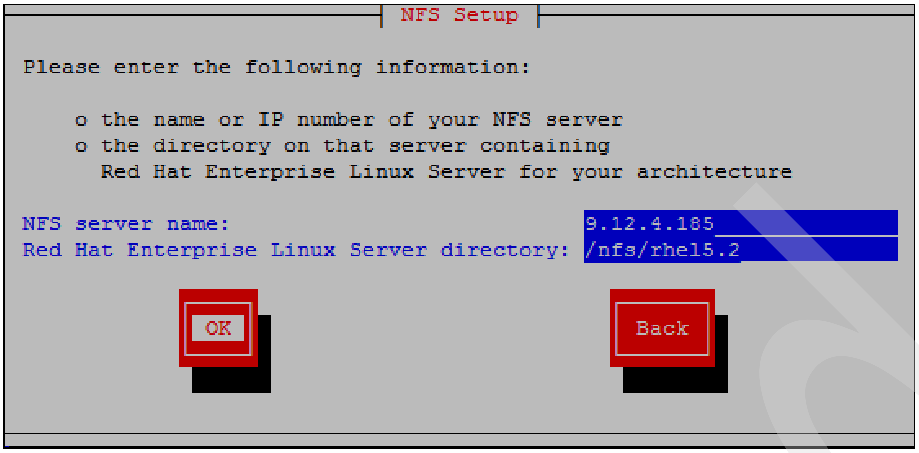


Choose a Language

What language would you like to use during the installation process?

*Figure 7-1 Initial window of installer*

3. The Choose a Language window should appear. Select your language, tab to **OK**, and then press **Enter**.
4. The Installation Method window should appear. Choose **NFS image** for the install method, and select **OK**.
5. The NFS Setup window should appear. Enter the IP address of the PC NFS server on the first line, then enter the path to the installation tree on the second line, and select **OK**. The example shown in Figure 7-2 on page 101 uses the NFS server at IP address 9.12.4.185.



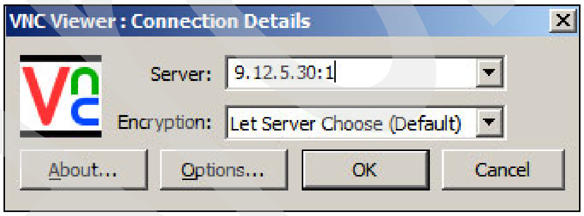
```
NFS Setup

Please enter the following information:

    o the name or IP number of your NFS server
    o the directory on that server containing
      Red Hat Enterprise Linux Server for your architecture

NFS server name:                            9.12.4.185
Red Hat Enterprise Linux Server directory:  /nfs/rhel5.2
```

*Figure 7-2 NFS setup window*

6. Now the curses windows should end and the install program (anaconda) should start a VNC server. You should see messages similar to the following:

```
Welcome to the anaconda install environment 1.1 for zSeries

Running anaconda, the Red Hat Enterprise Linux Server system installer - please
wait...
Starting VNC...
The VNC server is now running.
Please connect to 9.12.5.30:1 to begin the install...
Starting graphical installation...

Press <enter> for a shell
XKB extension not present on :1
```

7. Start a VNC client (for example, RealVNC) and connect to the server with your IP address with a `:1` appended to the end, as shown in Figure 7-3. When prompted for a password, enter the password specified in the `LNXINST PARM-RH5` file (`lnx4vm`, in the sample file). In Figure 7-3, Linux is being installed with the IP address 9.12.5.30.



VNC Viewer : Connection Details

Server: 9.12.5.30:1

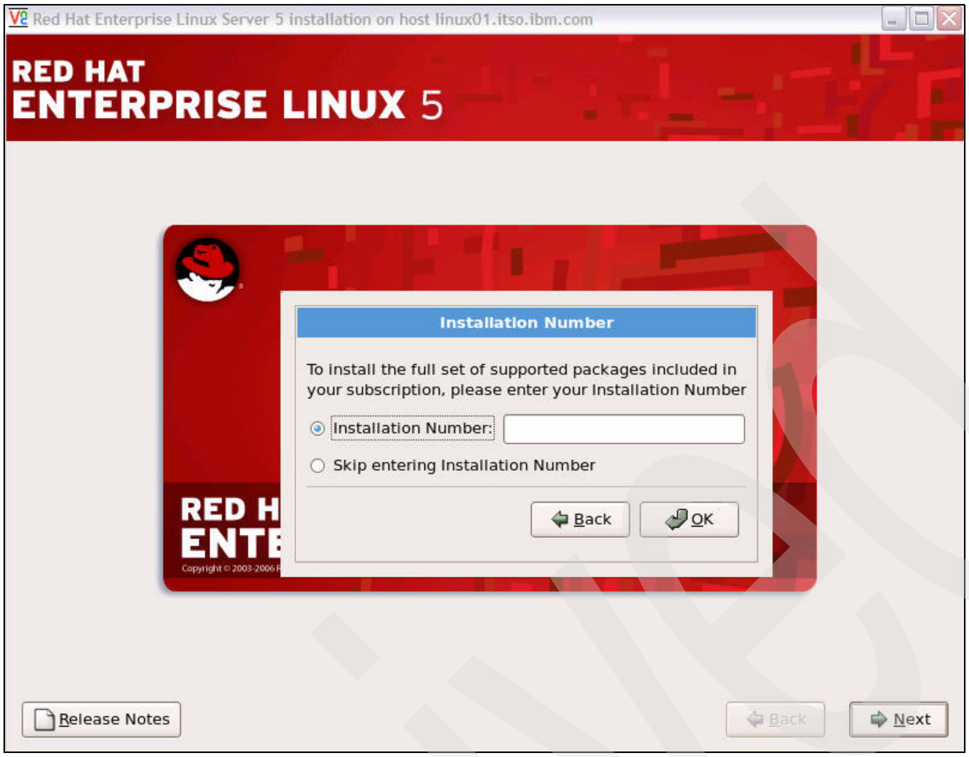
Encryption: Let Server Choose (Default)

*Figure 7-3 Connecting with VNC client*

## 7.1.6 Stage 2 of the RHEL 5.2 installation

After you have connected using VNC, perform the following steps:

1. The welcome window opens. Click **Next**.
2. The installer prompts for an installation number as shown in Figure 7-4 on page 102, which determines the packages that are available for installation. This information should have come with the details of your subscription. If you select Skip, this will result in a core server installation, and additional functionality can be added manually after installation. In this example, **Skip** was chosen.


Red Hat Enterprise Linux Server 5 installation on host linux01.itso.ibm.com

RED HAT ENTERPRISE LINUX 5

Installation Number

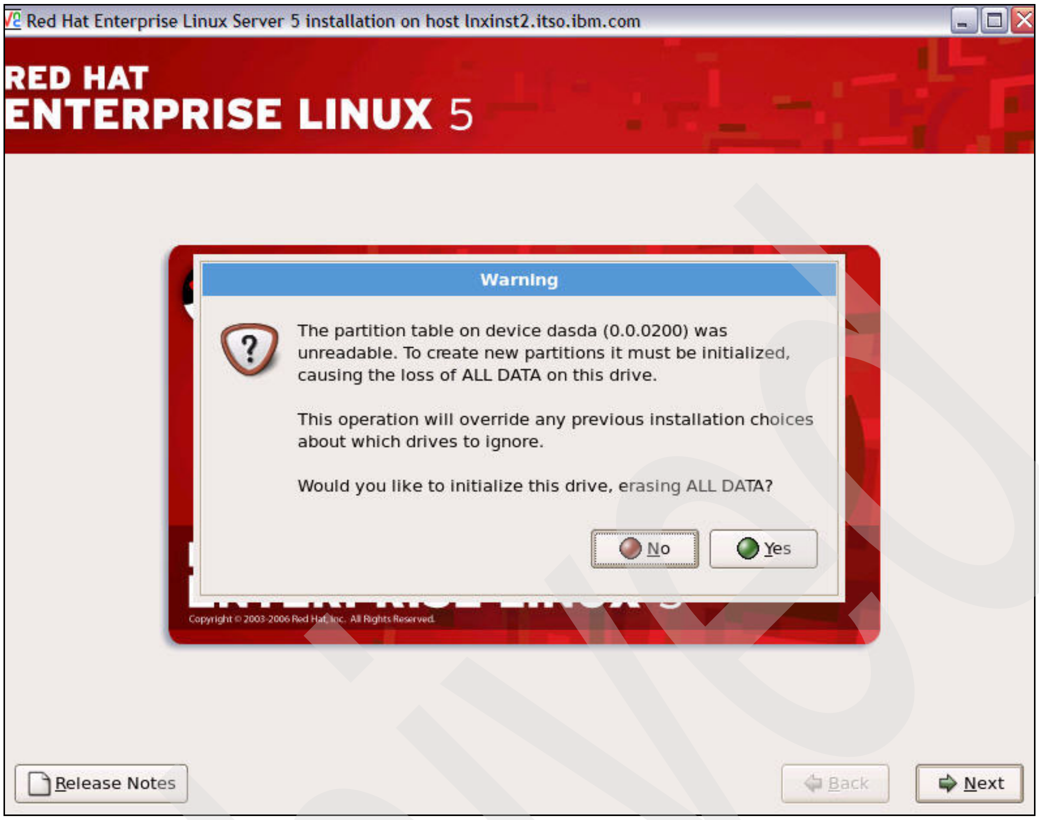
To install the full set of supported packages included in your subscription, please enter your Installation Number

● Installation Number:

○ Skip entering Installation Number

*Figure 7-4 Enter installation number*

3. Because the disks were just formatted by **CPFMTXA** (using the **CPFORMAT EXEC** wrapper), you will be prompted to format each of them. Click **Yes** to format `dasda` through `dasde` (minidisks 100-104) as shown in Figure 7-5 on page 103. A progress indicator will be shown. You will be asked to format disks `300` and `301` because they are the VDISK swap spaces, even though they do not appear in the installer. The VDISK swap spaces will be configured after installation.
4. When the installer prompts you to initialize the drive, select **Yes**.


Red Hat Enterprise Linux Server 5 installation on host lnxinst2.itso.ibm.com
RED HAT
ENTERPRISE LINUX 5
Copyright © 2003-2006 Red Hat, Inc. All Rights Reserved.
Release Notes
Back
Next

Warning

The partition table on device dasda (0.0.0200) was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

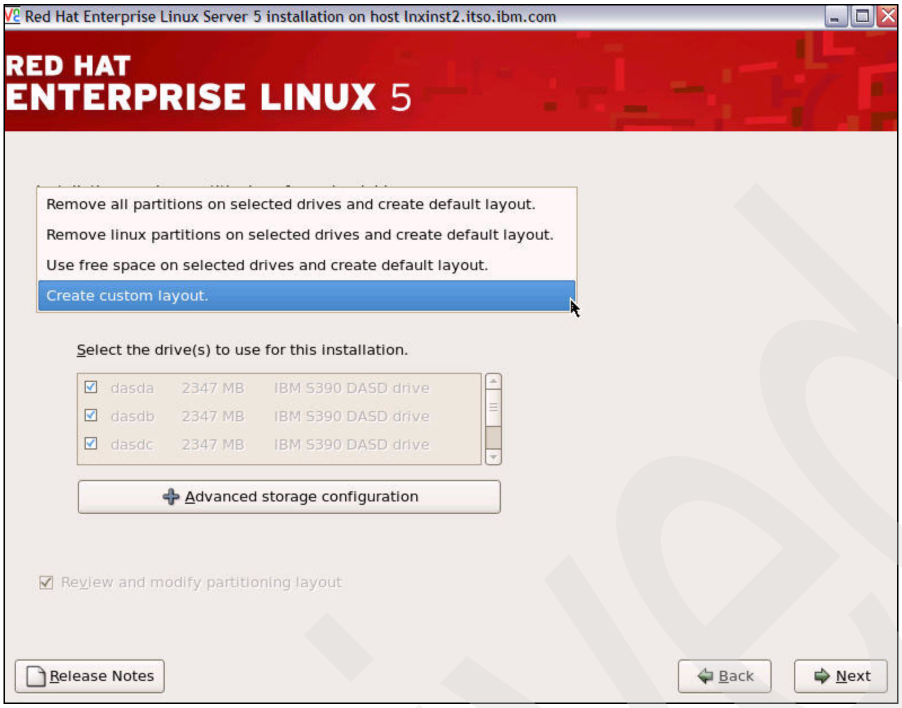
This operation will override any previous installation choices about which drives to ignore.

Would you like to initialize this drive, erasing ALL DATA?

No Yes

*Figure 7-5   Formatting DASD for Linux use*

5. The installer now searches for a previous installation. If this is the first installation, nothing will be found. If a previous installation is found, the installer prompts you whether to Upgrade or Install. Choose **Install** here.
6. At the disk partitioning setup window, choose **Create Custom Layout** as shown in Figure 7-6 on page 104. Click **Next** to proceed to the Disk Setup window.



Red Hat Enterprise Linux Server 5 installation on host lnxinst2.itso.ibm.com

RED HAT
ENTERPRISE LINUX 5

Remove all partitions on selected drives and create default layout.
Remove linux partitions on selected drives and create default layout.
Use free space on selected drives and create default layout.
Create custom layout.

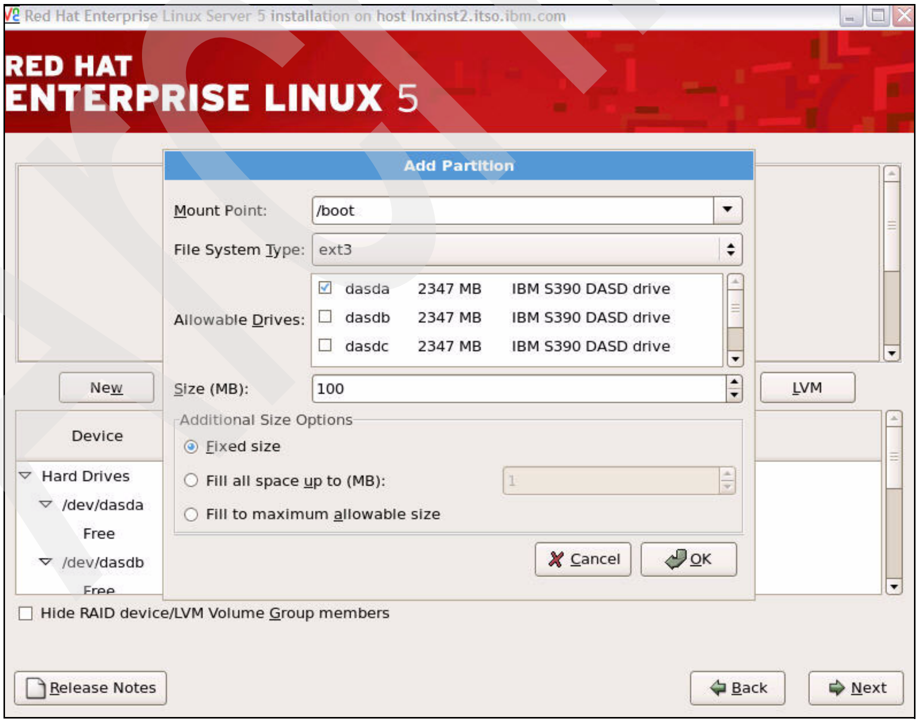
Select the drive(s) to use for this installation.

| | | | |
|---|---|---|---|
| ☑ | dasda | 2347 MB | IBM S390 DASD drive |
| ☑ | dasdb | 2347 MB | IBM S390 DASD drive |
| ☑ | dasdc | 2347 MB | IBM S390 DASD drive |

☑ Review and modify partitioning layout

*Figure 7-6 Choosing custom disk partitioning*

7. Disks `dasda` through `dasde` should show free space. Click the **New** button to add a /boot file system to the 100 disk. Enter **`/boot`** in the field Mount Point, choose to format the file system as ext3, and deselect all drives *except* `dasda`, as shown in Figure 7-7. Leave the default size of 100 in the Size (MB) field and click **OK**.



Red Hat Enterprise Linux Server 5 installation on host lnxinst2.itso.ibm.com

RED HAT
ENTERPRISE LINUX 5

Add Partition

Mount Point: /boot

File System Type: ext3

Allowable Drives:

| | | | |
|---|---|---|---|
| ☑ | dasda | 2347 MB | IBM S390 DASD drive |
| ☐ | dasdb | 2347 MB | IBM S390 DASD drive |
| ☐ | dasdc | 2347 MB | IBM S390 DASD drive |

Size (MB): 100

Additional Size Options
● Fixed size
○ Fill all space up to (MB): 1
○ Fill to maximum allowable size

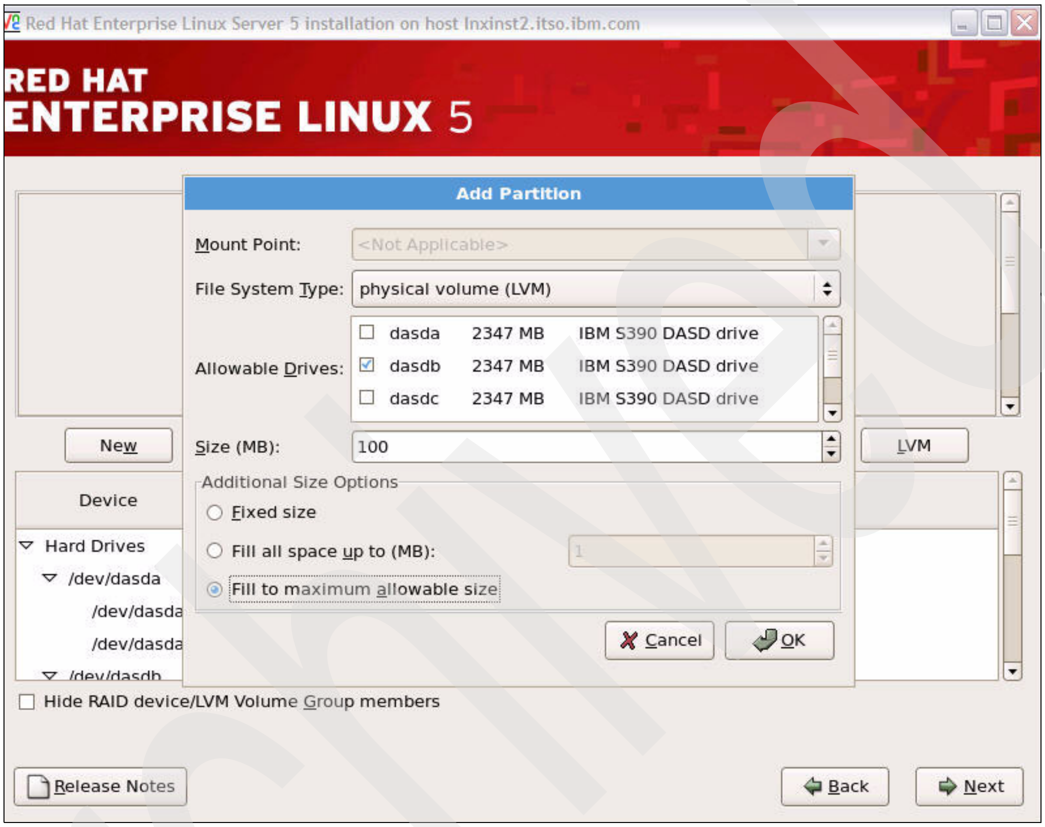
Device
Hard Drives
/dev/dasda
Free
/dev/dasdb
Free

☐ Hide RAID device/LVM Volume Group members

*Figure 7-7 Adding the /boot file system*

8. Click the **New** button again, again deselecting all drives except `dasda`. Under `Mount Point`, select **/** (forward slash) to specify the root partition. Choose **Fill to maximum allowable size** and click **OK**.

9. Create the LVM by clicking the **New** button again. Under File System Type, choose **physical volume (LVM)**. Deselect all drives *except* `dasdb` as Figure 7-8 shows. Under `Additional Size Options`, select **Fill to maximum allowable size** and click **OK**.


Red Hat Enterprise Linux Server 5 installation on host lnxinst2.itso.ibm.com

Add Partition

Mount Point: <Not Applicable>

File System Type: physical volume (LVM)

Allowable Drives:

| | | | |
|---|---|---|---|
| □ | dasda | 2347 MB | IBM S390 DASD drive |
| ☑ | dasdb | 2347 MB | IBM S390 DASD drive |
| □ | dasdc | 2347 MB | IBM S390 DASD drive |

Size (MB): 100

Additional Size Options

○ Fixed size

○ Fill all space up to (MB): 1

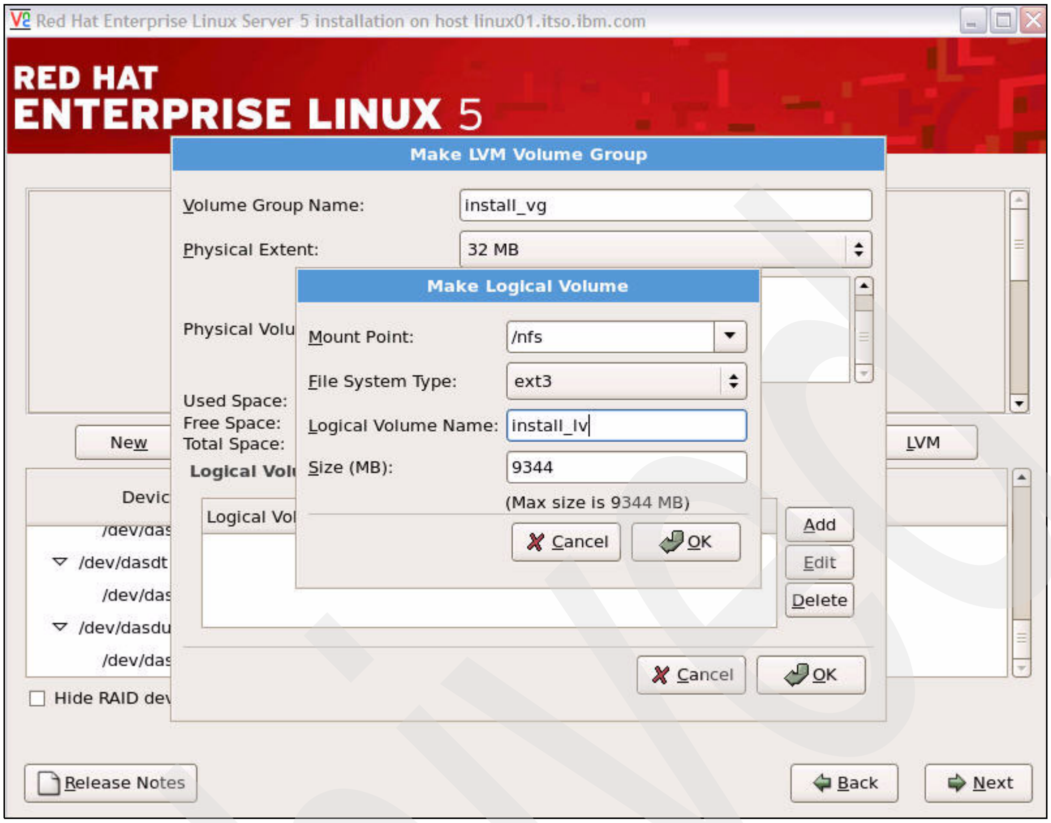
● Fill to maximum allowable size

*Figure 7-8 Adding the LVM partitions*

10. Repeat this step for `dasdc`, `dasdd`, and `dasde`.

11. Click the **LVM** button. Under Volume Group Name, enter the name **install_vg**.

12. Click the **Add** button on the right. The Make Logical Volume window should appear as shown in Figure 7-9 on page 106.



Red Hat Enterprise Linux Server 5 installation on host linux01.itso.ibm.com

Make LVM Volume Group

Volume Group Name: install_vg

Physical Extent: 32 MB

Make Logical Volume

Mount Point: /nfs

File System Type: ext3

Logical Volume Name: install_lv

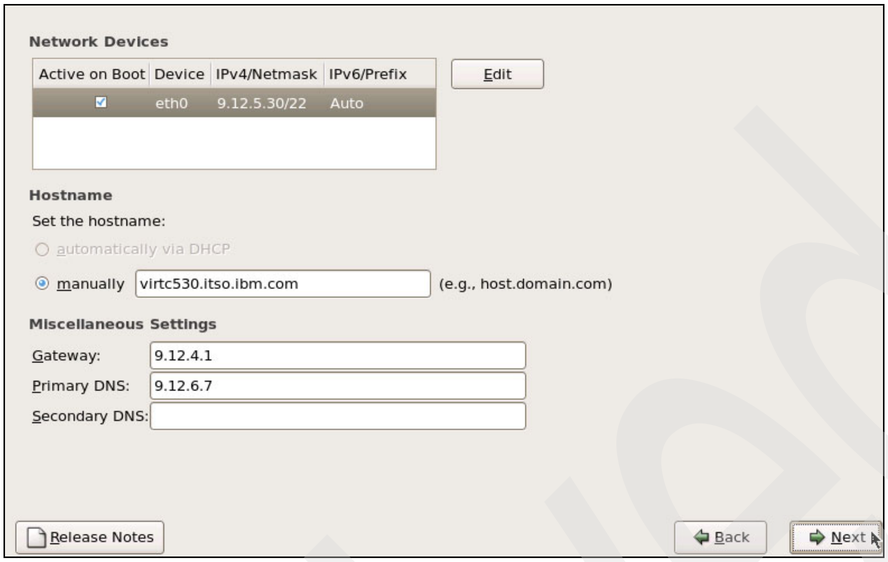
Size (MB): 9344

(Max size is 9344 MB)

*Figure 7-9 Configuring the /nfs logical volume*

13. Enter `install_lv` for the Logical Volume Name and `/nfs` for the Mount Point. Accept the maximum size, then click **OK**.
14. Click **OK** again.
15. Click **Next**. You will see a warning about missing swap a partition. You can safely ignore this because there will be VDISK swap defined later. Click **Yes**.

16. At the Network Configuration window, network settings have been automatically taken from the `LNXINST CONF-RH5` file, as shown in Figure 7-10. Click **Next**.



**Network Devices**

| Active on Boot | Device | IPv4/Netmask | IPv6/Prefix |
|---|---|---|---|
| ☑ | eth0 | 9.12.5.30/22 | Auto |

**Hostname**

Set the hostname:

○ automatically via DHCP

● manually: virtc530.itso.ibm.com (e.g., host.domain.com)

**Miscellaneous Settings**

Gateway: 9.12.4.1

Primary DNS: 9.12.6.7

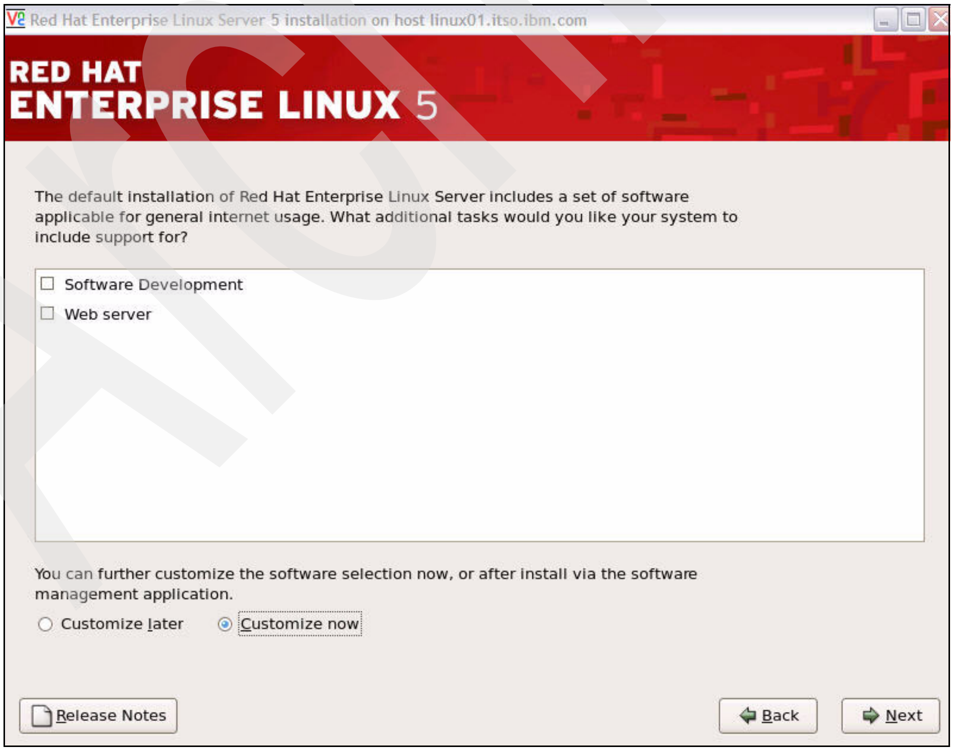
Secondary DNS:

*Figure 7-10 Configuring the network device*

17. At the Time Zone Selection window, highlight the nearest city in your time zone. Deselect System clock uses UTC and click **Next**.

18. Set the root password, then click **Next**.

19. At the package selection window, deselect all package groups, select **Customize Now** and click **Next**.



Red Hat Enterprise Linux Server 5 installation on host linux01.itso.ibm.com

The default installation of Red Hat Enterprise Linux Server includes a set of software applicable for general internet usage. What additional tasks would you like your system to include support for?

□ Software Development

□ Web server

You can further customize the software selection now, or after install via the software management application.

○ Customize later ● Customize now

*Figure 7-11 Customizing software packages*

20. Remove the majority of the package groups as follows:

   a. Under Desktop Environments, deselect **Gnome** so nothing is selected

   b. Under Applications, deselect all package groups

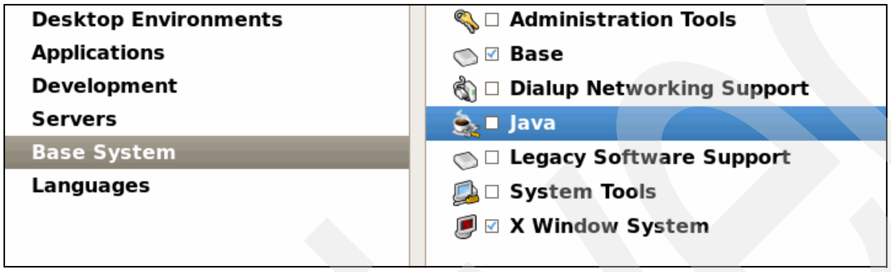
   c. Under Development, nothing should be selected

   d. Under Servers, deselect Printing Support so nothing is selected

   e. Under Base System, leave **Base** and **X Window System** selected and deselect everything else, as shown in Figure 7-12.

   f. Under Languages, select a package group if you need specific language support.

   g. Click **Next** and package dependencies will be determined.



Desktop Environments
Applications
Development
Servers
Base System
Languages

□ Administration Tools
☑ Base
□ Dialup Networking Support
□ Java
□ Legacy Software Support
□ System Tools
☑ X Window System

*Figure 7-12 Customizing packages*

21. Finally, click **Next** to begin the installation. This lasts from five to ten minutes, depending on network speeds.

22. When the installation is complete, click **Reboot**. The VNC session will end, but your system will not reboot.

## 7.1.7 Booting your new Linux system from disk

A minimal system should now be installed onto minidisk 100. Return to your z/VM 3270 session and IPL the newly installed system using the command **#CP IPL 100**.

```
  /mnt/sysimage/dev done
  /mnt/sysimage done
 you may safely reboot your system
 ==> #cp ipl 100
 CP IPL 100
 zIPL v1.3.2 interactive boot menu
   0. default (linux)
   1. linux
 Note: VM users please use '#cp vi vmsg <input>'
 Please choose (default will boot in 15 seconds):
 ...
```

Linux will boot after 15 seconds if you take no action. To boot immediately, issue the following command:

```
==> #cp vi vmsg 0
```

You system should continue to boot until a login prompt is presented. Start an SSH session into the master image as root. At this point, you can disconnect from the 3270 session:

```
==> #cp disc hold
```

## 7.2 Configuring the controller

Now that your controller is installed, it must be configured. The following steps are involved:

- Copying files to the controller
- Retiring the PC NFS server
- Configuring VDISK swap
- Configuring yum
- Adding additional RPMs
- Turning off unneeded services
- Configuring the VNC server
- Setting the system to halt on SIGNAL SHUTDOWN
- Turning on the NFS server
- Configuring SSH keys
- Inserting the vmcp module
- Rebooting the system
- Verifying the changes

### 7.2.1 Copying files to the controller

It is now time to copy the RHEL 5.2 install tree to the controller, along with other files associated with this book. First mount the directory `/nfs/rhel5.2/` on the PC NFS server over the directory `/mnt/`. In this example, the PC NFS server is at IP address `9.12.4.185`:

```
# mount <9.12.4.185>:/nfs/rhel5.2 /mnt
# ls /mnt
EULA             README-or.html          RELEASE-NOTES-ja.html
...
```

Now create a local directory of the same name and recursively copy the tree using the **`cp -a`** command:

```
# mkdir /nfs/rhel5.2
# cd /mnt
# cp -a * /nfs/rhel5.2
```

This command will take some time, perhaps 5 to 10 minutes, depending on network speed.

Unmount the RHEL 5.2 install tree and repeat the process to copy the files associated with this book:

```
# cd /
# umount /mnt
# mount <9.12.4.185>:/nfs/virt-cookbook-RH5.2 /mnt
# mkdir /nfs/virt-cookbook-RH5.2
# cd /mnt
# cp -a * /nfs/virt-cookbook-RH5.2
```

When the files are copied, unmount the `/mnt/` directory. Then view the files that you copied:

```
# umount /mnt/
# cd /nfs/virt-cookbook-RH5.2
# ls
clone-1.0-5.s390x.rpm  README.txt  vm/
```

The RPM `clone-1.0-5.s390x.rpm` contains files for use later in Chapter 9, “Configuring RHEL 5.2 for cloning” on page 129.

## 7.2.2 Retiring the PC NFS server

You have now copied all files related to this book to the controller. You should be in a position to retire your PC NFS server, if appropriate. The remainder of the book uses files located on the controller instead of the files on the PC NFS server.

## 7.2.3 Configuring VDISK swap

Recall that you were warned during the installation that you were not using any swap spaces. This can be confirmed with the **swapon -s** command which lists swap spaces:

```
# swapon -s
```

Receiving no output means there are no swap spaces. Linux must now be configured to use the VDISK swap spaces. Observe the first three DASD devices using the following **grep** command:

```
# grep FBA /proc/dasd/devices
0.0.0300(FBA ) at ( 94: 128) is dasdq : active at blocksize: 512, 524288 blocks, 256 MB
0.0.0301(FBA ) at ( 94: 132) is dasdr : active at blocksize: 512, 1048576 blocks, 512 MB
```

You should see that /dev/dasdq and /dev/dasdr are FBA devices, which signifies VDISKs. The **SWAPGEN EXEC** creates a single partition with a swap signature on each of the two VDISKs. Therefore, you can use /dev/dasdq1 and /dev/dasdr1 as swap devices. Make a backup copy of the file system table, /etc/fstab, then add the two swap spaces as follows:

```
# cd /etc
# cp fstab fstab.orig
# vi /etc/fstab
LABEL=/                 /                       ext3    defaults        1 1
/dev/install_vg/install_lv /nfs                 ext3    defaults        1 2
LABEL=/boot             /boot                   ext3    defaults        1 2
/dev/dasdq1             swap                    swap    defaults        0 0
/dev/dasdr1             swap                    swap    defaults        0 0
tmpfs                   /dev/shm                tmpfs   defaults        0 0
devpts                  /dev/pts                devpts  gid=5,mode=620  0 0
sysfs                   /sys                    sysfs   defaults        0 0
proc                    /proc                   proc    defaults        0 0
```

Save the file and turn on the swap spaces using the **swapon -a** command, which will read the /etc/fstab file and then verify that the swap spaces have been activated:

```
# swapon -a
# swapon -s
Filename        Type            Size    Used    Priority
/dev/dasdq1     partition       259956  0       -1
/dev/dasdr1     partition       519924  0       -2
```

## 7.2.4 Configuring yum

Now configure **yum** so it can install RPMs from local install tree. Create a file named rhel5.2.repo in the /etc/yum.repos.d directory:

```
# cd /etc/yum.repos.d
# vi rhel5.2.repo
[RHEL5.2]
name=Red Hat Enterprise Linux 5.2
baseurl=file:///nfs/rhel5.2/Server
```

Import the RPM key:

```
# cd /nfs/rhel5.2
# rpm --import RPM-GPG-KEY-redhat-release
```

> **Note:** Red Hat signs each RPM with a private GPG key, which is compared to your public key each time a package is installed. This method ensures that the RPM is a genuine, unaltered package. When installing an RPM, if you ever see a message similar to the following, it means that either the correct GPG key has not been imported, or the package itself has been altered by a third party:
>
> ```
> Header V3 DSA signature: NOKEY, key ID 897da07a
> ```

You are now ready to use **yum** to install or upgrade an RPM package. To install a package, use **yum install <packagename>**; yum will conveniently install the packages specified and automatically resolve dependencies for you. Do *not* specify the package version on the command line; specify only the package name.

### 7.2.5 Adding additional RPMs

Now that **yum** is configured, you will install some additional RPM packages onto the master image. You may choose to add additional RPMs, or may choose to omit some of the following:

**openmotif** This is a window manager that is used in conjunction with VNC, and is more usable than the default window manager.

**xinetd** This is a service that controls access to other services.

To add these two RPMs, use **yum -y install** command packages. Notice that **yum** has added multiple dependencies to the package list:

```
# yum -y install openmotif xinetd
Loading "rhnplugin" plugin
Loading "installonlyn" plugin
...
Installed: openmotif.s390 0:2.3.0-0.5.el5 openmotif.s390x 0:2.3.0-0.5.el5 xinetd.s390x
2:2.3.14-10.el5
Dependency Installed: expat.s390 0:1.95.8-8.2.1 fontconfig.s390 0:2.4.1-7.el5
freetype.s390 0:2.2.1-19.el5 libXft.s390 0:2.1.10-1.1 libXmu.s390 0:1.0.2-5 libXp.s390x
0:1.0.0-8.1.el5 libXp.s390 0:1.0.0-8.1.el5 libXrender.s390 0:0.9.1-3.1 libjpeg.s390
0:6b-37 libpng.s390 2:1.2.10-7.1.el5_0.1
Complete!
```

Now you have a good set of packages in the master image that will be cloned into virtual servers. You may choose to add more packages, or remove some packages.

**Important:** If you get an error of the following form, it means that there is a problem:

```
Resolving Dependencies
--> Populating transaction set with selected packages. Please wait.
---> Downloading header for openmotif to pack into transaction set.
media://1169877309.407096%233/openmotif-2.3.0-0.3.el5.s390x.rpm: [Errno 4] IOError:
<urlopen error unknown url type: media>
Trying other mirror.
Error: failed to retrieve openmotif-2.3.0-0.3.el5.s390x.rpm from RHEL5.2
error was [Errno 4] IOError: <urlopen error unknown url type: media>
```

The repository can be built to work around this problem. First, install the `createrepo` RPM:

```
# cd /nfs/rhel5.2/Server/
# rpm -ivh createrepo-0.4.4-2.fc6.noarch.rpm
warning: createrepo-0.4.4-2.fc6.noarch.rpm: Header V3 DSA signature: NOKEY, key ID
37017186
Preparing...                ########################################### [100%]
   1:createrepo             ########################################### [100%]
```

Rename the `repodata/` directory and create a new one using the **`createrepo`** command:

```
# mv repodata/ repodata.orig
# createrepo .
844/2495 -
jakarta-commons-collections-testframework-javadoc-3.1-6jpp.1.s390x.rp2495/2495 -
rsh-server-0.17-37.el5.s390x.rpm
Saving Primary metadata
Saving file lists metadata
Saving other metadata
```

The new repository should work for any **`yum`** operations, but if you ever want to reinstall with this tree, replace the `repodata/` with `repodata.orig/`.

### 7.2.6 Turning off unneeded services

A number of services are started in an RHEL 5.2 minimum system. To keep the controller as lean as possible, some of these can be turned off, as described in this section.

Turn off the following services using the **chkconfig** command:

```
# chkconfig cups off
# chkconfig iptables off
# chkconfig ip6tables off
# chkconfig auditd off
# chkconfig haldaemon off
# chkconfig atd off
# chkconfig kudzu off
# chkconfig mdmonitor off
# chkconfig rpcgssd off
# chkconfig rpcidmapd off
# chkconfig anacron off
# chkconfig mcstrans off
# chkconfig yum-updatesd off
```

**Note:** Only disable the **iptables** service if you are on a trusted network. Otherwise, you will need to configure **iptables** to allow network traffic for the VNC server and NFS, as well as any other services that require network access.

For more information about configuring **iptables** for NFS traffic, refer to the article located at:

http://www.redhat.com/magazine/010aug05/departments/tips_tricks/

You may choose to leave these services on, or turn others off. You can review which services are now configured to start in run level 3 with the following command:

```
# chkconfig --list | grep 3:on
autofs          0:off   1:off   2:off   3:on    4:on    5:on    6:off
avahi-daemon    0:off   1:off   2:off   3:on    4:on    5:on    6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
lvm2-monitor    0:off   1:on    2:on    3:on    4:on    5:on    6:off
messagebus      0:off   1:off   2:off   3:on    4:on    5:on    6:off
netfs           0:off   1:off   2:off   3:on    4:on    5:on    6:off
network         0:off   1:off   2:on    3:on    4:on    5:on    6:off
nfslock         0:off   1:off   2:off   3:on    4:on    5:on    6:off
portmap         0:off   1:off   2:off   3:on    4:on    5:on    6:off
readahead_early 0:off   1:off   2:on    3:on    4:on    5:on    6:off
restorecond     0:off   1:off   2:on    3:on    4:on    5:on    6:off
rhnsd           0:off   1:off   2:off   3:on    4:on    5:on    6:off
sendmail        0:off   1:off   2:on    3:on    4:on    5:on    6:off
sshd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
xfs             0:off   1:off   2:on    3:on    4:on    5:on    6:off
xinetd          0:off   1:off   2:off   3:on    4:on    5:on    6:off
```

### 7.2.7 Configuring the VNC server

Often, applications require a graphical environment. The Virtual Network Computing (VNC) server allows for a graphical environment to be set up easily by starting the **vncserver** service. RHEL 5.2 configures the VNC server using the /etc/sysconfig/vncservers configuration file. Add a line at the bottom of this file to specify the VNC user:

```
# vi /etc/sysconfig/vncservers
...
VNCSERVERS="1:root"
```

Set a VNC password using the **vncpasswd** command. This password will be needed to connect to the VNC server:

```
# vncpasswd
Password: <lnx4vm>
Verify: <lnx4vm>
```

First stop the firewall:

```
# service iptables stop
Flushing firewall rules: [  OK  ]
Setting chains to policy ACCEPT: filter [  OK  ]
Unloading iptables modules: [  OK  ]
```

Start the VNC server. This will create some initial configuration files under the /root/.vnc/ directory:

```
# service vncserver start
Starting VNC server: 1:root xauth:  creating new authority file /root/.Xauthority
New 'virtc530.itso.ibm.com:1 (root)' desktop is virtc530.itso.ibm.com:1
Creating default startup script /root/.vnc/xstartup
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/virtc530.itso.ibm.com:1.log
[  OK  ]
```

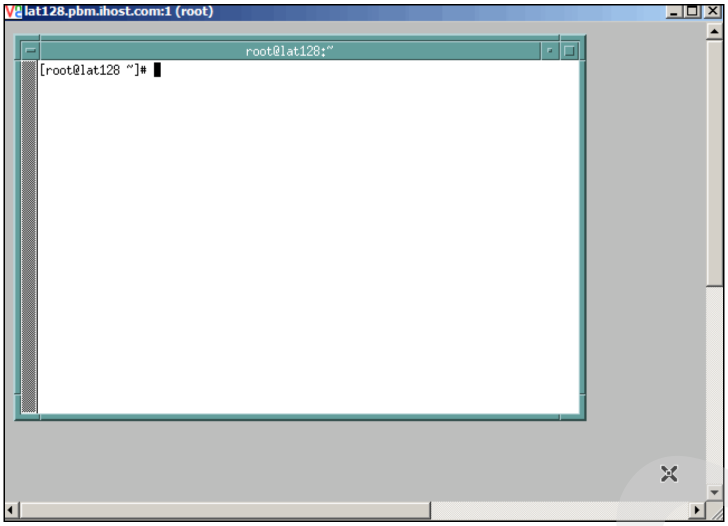
There is one more configuration to be done. Change from the Tiny window manger, **twm**, to the Motif window manager, **mwm**:

```
# cd /root/.vnc
# vi xstartup   // change last line
...
xsetroot -solid grey
vncconfig -iconic &
xterm -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
mwm &
```

Now restart the VNC server with the **service** command:

```
# service vncserver restart
Shutting down VNC server: 1:root [  OK  ]
Starting VNC server: 1:root
...
[  OK  ]
```

You can now use the VNC client to connect to the IP address of the controller with a `:1` appended. A sample session is shown in Figure 7-13 on page 115.



```
[root@lat128 ~]#
```

*Figure 7-13 VNC client session to the VNC server*

Note that the VNC server will not be started automatically across reboots. When you need a graphical environment, you can either start the **vncserver** process manually (which is recommended), or you can use **chkconfig** to enable automatic startup.

## 7.2.8 Setting the system to halt on SIGNAL SHUTDOWN

By default, RHEL 5.2 reboots when a Ctrl-Alt-Del key sequence is trapped. This key sequence is simulated by z/VM when it issues a **SIGNAL SHUTDOWN** command. Rather than rebooting, you want your system to halt (shut down).

Change this setting by changing **shutdown -r** to **shutdown -h** in the /etc/inittab file:

```
# cd /etc
# vi inittab    // change shutdown -r to shutdown -h
...
# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -h now
...
```

This change will be picked up when the system is rebooted.

## 7.2.9 Turning on the NFS server

The NFS server will be needed to export the RHEL 5.2 install tree and the files associated with this book to the other virtual servers. Enable NFS as follows:

```
# vi /etc/exports
/nfs/rhel5.2             *(ro,sync)
/nfs/virt-cookbook-RH5.2 *(ro,sync)
```

These two lines will cause NFS to export:

- The /nfs/rhel5.2/ directory, which contains the Red Hat Enterprise Linux 5.2 installation.
- The /nfs/virt-cookbook-RH5.2/ directory, which has the files associated with this book.

Set the NFS server to start at boot time and for this session.

```
# chkconfig nfs on
# service nfs start
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
```

You could test mounting the directories locally if you choose to.

## 7.2.10 Configuring SSH keys

SSH sessions are typically authenticated using passwords typed in from the keyboard. With SSH *key-based authentications*, sessions can be authenticated using public and private keys so that no password is needed. To accomplish this, the following must be true:

- The SSH server must have the client’s public key.
- The SSH client must send its private key.
- The keys must match.

SSH key-based authentication can be set up from the controller (client) to the virtual servers. If the master image has a copy of controller’s public key in the file `/etc/ssh/authorized_keys`, and the controller has a symbolic link to its private key in the file `/root/.ssh/id_dsa`, then key-based authentication will work to the cloned virtual servers.

Create a new DSA key in the directory `/etc/ssh/`:

```
# cd /etc/ssh
# ssh-keygen -t dsa -P "" -f id_dsa
Generating public/private dsa key pair.
Your identification has been saved in id_dsa.
Your public key has been saved in id_dsa.pub.
The key fingerprint is:
c1:98:6e:c1:48:52:80:56:98:be:67:aa:1d:d5:76:95 root@virtc530.itso.ibm.com
```

This creates a key pair where the file with the `.pub` suffix is the public key, and the other file is the private key. Note that the private key is only readable by `root`:

```
# ls -l id_dsa*
-rw------- 1 root root 668 Dec 22 13:47 id_dsa
-rw-r--r-- 1 root root 615 Dec 22 13:47 id_dsa.pub
```

We copy these files to the golden image in 8.2.8, “Configuring SSH keys” on page 127.

Make a symbolic link to the private key, `id_dsa`, in the new directory `/root/.ssh/`:

```
# mkdir /root/.ssh
# chmod 700 /root/.ssh
# cd /root/.ssh
# ln -s /etc/ssh/id_dsa
```

## 7.2.11 Inserting the vmcp module

To issue CP commands, the vmcp module is needed. By default, it is not loaded at boot time. One way to load this module at boot time is to add the **`modprobe vmcp`** command, which will insert the module, to the file `/etc/rc.d/rc.local` which is run at boot time:

```
# cd /etc/rc.d
# vi rc.local   // add one line
```

```
...
touch /var/lock/subsys/local
modprobe vmcp
```

The **vmcp** command will now be available after the next reboot.

### 7.2.12 Rebooting the system

Reboot the system to test the changes:

```
# reboot
```

After your system comes back in a couple of minutes, start a new SSH session to the controller.

### 7.2.13 Verifying the changes

You are now finished customizing the Linux controller. SSH back into the controller and check a few settings. Test the **vmcp** command with a CP command such as **QUERY NAMES**:

```
# vmcp q n
FTPSERVE - DSC , DTCVSW2  - DSC , DTCVSW1  - DSC , TCPIP    - DSC
OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
LNXINST  - DSC
VSM     - TCPIP
```

Confirm that both of your swap spaces are operational:

```
# swapon -s
Filename        Type            Size    Used    Priority
/dev/dasdq1    partition       259956  0       -1
/dev/dasdr1    partition       519924  0       -2
```

Verify that the NFS server is running:

```
# service nfs status
rpc.mountd (pid 6776) is running...
nfsd (pid 6770 6769 6768 6767 6766 6765 6764 6763) is running...
rpc.rquotad (pid 6748) is running...
```

# 8

# Installing and configuring RHEL 5.2

**Note:** Before proceeding with this chapter, you must complete the tasks described in Chapter 4, “Installing and configuring z/VM” on page 27, Chapter 5, “Servicing z/VM” on page 71, and Chapter 6, “Configuring a Network File System server for RHEL 5.2” on page 85.

In this chapter, you will install the copy of Linux which will be cloned. This is referred to as the *golden image*. This should be as “lean” as possible so as to be a generic virtual server and to fit comfortably on two 3390-3 DASDs.

You will perform the following tasks:

- Install the golden image
- Configure the golden image

## 8.1 Installing the golden image

In this section you will install the RHEL 5.2 golden image onto the user ID `RHEL52`.

### 8.1.1 Creating the user ID RHEL52

In this section you will define the `RHEL52` user ID to z/VM.

1. Logon to `MAINT` and edit the `USER DIRECT` file:

```
==> x user direct c
```

2. Go to the bottom of the file and add the definition for a new user ID named `RHEL52`. This user ID is given class G privilege only. Be sure to replace the volume labels (`MMD956` and `MMD957`, in this example) with the labels of your DASD:

```
USER RHEL52 LNX4VM 256M 1G BDEG
  INCLUDE LNXDFLT
  OPTION LNKNOPAS APPLMON
  MDISK 100 3390 0001 3338 <MMD955> MR LNX4VM LNX4VM LNX4VM
  MDISK 101 3390 0001 3338 <MMD956> MR LNX4VM LNX4VM LNX4VM
*
```

This Linux user ID will have the minidisks and virtual disks (VDISKs) listed in Table 8-1.

*Table 8-1 Minidisks to be defined*

| Minidisk | Description |
|---|---|
| 100-101 | Minidisks used to create the /boot file system, plus an LVM containing the other file systems of the Linux golden image. |
| 300-301 | These are virtual disk (VDISK) swap spaces that are not defined in the `USER DIRECT` file, but instead defined by calls to the **`SWAPGEN EXEC`** in the user's **`PROFILE EXEC`** so that when the user ID logs on, the VDISKs are created. |

3. Go back to the top of the file and search for string `USER $ALLOC$`. Add cylinder 0 of each of the two new volumes to this dummy user ID so that they do not show up as gaps in the `USER DISKMAP` report file:

```
====> top
====> /user $alloc$
USER $ALLOC$  NOLOG
 MDISK A01 3390 000 001 <MVD850> R
 MDISK A02 3390 000 001 <MVD853> R
 MDISK A03 3390 000 001 <MVD854> R
 MDISK A04 3390 000 001 <MMD857> R
 MDISK A05 3390 000 001 <MMD856> R
 MDISK A06 3390 000 001 <MMD952> R
 MDISK A07 3390 000 001 <MMD953> R
 MDISK A08 3390 000 001 <MMD954> R
 MDISK A09 3390 000 001 <MMD955> R
 MDISK A0A 3390 000 001 <MMD956> R
 ...
====> file
```

4. Run **`DISKMAP`** to check for overlaps and gaps. You should only see the single 501 cylinder gap.

```
==> diskmap user
==> x user diskmap
```

```
====> all /gap/|/overlap/
--------------------  4  line(s) not displayed --------------------
                                            0         500       501     GAP
--------------------  322  line(s) not displayed --------------------
====> quit
```

5. When the disk layout is correct, run **DIRECTXA** to bring the changes online:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 5 RELEASE 3.0
EOJ DIRECTORY UPDATED AND ON LINE
```

You have now defined the user ID that will be the master Linux image.

## 8.1.2 Adding RHEL52 to the AUTOLOG1 PROFILE EXEC

The new Linux ID you defined needs access to the VSWITCH. Just as with the LNXINST user, a **SET VSWITCH** command with the **GRANT** parameter will now be added to the AUTOLOG1 **PROFILE EXEC**. Also, an **XAUTOLOG** statement is added so that the RHEL52 user ID is automatically logged on at z/VM IPL time.

Link and access the AUTOLOG1 191 disk as read/write and edit the file **PROFILE EXEC**. Add the RHEL52 user ID to the section that grants access to the VSWITCH. Do not add RHEL52 to the **XAUTOLOG** section, because this Linux user ID will not normally be logged on:

```
==> link autolog1 191 1191 mr
==> acc 1191 f
==> x profile exec f   // add two lines
/***************************/
/*  Autolog1 Profile Exec  */
/***************************/
'cp xautolog tcpip'                 /* start up TCPIP */
'CP XAUTOLOG DTCVSW1'               /* start VSWITCH controller 1 */
'CP XAUTOLOG DTCVSW2'               /* start VSWITCH controller 2 */
'cp set pf12 ret'                   /* set the retrieve key */
'cp set mdc stor 0m 128m'           /* Limit minidisk cache in CSTOR */
'cp set mdc xstore 0m 0m'           /* Disable minidisk cache in XSTOR */
'cp set srm storbuf 300% 250% 200%' /* Overcommit memory */
'cp set signal shutdown 300'        /* Allow guests 5 min to shut down */

/* Grant access to VSWITCH for each Linux user */
'cp set vswitch vsw1 grant lnxinst'
'cp set vswitch vsw1 grant rhel52'

/* XAUTOLOG each Linux user that should be started */
'cp xautolog lnxinst'

'cp logoff'                          /* logoff when done */
====> file
```

These changes will not take effect until the next IPL, so you must grant this user ID access to the VSWITCH for this z/VM session:

```
==> set vswitch vsw1 grant rhel52
Command complete
```

### 8.1.3 Preparing RHEL52 bootstrap files

Now that the RHEL52 user is defined, you must create the PARM and CONF configuration files used by the RHEL 5.2 installer. To save time, copy the LNXINST PARM-RH5 and LNXINST CONF-RH5 files, then make the necessary changes.

1. Now in your 3270 session, logoff from MAINT and logon to LNXMAINT.
2. The files LNXINST PARM-RH5, LNXINST CONF-RH5, and RHEL52 EXEC should exist on the LNXMAINT 192 (D) disk as they were copied in 4.7.5, “Copying files associated with this book to LNXMAINT” on page 58. Copy these files to new files named RHEL52 PARM-RH5 and RHEL52 CONF-RH5:

   ```
   ==> copy lnxinst parm-rh5 d rhel52 = =
   ==> copy lnxinst conf-rh5 d rhel52 = =
   ```

3. Change the CMSCONFFILE variable in the PARM-RH5 file to point to the new CONF file:

   ```
   CMSCONFFILE=RHEL52.CONF-RH5
   ```

4. Change the appropriate networking values in the RHEL52 CONF-RH5 configuration file. Modify at least the fields shown in bold font and perhaps others; refer to the worksheet in 2.7.4, “Linux user ID worksheet” on page 18, for guidance.

   Before:

   ```
   DASD=100-10f,300-30f
   HOSTNAME=<virtc530.itso.ibm.com>
   NETTYPE=qeth
   IPADDR=<9.12.5.30>
   SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
   NETWORK=<9.12.4.0>
   NETMASK=255.255.255.0
   SEARCHDNS=<itso.ibm.com>
   BROADCAST=<9.12.4.255>
   GATEWAY=<9.12.4.1>
   DNS=<9.12.6.7>
   MTU=1500
   PORTNAME=DONTCARE
   LAYER2=0
   VSWITCH=1
   ```

   After:

   ```
   ==> x rhel52 conf-rh5
   DASD=100-10f,300-30f
   HOSTNAME=<virtc529.itso.ibm.com>
   NETTYPE=qeth
   IPADDR=<9.12.5.29>
   SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
   NETWORK=<9.12.4.0>
   NETMASK=255.255.255.0
   SEARCHDNS=<itso.ibm.com>
   BROADCAST=<9.12.5.255>
   GATEWAY=<9.12.4.1>
   DNS=<9.12.6.7>
   MTU=1500
   PORTNAME=DONTCARE
   LAYER2=0
   VSWITCH=1
   ```

5. Save your changes with the **FILE** subcommand.

Now you are ready to start the golden image installation.

## 8.1.4 Installing RHEL 5.2 to the golden image

Install Linux again as described in 7.1.4, “Beginning the Linux installation” on page 97. However, because the controller is now up and running, install RHEL 5.2 using the installation tree exported using NFS from the controller.

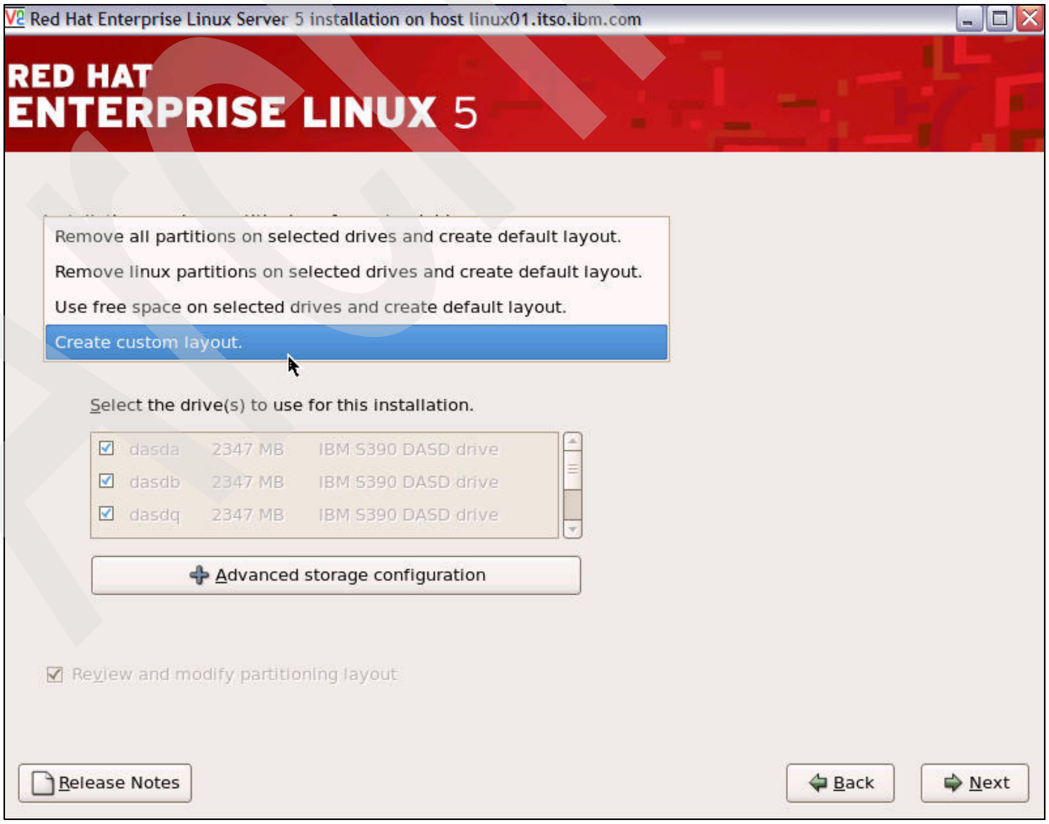
You follow the same steps as when installing Linux to the controller, except that you will create a different disk layout. If necessary, review 7.1.6, “Stage 2 of the RHEL 5.2 installation” on page 101 for the steps leading up to the Disk Partitioning section.

At the Disk Partitioning window, create an LVM containing separate file systems mounted at `/tmp`, `/opt`, `/var`, `/usr, and /`. Table 8-2 lists the recommended logical volume layout and sizes to be used for the golden image.

*Table 8-2 LVM logical volume layout*

| Mount point | Logical Volume Name | Size (MB) |
|---|---|---|
| /tmp | tmp_lv | 384 |
| /opt | opt_lv | 384 |
| /var | var_lv | 384 |
| /usr | usr_lv | 1600 |
| / | root_lv | 512 |

At the Disk Partitioning section, you should see the window that allows you to partition disks as shown in Figure 8-1.


Red Hat Enterprise Linux Server 5 installation on host linux01.itso.ibm.com
RED HAT
ENTERPRISE LINUX 5
Advanced storage configuration
Release Notes
Back
Next

Remove all partitions on selected drives and create default layout.
Remove linux partitions on selected drives and create default layout.
Use free space on selected drives and create default layout.
Create custom layout.

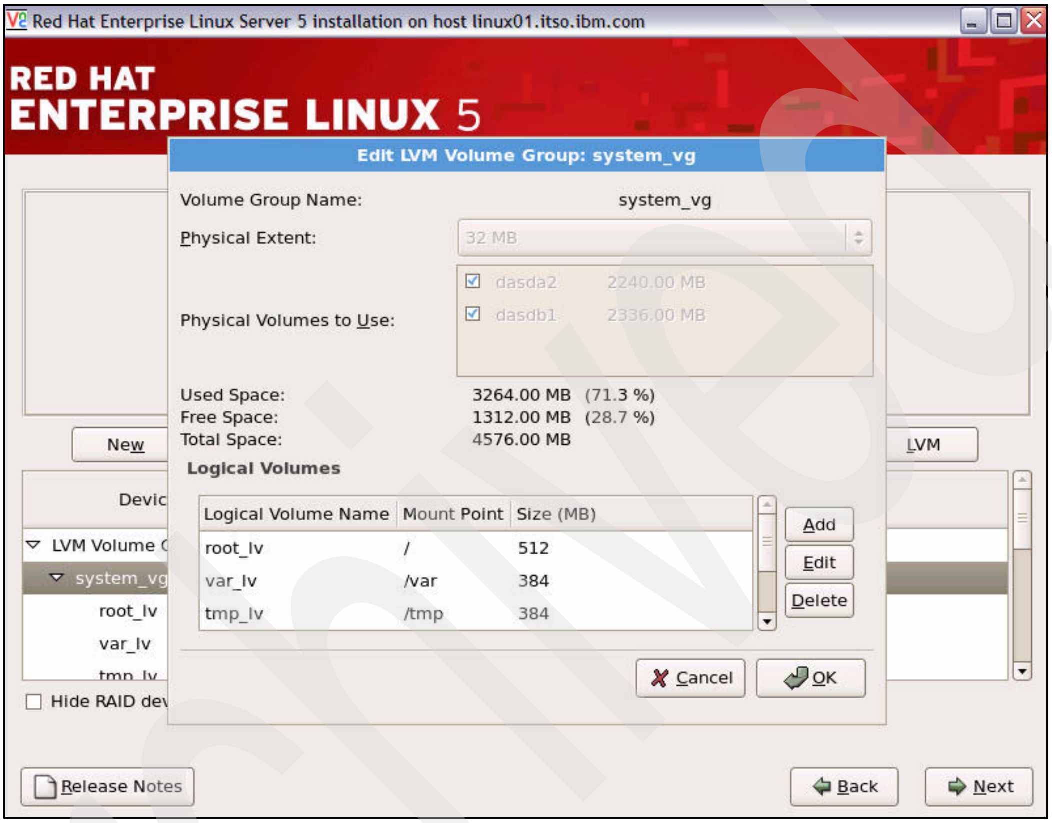
Select the drive(s) to use for this installation.

| | | | |
|---|---|---|---|
| ☑ | dasda | 2347 MB | IBM S390 DASD drive |
| ☑ | dasdb | 2347 MB | IBM S390 DASD drive |
| ☑ | dasdq | 2347 MB | IBM S390 DASD drive |

☑ Review and modify partitioning layout

*Figure 8-1 Choosing custom layout*

1. Select **Choose custom layout.** from the dropdown menu, then click **Next**.
2. Create the **/boot** file system on `dasda` with the default size of 100 MB. Use the remaining disk space on `dasda` to create an LVM. Then, use all of disk space on `dasdb` as an LVM physical volume.
3. Click the **LVM** button, and create the logical volumes as specified in Table 8-2 on page 123. The LVM setup should look similar to Figure 8-2.



Red Hat Enterprise Linux Server 5 installation on host linux01.itso.ibm.com

Edit LVM Volume Group: system_vg

Volume Group Name: system_vg

Physical Extent: 32 MB

Physical Volumes to Use:
☑ dasda2 2240.00 MB
☑ dasdb1 2336.00 MB

Used Space: 3264.00 MB (71.3 %)
Free Space: 1312.00 MB (28.7 %)
Total Space: 4576.00 MB

Logical Volumes

| Logical Volume Name | Mount Point | Size (MB) |
|---|---|---|
| root_lv | / | 512 |
| var_lv | /var | 384 |
| tmp_lv | /tmp | 384 |

*Figure 8-2 Disk partitioning window*

4. The remainder of the installation is identical to the controller. When finished, click **Next** and the VNC viewer session will end.
5. Back on your 3270 session, IPL minidisk 100 to boot the golden image.

```
==> #cp ipl 100
```

Start an SSH session to the golden image. You may see a warning from PuTTY about a `POTENTIAL SECURITY BREACH`. This is expected because a new set of SSH keys were generated for the same IP address. Click **Yes** to begin the session.

## 8.1.5 Verifying the installation

Verify some settings using the **mount** and **df -h** commands. You should see output similar to the following:

```
# mount
/dev/dasda1 on /boot type ext3 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
```

```
/dev/mapper/system_vg-root_lv on / type ext3 (rw)
/dev/mapper/system_vg-usr_lv on /usr type ext3 (rw)
/dev/mapper/system_vg-opt_lv on /opt type ext3 (rw)
/dev/mapper/system_vg-tmp_lv on /tmp type ext3 (rw)
/dev/mapper/system_vg-var_lv on /var type ext3 (rw)
tmpfs on /dev/shm type tmpfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)

# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1            97M   12M   81M  13% /boot
/dev/mapper/system_vg-root_lv
                      496M  228M  244M  49% /
/dev/mapper/system_vg-usr_lv
                      1.6G  926M  545M  63% /usr
/dev/mapper/system_vg-opt_lv
                      372M   17M  337M   5% /opt
/dev/mapper/system_vg-tmp_lv
                      372M   17M  337M   5% /tmp
/dev/mapper/system_vg-var_lv
                      372M   66M  288M  19% /var
tmpfs                 250M     0  250M   0% /dev/shm
```

This shows that the root file system is about half full.

## 8.2 Configuring the golden image

Customize the golden image as much as possible before cloning. The following high level steps are recommended, although you may add or omit some steps:

- Configuring swap spaces
- Configuring automount of install tree
- Configuring yum for online updates
- Adding additional RPMs
- Turning of unneeded services
- Configuring the VNC server
- Setting system to halt on SIGNAL SHUTDOWN
- Configuring SSH keys
- Rebooting the system
- Verifying the changes

### 8.2.1 Configuring swap spaces

Turn on the two swap spaces just as you did in 7.2.3, “Configuring VDISK swap” on page 110.

### 8.2.2 Configuring automount of install tree

You will now configure the Linux automount service to mount the installation tree on demand. The automounter will automatically mount a remote directory when it is accessed, and automatically unmount it after a period of inactivity.

Make a backup copy of the file /etc/auto.master, then add the following line at the bottom:

```
# cd /etc
# cp auto.master auto.master.orig
# vi /etc/auto.master   // add one line at the bottom
...
#
+auto.master
/nfs    /etc/auto.controller
```

The new line specifies that the file system mounted beneath the directory /nfs/ will be configured in the file /etc/auto.controller. Now create the file /etc/auto.controller, and add one line which points to the RHEL 5.2 install tree that is exported from the controller:

```
# vi /etc/auto.controller
rhel5.2     -ro,hard,intr   <9.12.5.30>:/nfs/rhel5.2
```

This line specifies that beneath /nfs/ (in auto.master), when the directory rhel5.2/ (field 1) is accessed, the automounter will use the specified options (field 2) to mount the directory (field 3).

Create the /nfs/ directory. Restart the **autofs** service to pick up the new configuration. Then list the contents of the /nfs/rhel5.2/ directory. Even though this directory does not exist as a local file system, it is automatically mounted when referenced:

```
# mkdir /nfs
# service autofs restart
Stopping automount: [  OK  ]
Starting automount: [  OK  ]
# ls /nfs/rhel5.2
EULA            README-or.html         RELEASE-NOTES-ja.html
eula.en_US      README-pa.html         RELEASE-NOTES-ko.html
...
```

## 8.2.3 Configuring yum for online updates

You will now configure **yum** so it can install RPMs from the automounted install tree. Note that the configuration is identical to the controller because in both instances, the install tree is in the directory /nfs/rhel5.2/. However, on the controller this directory is local, but on the golden image (and later, the clones), the directory is automounted. Create a file named rhel5.2.repo in the /etc/yum.repos.d directory:

```
# cd /etc/yum.repos.d
# vi rhel5.2.repo
[RHEL5.2]
name=Red Hat Enterprise Linux 5.2
baseurl=file:///nfs/rhel5.2/Server
```

You must import the RPM GPG key so that **yum** knows you are installing official Red Hat packages. The Red Hat GPG key is located in the install tree. Import the key with the following command:

```
# rpm --import /nfs/rhel5.2/RPM-GPG-KEY-redhat-release
```

## 8.2.4 Adding additional RPMs

We recommend that you install the same packages as you did on the controller; refer to 7.2.5, “Adding additional RPMs” on page 111. Use the **yum -y install** command to install the openmotif and xinetd packages from the remote RPM repository:

```
# yum -y install openmotif xinetd
```

## 8.2.5 Turning off unneeded services

As with the golden image, follow the steps in 7.2.6, “Turning off unneeded services” on page 112. Following is a summary:

```
# chkconfig cups off
# chkconfig iptables off
# chkconfig ip6tables off
# chkconfig auditd off
# chkconfig haldaemon off
# chkconfig atd off
# chkconfig kudzu off
# chkconfig mdmonitor off
# chkconfig rpcgssd off
# chkconfig rpcidmapd off
# chkconfig anacron off
# chkconfig mcstrans off
# chkconfig yum-updatesd off
```

## 8.2.6 Configuring the VNC server

The VNC server is configured the same way as on the controller. Follow the same steps as described in 7.2.7, “Configuring the VNC server” on page 113.

## 8.2.7 Setting system to halt on SIGNAL SHUTDOWN

Again, RHEL 5.2 reboots when a Ctrl-Alt-Del key sequence is trapped. This key sequence is simulated by z/VM when a **SIGNAL SHUTDOWN** command is issued. Rather than rebooting, you want your system to halt (shut down). Change this setting by changing **shutdown -r** to **shutdown -h** in the /etc/inittab file:

```
# cd /etc
# vi inittab    // change shutdown -r to shutdown -h
...
# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -h now
...
```

This change will be picked up when the system is rebooted.

## 8.2.8 Configuring SSH keys

Recall that you generated SSH keys on the controller in 7.2.10, “Configuring SSH keys” on page 116. Now it is time to copy these keys from the controller to the golden image. Create a new directory on the golden image where the public key will be copied:

```
# mkdir /root/.ssh
# chmod 700 /root/.ssh
```

Copy the public key to the name `authorized_keys` using the secure copy command **`scp`**:

```
# scp virtc530.itso.ibm.com:/etc/ssh/id_dsa.pub /root/.ssh/authorized_keys
```

This allows the controller to initiate an encrypted SSH connection to the Linux server without the need to type the root password.

### 8.2.9 Rebooting the system

Now **`reboot`** to test your changes:

```
# reboot
Broadcast message from root (pts/0) (Sun Nov 19 08:57:32 2006):

The system is going down for reboot NOW!
```

### 8.2.10 Verifying the changes

You have now completed the customization of the master Linux image. When the system comes back up, verify the changes that you made.

1. SSH back into the controller and check a few settings.
2. Use the **`df`** command to display your file systems (your output may differ):

```
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/system_vg-root_lv
                      496M  237M  235M  51% /
/dev/mapper/system_vg-tmp_lv
                      372M   17M  337M   5% /tmp
/dev/mapper/system_vg-opt_lv
                      372M   17M  337M   5% /opt
/dev/mapper/system_vg-usr_lv
                      1.6G  949M  522M  65% /usr
/dev/mapper/system_vg-var_lv
                      372M   66M  288M  19% /var
/dev/dasda1            97M   12M   81M  13% /boot
tmpfs                 250M     0  250M   0% /dev/shm
```

3. Confirm that both of your swap spaces are operational:

```
# swapon -s
Filename      Type            Size    Used    Priority
/dev/dasdq1   partition       259956  0       -1
/dev/dasdr1   partition       519924  0       -2
```

   The device with higher priority will be used first; if full, the device next in priority is used. If using defaults, the first VDISK (`300`, `/dev/dasdq1`) gets priority -1 and the second VDISK (`301`, `/dev/dasdr1`) with priority -2 is used after the first is full. You can also specify **`pri=nn`** in `/etc/fstab` for swap devices.

4. You may choose to confirm other settings.

   At this point you have successfully installed the golden image. This image will normally be shut down or quiesced.

   You are now ready to clone the golden image to a new virtual server.

9

# Configuring RHEL 5.2 for cloning

At this point you have completed the install of LNXINST, the Linux controller, and RHEL52, the golden image. The controller must be up and running. In this chapter, you perform the following steps:

- Formatting DASD for minidisks
- Cloning a virtual server manually
- Cloning one new virtual server
- Cloning three more virtual servers
- Reviewing system status

## 9.1 Formatting DASD for minidisks

In 4.6.2, “Formatting DASD for minidisks” on page 50, DASD was formatted to become minidisks for the controller and the golden image. The **CPFMTXA** command can be used to format one DASD at a time, but the **CPFORMAT EXEC** is a wrapper around **CPFMTXA** that allows the formatting of multiple DASD.

To have access to enough DASD to define four more user IDs with two 3390-3 volumes each, seven more volumes must be formatted (eight are needed, but one volume, D957 in this example, is still available). In the examples used in this book, the seven volumes that will be used are at addresses DA51-DA57. Consult your worksheets on 2.7.2, “z/VM DASD worksheet” on page 17, for guidance.

1. Logon to a 3270 session as MAINT.
2. Query the devices that will be used for the remaining Linux user IDs.

```
==> q <da51-da57>
DASD DA51 NWDA51 , DASD DA52 NWDA52 , DASD DA53 NWDA53 , DASD DA54 NWDA54
DASD DA55 NWDA55 , DASD DA56 NWDA56 , DASD DA57 NWDA57
```

3. Attach the eight volumes that will be used for the controller, the common CMS disk and the golden image. In this example it is the DASD at addresses:

```
==> att <da51-da57> *
DA51-DA57 ATTACHED TO MAINT
```

4. Invoke the **CPFORMAT** command against these volumes using the parameter **as perm**:

```
==> cpformat <da51-da57> as perm

Format the following DASD:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA51 MAINT    DA51 3390  NWDA51 DA51          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA52 MAINT    DA52 3390  NWDA52 DA52          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA53 MAINT    DA53 3390  NWDA53 DA53          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA54 MAINT    DA54 3390  NWDA54 DA54          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA55 MAINT    DA55 3390  NWDA55 DA55          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA56 MAINT    DA56 3390  NWDA56 DA56          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA57 MAINT    DA57 3390  NWDA57 DA57          0       3339

WARNING - this will destroy data!
ARE YOU SURE you want to format the DASD as PERM space (y/n)?
y
...
DA51-DA57 DETACHED
DA51-DA57 ATTACHED TO MAINT

DASD status after:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    DA51 MAINT    DA51 3390  MMDA51 DA51          0       3339
MAINT    DA52 MAINT    DA52 3390  MMDA52 DA52          0       3339
MAINT    DA53 MAINT    DA53 3390  MMDA53 DA53          0       3339
MAINT    DA54 MAINT    DA54 3390  MMDA54 DA54          0       3339
MAINT    DA55 MAINT    DA55 3390  MMDA55 DA55          0       3339
MAINT    DA56 MAINT    DA56 3390  MMDA56 DA56          0       3339
MAINT    DA57 MAINT    DA57 3390  MMDA57 DA57          0       3339
```

Detach the seven volumes from `MAINT` using the **`DETACH`** command:

```
==> det <da51-da57>
DA51-DA57 DETACHED
```

Attach the newly formatted DASD to `SYSTEM` so they can be used for minidisks:

```
==> att <da51-da57> system
DASD DA51 ATTACHED TO SYSTEM MMDA51
DASD DA52 ATTACHED TO SYSTEM MMDA52
DASD DA53 ATTACHED TO SYSTEM MMDA53
DASD DA54 ATTACHED TO SYSTEM MMDA54
DASD DA55 ATTACHED TO SYSTEM MMDA55
DASD DA56 ATTACHED TO SYSTEM MMDA56
DASD DA57 ATTACHED TO SYSTEM MMDA57
```

The seven DASD volumes will now be available to be used for minidisks in the `USER DIRECT` file. They will also be available after the next IPL because their new labels match the pattern specified by the `User_Volume_Include MM*` statement in the `SYSTEM CONFIG` file.

### 9.1.1 Defining a new user ID for a virtual server

Here you will define a new user ID, `LINUX01`, in z/VM and clone the golden image to it.

1. Logon to `MAINT` and edit the `USER DIRECT` file to add more Linux IDs.

```
==> x user direct c
```

2. Go to the bottom of the file and add the following 5 lines. In this example, the user ID will be `LINUX01` with a password of `LNX4VM`. It will default to have 256 MB of memory, but can be set up to 1 GB. It will have only G permission (General user). It will have two 3338-cylinder (about 2.2 GB each) minidisks for the RHEL installation. In this example, they are located at device addresses `D957` and `DA51`, which were formatted and given a label of `MMD957` and `MMDA51`, respectively:

```
USER LINUX01 LNX4VM 256M 1G G
 INCLUDE LNXDFLT
 OPTION APPLMON
 MDISK 100 3390 0001 3338 <MMD957> MR LNX4VM LNX4VM LNX4VM
 MDISK 101 3390 0001 3338 <MMDA51> MR LNX4VM LNX4VM LNX4VM
```

3. Add the new volumes to the `$ALLOC$` user ID so cylinder 0 will not show up in the disk map as a gap. Save your changes with the **`FILE`** subcommand:

```
====> top
====> /alloc
USER $ALLOC$  NOLOG
 MDISK A01 3390 000 001 <MVD850> R
 MDISK A02 3390 000 001 <MVD853> R
 MDISK A03 3390 000 001 <MVD854> R
 MDISK A04 3390 000 001 <MMD857> R
 MDISK A05 3390 000 001 <MMD856> R
 MDISK A06 3390 000 001 <MMD952> R
 MDISK A07 3390 000 001 <MMD953> R
 MDISK A08 3390 000 001 <MMD954> R
 MDISK A09 3390 000 001 <MMD955> R
 MDISK A0A 3390 000 001 <MMD956> R
 MDISK A0B 3390 000 001 <MMD957> R
 MDISK A0C 3390 000 001 <MMDA51> R
====> file
```

4. Again check for gaps and overlaps. You can use the **ALL** subcommand with the logical OR operator (|) to check for both strings. You should see only one 501 cylinder gap.

```
==> diskmap user
==> x user diskmap
====> all /gap/|/overlap/
--------------------  4  line(s) not displayed --------------------
                                            0         500         501     GAP
--------------------  368  line(s) not displayed --------------------
====> quit
```

5. Bring the changes online with the **DIRECTXA** command:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 5 RELEASE 3.0
EOJ DIRECTORY UPDATED AND ON LINE
```

The new Linux user ID has now been defined.

## 9.1.2 Adding LINUX01 to AUTOLOG1's PROFILE EXEC

The new Linux ID you defined needs access to the VSWITCH. A **SET VSWITCH** command with the **GRANT** parameter can be added to the AUTOLOG1 PROFILE EXEC to do this. Also, an **XAUTOLOG** statement can be added if the user ID is automatically logged on at z/VM IPL time.

Link and access the AUTOLOG1 191 disk read/write and edit the file PROFILE EXEC. Add LINUX01 to the sections that grant access to the VSWITCH and that **XAUTOLOG** the Linux user IDs:

```
==> link autolog1 191 1191 mr
==> acc 1191 f
==> x profile exec f   // add two lines
/***************************/
/*  Autolog1 Profile Exec  */
/***************************/
'cp xautolog tcpip'                 /* start up TCPIP */
'CP XAUTOLOG DTCVSW1'               /* start VSWITCH controller 1 */
'CP XAUTOLOG DTCVSW2'               /* start VSWITCH controller 2 */
'cp set pf12 ret'                   /* set the retrieve key */
'cp set mdc stor 0m 128m'           /* Limit minidisk cache in CSTOR */
'cp set mdc xstore 0m 0m'           /* Disable minidisk cache in XSTOR */
'cp set srm storbuf 300% 250% 200%' /* Overcommit memory */
'cp set signal shutdown 300'        /* Allow guests 5 min to shut down */

/* Grant access to VSWITCH for each Linux user */
'cp set vswitch vsw1 grant lnxinst'
'cp set vswitch vsw1 grant rhel52'
'cp set vswitch vsw1 grant linux01'

/* XAUTOLOG each Linux user that should be started */
'cp xautolog lnxinst'
'cp xautolog linux01'

'cp logoff'                          /* logoff when done */
====> file
```

These changes will not take effect until the next IPL, so you must grant this user ID access to the VSWITCH for this z/VM session. This is done as follows:

```
==> set vswitch vsw1 grant linux01
Command complete
```

## 9.2 Cloning a virtual server manually

Before using the **clone** script to clone a server, it is recommended that you clone a server manually to better understand the process.

There are many ways to clone Linux under z/VM. The steps in this section are just one way to do it. The following assumptions are made based on what you have done so far:

- The source user ID, RHEL52 in this example, has a root file system on LVM located on minidisks 100-101.
- The target user ID, LINUX01 in this example, has identically-sized mindisks 100-101.
- The **vmcp** command is available to issue z/VM CP commands.
- The z/VM **FLASHCOPY** command can be used but if you do not have that support, the Linux **dd** command will work.

Given these assumptions, here is one set of steps that can be used to clone a system:

1. Link the source disks read-only.
2. Link the target disks read/write.
3. Copy the source to the target disk using **FLASHCOPY** or the Linux **dd** command.
4. Bring the newly copied LVM online.
5. Mount the newly copied root file system.
6. Modify the networking information about the target system.
7. Detach the source disks.
8. Detach the target disks.
9. IPL the target system.
10. Modify the SSH keys on the target system.

### Link the source and target disks

Start an SSH session to the controller as root.

The source disks, RHEL52 100-101, are linked read-only as virtual devices 1100 and 1101 using the CP **LINK** command:

```
# vmcp link rhel52 100 1100 rr
# vmcp link rhel52 101 1101 rr
```

The target disks, LINUX01 100-101, are linked multi-read (read/write if no other user ID has write access) as virtual devices 2100 and 2101 using the CP **LINK** command:

```
# vmcp link linux01 100 2100 mr
# vmcp link linux01 101 2101 mr
```

### Copy the source to the target disk using FLASHCOPY

The two disks are copied using the CP **FLASHCOPY** command:

```
# vmcp flashcopy 1100 0 end to 2100 0 end
Command complete: FLASHCOPY 1100 0 END TO 2100 0 END
# vmcp flashcopy 1101 0 end to 2101 0 end
Command complete: FLASHCOPY 1101 0 END TO 2101 0 END
```

### Detach the source disks

Now that you no longer need the source disks linked, detach them:

```
# vmcp det 1100-1101
1100-1101 DETACHED
```

**Attention:** If you do not have **FLASHCOPY** support, you can use the Linux **dasdfmt** and **dd** commands. You must first enable the 1100-1101 and 2100-2101 disks using the **chccwdev -e** command, then determine the newly created device nodes using the **lsdasd** command:

```
# chccwdev -e 1100-1101,2100-2101
Setting device 0.0.1100 online
Done
Setting device 0.0.1101 online
Done
Setting device 0.0.2100 online
Done
Setting device 0.0.2101 online
Done
# lsdasd
...
0.0.1100(ECKD) at ( 94:128) is dasdag     : active at blocksize 4096, 600840 blocks,
2347 MB
0.0.1101(ECKD) at ( 94:132) is dasdah     : active at blocksize 4096, 600840 blocks,
2347 MB
0.0.2100(ECKD) at ( 94:136) is dasdai     : active at blocksize 4096, 600840 blocks,
2347 MB
0.0.2101(ECKD) at ( 94:140) is dasdaj     : active at blocksize 4096, 600840 blocks,
2347 MB
```

In this example the source minidisks (1100-1101) are named /dev/dasdag and /dev/dasdah, while the target minidisks (2100-2101) are named /dev/dasdai and /dev/dasdaj. Format the target devices using the dasdfmt command using a 4 KB (4096 byte) block size:

```
# dasdfmt -b 4096 -y -f /dev/dasdai
Finished formatting the device.
Rereading the partition table... ok
# dasdfmt -b 4096 -y -f /dev/dasdaj
Finished formatting the device.
Rereading the partition table... ok
```

Now that the devices have been formatted, you can copy the volumes of the golden image using the **dd** command, again using a block size of 4 KB (4096) bytes:

```
# dd if=/dev/dasdag of=/dev/dasdai bs=4096
600840+0 records in
600840+0 records out
2461040640 bytes (2.5 GB) copied, 81.2098 seconds, 30.3 MB/s
# dd if=/dev/dasdah of=/dev/dasdaj bs=4096
600840+0 records in
600840+0 records out
2461040640 bytes (2.5 GB) copied, 81.2098 seconds, 30.3 MB/s
```

Then bring the devices offline so the new file systems will be recognized when brought back online:

```
# chccwdev -d 1100-1101,2100-2101
```

### Bring the newly copied LVM online

To access the root file system on the newly cloned golden image, you must activate the logical volume on the newly copied disks. Enable the new minidisks at virtual device addresses 2100 and 2101:

```
# chccwdev -e 2100-2101
Setting device 0.0.2100 online
```

```
Done
Setting device 0.0.2101 online
Done
```

Now run **vgscan** and **vgchange** to activate the newly cloned LVM:

```
# vgscan
  Reading all physical volumes.  This may take a while...
  Found volume group "system_vg" using metadata type lvm2
  Found volume group "install_vg" using metadata type lvm2
# vgchange -a y system_vg
  5 logical volume(s) in volume group "system_vg" now active
```

## Mount the newly copied root file system

Run **lvdisplay** to show the new root logical volume:

```
# lvdisplay
--- Logical volume ---
  LV Name                /dev/system_vg/root_lv
  VG Name                system_vg
...
```

In this case, the root file system is located at /dev/system_vg/root_lv. Because you only need to modify files in /etc, there is no need to mount the /opt, /var, or /usr logical volumes in system_vg. Mount root_lv over a newly created mount point /mnt/linux01:

```
# mkdir /mnt/linux01
# mount /dev/system_vg/root_lv /mnt/linux01
```

Observe that this appears to be a root file system:

```
# ls /mnt/linux01/
bin   dev  home  lib64       media  opt   root  selinux  sys  usr
boot  etc  lib   lost+found  mnt    proc  sbin  srv      tmp  var
```

## Modify networking information about the target system

In this example, the only two pieces of networking information that are modified are the IP address (from 9.12.5.29 to 9.12.5.31) and the host name (from virtc529 to virtc531).

The host name and IP address are changed in the file /etc/hosts:

```
# cd /mnt/linux01/etc
# vi hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1               localhost.localdomain localhost
::1             localhost6.localdomain6 localhost6
9.12.5.31          virtc531.itso.ibm.com virtc531
```

The host name is changed in the file /etc/sysconfig/network:

```
# cd sysconfig
# vi network
NETWORKING=yes
NETWORKING_IPV6=yes
HOSTNAME=virtc531.itso.ibm.com
GATEWAY=9.12.4.1
```

The IP address is changed in the file /etc/sysconfig/network-scripts/ifcfg-eth0:

```
# cd network-scripts
# vi ifcfg-eth0
# IBM QETH
DEVICE=eth0
BOOTPROTO=static
IPADDR=9.12.5.31
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
MTU=1500
NETMASK=255.255.252.0
NETTYPE=qeth
ONBOOT=yes
PORTNAME=DONTCARE
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
MTU=1500
ARP=no
```

## Detach the target disks

Change to the default directory. Use the **sync** command to flush the disks, and the **umount** command to unmount the modified root file system:

```
# cd
# sync
# umount /mnt/linux01
```

Deactivate the LVM with **vgchange**:

```
# vgchange -a n system_vg
  0 logical volume(s) in volume group "system_vg" now active
```

Lastly, set the LINUX01 1100-1101 disks offline with the **chccwdev** command and detach them using the CP **DETACH** command:

```
# chccwdev -d 2100-2101
Setting device 0.0.2100 offline
Done
Setting device 0.0.2101 offline
Done
# vmcp det 2100-2101
1100-1101 DETACHED
```

You should now be ready to IPL the manually cloned system.

## IPL the target system

Logon to a 3270 session as LINUX01. CMS will IPL and the **PROFILE EXEC** will ask you if you want to IPL from minidisk 100. Type y for yes, and then you can enter the command **#cp vi vmsg 0** to select the default Linux image from the boot menu:

```
LOGON LINUX01
...
Do you want to IPL Linux from minidisk 100? y/n
y
00: zIPL v1.5.3 interactive boot menu
00:
00:  0. default (linux)
00:
00:  1. linux
00:
00: Note: VM users please use '#cp vi vmsg <input>'
00:
00: Please choose (default will boot in 15 seconds):
00:
00: CP VI VMSG 0
00: Booting default (linux)...
```

```
Linux version 2.6.18-92.el5 (brewbuilder@btest1.z900.redhat.com) (gcc version 4.1.2
20071124 (Red Hat 4.1.2-41)) #1 SMP Tue Apr 1 19:03:24 EDT 2008
We are running under VM (64 bit mode)
...
Red Hat Enterprise Linux Server release 5.2 (Tikanga)
Kernel 2.6.18-92.el5 on an s390x

virtc531 login:
```

Your new system should come up cleanly using the modified IP address and host name.

### Modify the SSH keys on the target system

Start an SSH session to the new clone as root.

The SSH keys that were copied are identical to those of the golden image. Remove them and restart the **sshd** service. This will trigger the **ssh-keygen** command to automatically create three new keys that are unique to this new guest. Note that **sshd** is designed so that it can be restarted remotely, so you will not lose your SSH client connection:

```
# cd /etc/ssh
# rm -f /etc/ssh/ssh_host_*
# service sshd restart
Stopping sshd: [  OK  ]
Generating SSH1 RSA host key: [  OK  ]
Generating SSH2 RSA host key: [  OK  ]
Generating SSH2 DSA host key: [  OK  ]
Starting sshd: [  OK  ]
```

At this point you have now cloned a Linux system manually. You can look around the new system. It should be identical to the golden image except for the IP address and host name.

Next you will learn how to clone a Linux system automatically. You will use the LINUX01 user ID again. To clone, the target user ID must be logged off. You could shut the new system down cleanly, but because you will be cloning again, it does not matter. Go to the 3270 session and log off the LINUX01 user ID:

```
==> #cp log
```

# 9.3 Cloning one new virtual server

Now that you have cloned a server manually and better understand the steps, you can use the **clone** script to clone automatically.

## 9.3.1 Creating a configuration file for LINUX01

For each Linux guest you want to clone, you must create a configuration file that you can use to customize the image after cloning. Perform the following steps on the LNXINST installation server:

1. Open an SSH session to LNXINST as root.
2. Install the clone script RPM:

```
# rpm -ivh /nfs/virt-cookbook-RH5.2/clone-1.0-5.s390x.rpm
Preparing...                ########################################### [100%]
   1:clone                  ########################################### [100%]
```

3. Copy and then edit the supplied sample configuration file to reflect the values of the new Linux system:

```
# cp /etc/clone/rhel.conf.sample /etc/clone/linux01.conf
# vi /etc/clone/linux01.conf
```

4. Edit the new configuration file with the appropriate values for your system. This file is based on the RHEL52 CONF file that you used earlier when building the RHEL 5.2 golden image; see 7.1.3, “Preparing LNXINST bootstrap files” on page 94.

   If the new Linux image is going to be on the same network as the golden image, you are likely to only have to change two variables: the Internet Protocol (IP) address (IPADDR) and the Domain Name System (DNS) name (HOSTNAME). In the following example, the IP address is set to 10.1.40.91 and the DNS name to linux01.example.com.

```
# vi /etc/clone/linux01.conf
# Define the DASD that should be included as a part
# of the clone.
DASD=100,101                      1
DASD_ROOT=100                     2
VG_NAME=system_vg                 3
LV_ROOT=root_lv                   4

# Define networking information that will be used for the host.
IPADDR=9.12.5.31
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
HOSTNAME=virtc531.itso.ibm.com
NETTYPE=qeth
NETMASK=255.255.252.0
NETWORK=9.12.4.0
SEARCHDNS=itso.ibm.com
BROADCAST=9.12.4.255
GATEWAY=9.12.4.1
DNS=9.12.6.7
MTU=1500
```

Note the following points for the numbers in black above:

1 This is the DASD range. You may enter dashes (-) or commas (,) to specify address ranges or specific disks, respectively. Make sure that the range following DASD= is one continuous block of text with no spaces added.

2 This is the DASD that contains the root file system. If the root file system is on LVM, this value is ignored.

3 If the root file system of the golden image is on LVM, specify the volume group name here. If you followed the installation procedure in Chapter 8, “Installing and configuring RHEL 5.2” on page 119, the volume group name is **system_vg**.

4 If you specified a value for 3 (VG_NAME), specify the logical volume name of the root file system.

5. Save the file and log off root.

   This is your first time cloning, therefore it is important to verify some initial settings of the new user ID. The next steps confirm that the LINUX01 user ID has the disk configuration required for a successful clone and IPL.

6. Log in to LINUX01.

7. Answer n to the question Do you want to IPL Linux from minidisk 100? y/n. Verify that the minidisks at addresses 100 and 101 and the virtual disk (VDISK) at addresses 300 and 301 are read/write:

```
==> q da
```

```
00: DASD 0100 3390 LXAE23 R/W       3338 CYL ON DASD  AE23 SUBCHANNEL = 0000
00: DASD 0101 3390 LXAE24 R/W       3338 CYL ON DASD  AE24 SUBCHANNEL = 0001
...
00: DASD 0300 9336 (VDSK) R/W     524288 BLK ON DASD  VDSK SUBCHANNEL = 000E
00: DASD 0301 9336 (VDSK) R/W    1048576 BLK ON DASD  VDSK SUBCHANNEL = 000F...
```

8. Log off `LINUX01`.

You are now ready to clone to this new user ID.

## 9.3.2 Using the clone script

Go back to your an SSH session to the controller. The **clone** script should be in your `PATH` in the directory `/usr/sbin/`. You can verify this with the **which** command:

```
# which clone
/usr/sbin/clone
```

The **clone** script can operate in two modes:

- Where the DASD information is provided on the command line
- Where the DASD information is included in the new user ID's configuration file

Running **clone** with no arguments prints a usage message as follows:

```
# clone
Usage: clone [-v] masterGuestID cloneGuestID [rootMinidisk [minidisk1 minidisk2..]]
    Switches
       -v Verbose output
    Required
       masterGuestID the z/VM user id you want to clone from
       cloneGuestID  the z/VM user id you want to clone to
    Optional
       rootMinidisk the minidisk address that contains the root filesystem
       minidisk1..n additional minidisks that should be copied
```

The masterGuestID is the z/VM ID of the master Linux image (`RHEL52`), and the cloneGuestID is the z/VM ID of the target (`LINUX01`). These values are *always* required.

If no DASD arguments are specified on the command line, then you can collect the DASD information from the user's configuration file stored in `/etc/clone/`. The script uses the DASD value to determine which minidisks to copy as a part of the clone process. In the following examples, DASD is set to 100-101, which implies that minidisks located at virtual addresses 100 and 101 are copied. The 300 and 301 VDISKs are omitted because SWAPGEN automatically creates them each time the user logs on. The `DASD_ROOT` value specifies which one of these minidisks contains the Linux root filesystem (/).

The script exits if either the golden image or the clone image is logged in. The script first attempts to copy the disks with FLASHCOPY using the **vmcp** module or command. If an error is returned, the script falls back to using Linux **dasdfmt** and **dd** commands. Finally, the script boots the new Linux image using the **xautolog** command. It takes less than a minute to clone with **FLASHCOPY** support, and three to 20 minutes with **dd**. The following is an example of cloning from `RHEL52` to `LINUX01` with **FLASHCOPY** support. The example uses the verbose switch (**-v**) to clarify its actions.

```
# clone -v rhel52 linux01
Invoking CP command: QUERY rhel52
Invoking CP command: QUERY linux01

This will copy disks from rhel52 to linux01
```

```
Host name will be:  linux01.example.com
IP address will be: 10.1.40.91
Do you want to continue? (y/n): y
```

The script makes sure the golden image user ID and the target user ID exist and are logged off. Then, it confirms the order of the cloning and displays information collected from the /etc/clone/linux01.conf file. Following this, it asks if you are sure you want to overwrite the disks on the target user ID.

Next, the script links to the master clone minidisk and the target minidisk. The master minidisks are linked to LNXINST at virtual address FFFE, and the target minidisks are linked as FFFF. The FFFE links are read-only and the FFFF links are read-write. With the links in place, the script issues a **FLASHCOPY** command to copy the master's 100 and 101 minidisks to the clone's 100 and 101 minidisks, respectively. The script then detaches the links. If **FLASHCOPY** fails, the script attempts to use **dasdfmt** and **dd**.

```
Cloning rhel52 to linux01 ...
Copying minidisks...
Invoking CP command: QUERY VIRTUAL FFFE
Invoking CP command: LINK RHEL52 0100 FFFE RR
Invoking CP command: QUERY VIRTUAL FFFF
Invoking CP command: LINK LINUX01 0100 FFFF W
Invoking CP command: FLASHCOPY FFFE 0 END FFFF 0 END
0100 disk copied ...
Invoking CP command: DETACH FFFE
Invoking CP command: DETACH FFFF
Invoking CP command: QUERY VIRTUAL FFFE
Invoking CP command: LINK RHEL52 0101 FFFE RR
Invoking CP command: QUERY VIRTUAL FFFF
Invoking CP command: LINK LINUX01 0101 FFFF W
Invoking CP command: FLASHCOPY FFFE 0 END FFFF 0 END
0101 disk copied ...
Invoking CP command: DETACH FFFE
Invoking CP command: DETACH FFFF
```

Next, the LVM is brought online if necessary to access the root file system. This only happens if the variables VG_NAME and LV_ROOT were set in the linux01.conf configuration file. If so, you will see output similar to the following:

```
  Reading all physical volumes.  This may take a while...
  Found volume group "system_vg" using metadata type lvm2
  Found volume group "install_vg" using metadata type lvm2
  5 logical volume(s) in volume group "system_vg" now active
```

Then, the root file system is mounted to /mnt/clone, and the networking information is modified in /mnt/clone/etc/sysconfig/network/ifcfg-eth0, /mnt/clone/etc/sysconfig/network, and /mnt/clone/etc/hosts:

```
Updating cloned image ...
Invoking CP command: QUERY VIRTUAL FFFF
Invoking CP command: LINK LINUX01 0100 FFFF W
Modifying networking info under /mnt/clone...
Regenerating SSH keys in /mnt/clone/etc/ssh/ ...
Invoking CP command: DETACH FFFF
Clone complete
```

Then the SSH keys are regenerated in such a way that they are unique for the new virtual server. The new root file system is then unmounted, set offline, and detached:

```
Invoking CP command: XAUTOLOG LINUX01
Booting linux01
```

```
Successfully cloned RHEL52 to LINUX01
```

In the final section, the `LINUX01` user ID is logged on using **`XAUTOLOG`**. Because the shared `PROFILE EXEC` detects that the user ID is in a disconnected mode, it carries out an IPL of Linux from minidisk 100.

**Note:** If the clone script fails, you can check that:

- The configuration contains all of the correct information in `/etc/clone/`.
- No other users have links to the clone's read-write disks.

A block diagram of this process is displayed in Figure 9-1.

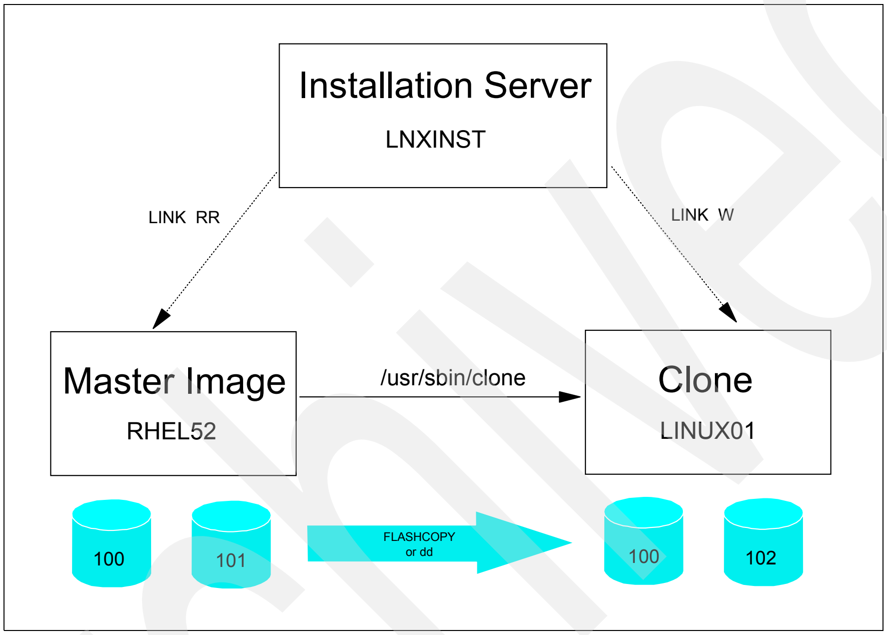


*Figure 9-1 Cloning block diagram*

The top of the figure shows the Linux installation server that is running from the `LNXINST` user ID. In order to **`FLASHCOPY`** or **`dd`**, the `LNXINST` user ID requires a **`LINK`** to the source minidisks that `RHEL52` owns and the destination minidisks that `LINUX01` owns. The figure shows that the **`LINK`** statement is issued as read-only (RR) for the golden image, and as read/write (W) for the clone image. Note that the VDISK-based swap space `RHEL52` 300-301 is created in memory; therefore, it does not need to be copied.

## 9.4 Cloning three more virtual servers

So far you have installed Linux manually twice onto `LINUX00` and `RHEL52`. You have created a new user ID `LINUX01` and cloned to it. Now it is time to clone three more times to have one system for each of the virtual servers described in the remaining chapters.

The following steps are involved:

- Defining three more user IDs
- Creating three new configuration files
- Granting user IDs access to VSWITCH
- Testing logging on to a new user ID

## 9.4.1 Defining three more user IDs

Define three more user IDs for Linux virtual servers in the USER DIRECT file named LINUX02 - LINUX04. You will need to use the DASD volumes you just formatted: two for each virtual server. You can repeat the definition of LINUX01 three times with the block copy **""3** prefix command. For example:

```
==> x user direct
====> /user linux01
...
""3   *
02142 USER LINUX01 LNX4VM 256M 1G G
02143  INCLUDE LNXDFLT
02144  OPTION APPLMON
02145  MDISK 100 3390 0001 3338 <MMD957> MR LNX4VM LNX4VM LNX4VM
""     MDISK 101 3390 0001 3338 <MMDA51> MR LNX4VM LNX4VM LNX4VM
```

This will create three more copies of the LINUX01 user definition. Modify them to have a user ID of LINUX02 - LINUX04, and give each correct DASD labels:

```
*
 USER LINUX02 LNX4VM 256M 1G G
  INCLUDE LNXDFLT
  OPTION APPLMON
  MDISK 100 3390 0001 3338 <MMDA52> MR LNX4VM LNX4VM LNX4VM
  MDISK 101 3390 0001 3338 <MMDA53> MR LNX4VM LNX4VM LNX4VM
 *
 USER LINUX03 LNX4VM 256M 1G G
  INCLUDE LNXDFLT
  OPTION APPLMON
  MDISK 100 3390 0001 3338 <MMDA54> MR LNX4VM LNX4VM LNX4VM
  MDISK 101 3390 0001 3338 <MMDA55> MR LNX4VM LNX4VM LNX4VM
 *
 USER LINUX04 LNX4VM 256M 1G G
  INCLUDE LNXDFLT
  OPTION APPLMON
  MDISK 100 3390 0001 3338 <MMDA56> MR LNX4VM LNX4VM LNX4VM
  MDISK 101 3390 0001 3338 <MMDA57> MR LNX4VM LNX4VM LNX4VM
```

Go to the top of the file and find the definition for the user $ALLOC$. Add dummy definitions for cylinder 0 of each of the new volumes and save the changes:

```
====> top
====> /alloc
USER $ALLOC$  NOLOG
 MDISK A01 3390 000 001 520RES R
...
 MDISK A0B 3390 000 001 <MMD957> R
 MDISK A0C 3390 000 001 <MMDA51> R
 MDISK A0D 3390 000 001 <MMDA52> R
 MDISK A0E 3390 000 001 <MMDA53> R
 MDISK A0F 3390 000 001 <MMDA54> R
 MDISK A10 3390 000 001 <MMDA55> R
 MDISK A11 3390 000 001 <MMDA56> R
 MDISK A12 3390 000 001 <MMDA57> R
...
====> file
```

Check for overlaps and the single gap. **Quit** out of the USER DISKMAP file:

```
==> diskmap user
==> x user diskmap
```

```
====> pre off
====> all /gap/|/overlap/
--------------------  4  line(s) not displayed --------------------
                                      0         500         501    GAP
--------------------  368  line(s) not displayed --------------------
====> quit
```

Bring the changes online with the **DIRECTXA USER** command:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 5 RELEASE 1.0
EOJ DIRECTORY UPDATED AND ON LINE
```

You have now created three new user IDs that can be cloned to.

### 9.4.2 Creating three new configuration files

You must create a new configuration file for each of the user IDs. Because most of the information generally remains the same across clones, **clone** has a feature that allows the definition of a shared.conf file to be included into all other configuration files. Create this file by first copying linux01.conf:

```
# cp /etc/clone/linux01.conf /etc/clone/shared.conf
# vi /etc/clone/shared.conf
```

Now edit the new file to remove any values specific to each user ID's individual configuration file. Remove HOSTNAME= and IPADDR=, as shown in the strikethrough text, because they always change:

```
~~HOSTNAME=linux01.example.com~~
~~IPADDR=10.1.40.91~~
DASD=100,101
DASD_ROOT=100
NETTYPE=qeth
...
```

All of the Linux clones reside on the same network, and come from the same golden image. Therefore, it is possible to place the majority of the configuration settings into the shared.conf file. Now create a new linux02.conf file using the new format:

```
# vi /etc/clone/linux02.conf
HOSTNAME=virtc532.itso.ibm.com
IPADDR=9.12.5.32
```

Copy this file and modify it for LINUX03 and LINUX04.

### 9.4.3 Granting user IDs access to VSWITCH

Modify the PROFILE EXEC on AUTOLOG1 191 to grant access to the VSWITCH for the six new user IDs and add **XAUTOLOG** commands so they will booted when the z/VM system IPLs.

Link and access the AUTOLOG1 191 disk so the file can be modified from MAINT:

```
==> link autolog1 191 1191 mr
==> acc 1191 f
```

Edit the **PROFILE EXEC**:

```
==> x profile exec f
...
/* Grant access to VSWITCH for each Linux user */
```

```
'CP SET VSWITCH VSW1 GRANT LNXINST'
'CP SET VSWITCH VSW1 GRANT RHEL52'
'CP SET VSWITCH VSW1 GRANT LINUX01'
'CP SET VSWITCH VSW1 GRANT LINUX02'
'CP SET VSWITCH VSW1 GRANT LINUX03'
'CP SET VSWITCH VSW1 GRANT LINUX04'

/* XAUTOLOG each Linux user that should be started */
'CP XAUTOLOG LNXINST'
'CP XAUTOLOG LINUX01'
'CP XAUTOLOG LINUX02'
'CP XAUTOLOG LINUX03'
'CP XAUTOLOG LINUX04'
====> file
```

It is easiest to grant access to the new user IDs for the current z/VM session with the **SET VSWITCH** command:

```
==> set vswitch vsw1 grant linux02
Command complete
==> set vswitch vsw1 grant linux03
Command complete
==> set vswitch vsw1 grant linux04
Command complete
```

Verify that the user IDs have access with the **QUERY VSWITCH ACCESSLIST** command:

```
==> query vswitch vsw1 acc
VSWITCH SYSTEM VSW1     Type: VSWITCH Connected: 4 Maxconn: INFINITE
  PERSISTENT  RESTRICTED    NONROUTER                 Accounting: OFF
  VLAN Unaware
  State: Ready
  IPTimeout: 5         QueueStorage: 8
Portname: UNASSIGNED RDEV: 3004 Controller: DTCVSW1 VDEV:  3004
  Portname: UNASSIGNED RDEV: 3008 Controller: DTCVSW2 VDEV:  3008 BACKUP
    Authorized userids:
       RHEL52    LINUX01   LINUX02   LINUX03   LINUX04    LNXINST
       SYSTEM
```

## 9.4.4 Testing logging on to a new user ID

You should now be able to logon to a new user ID and verify the integrity of the definitions. Logon to LINUX02 and you should first notice that a NIC is created, as well as two VDISKs:

```
LOGON  LINUX02
00: NIC 0600 is created; devices 0600-0602 defined
00: z/VM Version 5 Release 3.0, Service Level 0702 (64-bit),
00: built on IBM Virtualization Technology
00: There is no logmsg data
00: FILES:   NO RDR,   NO PRT,   NO PUN
00: LOGON AT 10:17:55 EST WEDNESDAY 12/13/06
z/VM V5.3.0    2007-11-18 09:54


DMSACP723I A (191) R/O
DMSACP723I C (592) R/O
DIAG swap disk defined at virtual address 300(64989 4K pages of swap space)
DIAG swap disk defined at virtual address 301(129981 4K pages of swap space)
Do you want to IPL Linux from minidisk 100? y/n
n
```

If you forgot to grant access to the VSWITCH you will see an error message. Verify that you have OSA devices at addresses 600-602:

```
==> q osa
00: OSA  0600 ON NIC  0600  UNIT 000 SUBCHANNEL = 0002
00:      0600 QDIO-ELIGIBLE       QIOASSIST-ELIGIBLE
...
```

Verify that you have two read/write devices at addresses 100-101 using the **QUERY DASD** command:

```
==> q da
00: DASD 0100 3390 LXAE23 R/W       3338 CYL ON DASD  AE23 SUBCHANNEL = 0000
00: DASD 0101 3390 LXAE24 R/W       3338 CYL ON DASD  AE24 SUBCHANNEL = 0001
00: DASD 0190 3390 LX8RES R/O        107 CYL ON DASD  CF31 SUBCHANNEL = 0009
00: DASD 0191 3390 LX8027 R/O        300 CYL ON DASD  8027 SUBCHANNEL = 000C
...
```

Logoff from LINUX02.

At this point you have cloned one Linux virtual server and defined three more user IDs that should now be ready for cloning or kickstarting to. You will clone to these user IDs in the chapters that follow.

## 9.5 Reviewing system status

Now you can view your system from a DASD point of view, as shown in Figure 9-2 on page 146. If you have implemented the instructions in all sections in this book, you should have used 24 3390-3 volumes: 10 for your z/VM system; 7 for the Linux controller and golden image; and one for each of the seven virtual servers.

You can also view the system from an administrator's perspective and from a user's perspective, as shown by the horizontal lines and the italicized text on the right side of Figure 9-2 on page 146. The z/VM and Linux system administration roles may be performed by the same person, but these roles can also be done by different administrators. The Linux users may not care or be aware that their servers are virtual machines which might have been cloned in a matter of minutes.

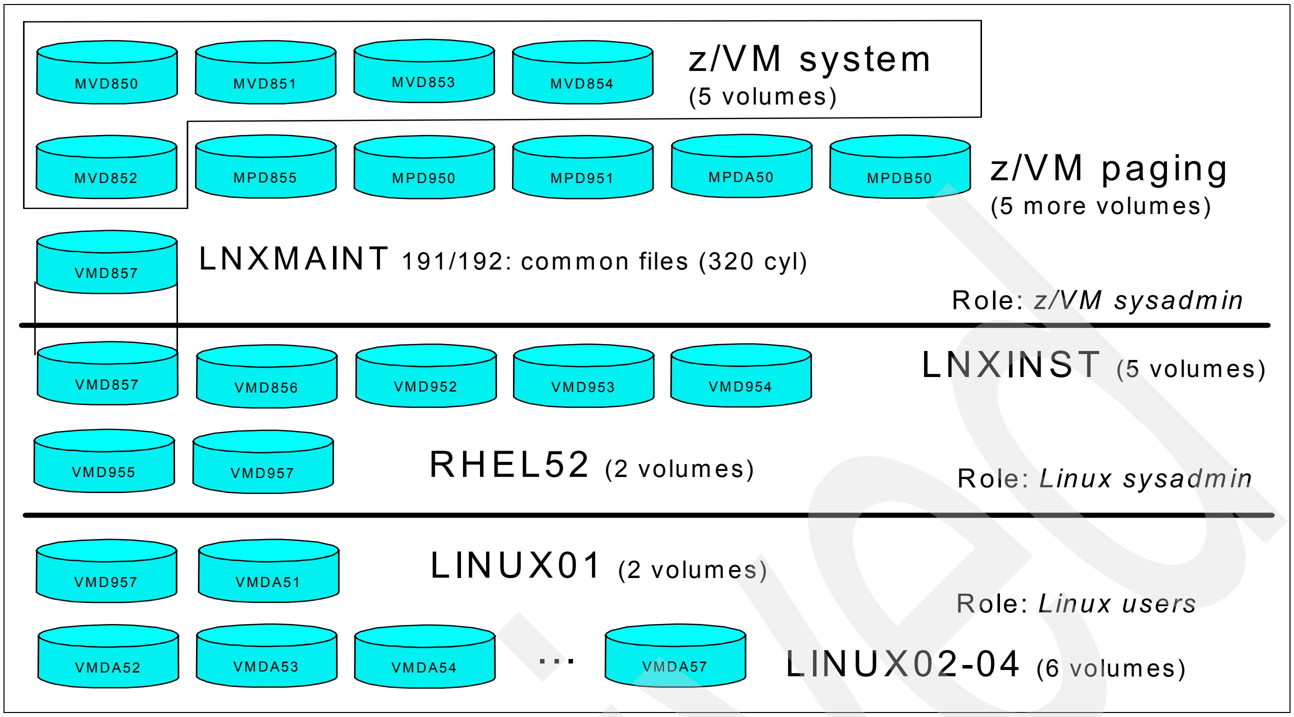


*Figure 9-2 Linux virtual server system - DASD view and role view*

10

# Installing Linux with kickstart

Kickstart is an automated way of installing RHEL 5.2. Using kickstart, you can create a single file that answers all of the questions usually asked during an interactive installation.

In Chapter 9, "Configuring RHEL 5.2 for cloning" on page 129, you cloned to LINUX01 and created three new user IDs for virtual servers. In this chapter, you will kickstart a RHEL 5.2 system to LINUX02. In comparison, cloning a server is faster, assuming the FLASHCOPY command is available. However, kickstarting a server is more flexible, because it allows for different package configurations as well as pre-installation and post-installation scripting.

The controller is now configured as an installation server, with the installation tree shared using NFS. You will now configure it as a kickstart server to perform automated installations over the network. Then you will install the final Linux server with kickstart.

The following steps are involved in installing Linux with kickstart:

- Configure the controller for kickstart
- Configure the LINUX02 user for kickstart
- Kickstart the LINUX02 user

## 10.1 Configure the controller for kickstart

In this section, you configure the controller to host the kickstart file, which you use to perform the automated installation of the Linux virtual server called `LINUX02`. This section assumes you have already set up the RHEL 5.2 install tree as described in 7.2, “Configuring the controller” on page 109.

The installer generates a kickstart file at the end of every installation. It is based on the answers provided during the interactive install. This kickstart file is named `anaconda-ks.cfg` and is located in the `/root/` directory. You use this kickstart file from `RHEL52` as a template for `LINUX02`. Perform the following steps:

1. From an Secure Shell (SSH) session to the controller, create a directory to hold the kickstart file. Then, copy the default kickstart file to this directory, rename it, and modify the permissions so others may read it:

```
# mkdir /nfs/ks
# cp /root/anaconda-ks.cfg /nfs/ks/linux02-ks.cfg
# chmod +r /nfs/ks/linux02-ks.cfg
```

2. Next, use **vi** to edit the kickstart configuration file. To customize this kickstart for `LINUX02`, remove the lines with the strikethrough, and edit the lines shown in bold font:

```
# vi /nfs/ks/linux02-ks.cfg
install
nfs --server=9.12.5.30 --dir=/nfs/rhel5.2                        1
lang en_US.UTF-8
network --device eth0 --mtu=1500 --bootproto static --ip 9.12.5.32   2
--netmask 255.255.255.0 --gateway 9.12.4.1 --nameserver 9.12.6.7
--hostname virtc532.itso.ibm.com                                     3
rootpw --iscrypted $1$d6Mi5lj9$jzwhlKOUK37qXQzTLVRSm.
firewall --disabled ~~--port=22:tcp~~                                4
authconfig --enableshadow --enablemd5
selinux --enforcing
timezone America/New_York
bootloader --location=mbr ~~--driveorder=dasda,dasdd,dasdg,dasdh,dasdi,dasdj,dasdk~~
                  5
reboot                                                                6
key --skip                                                            7
# The following is the partition information you requested
# Note that any partitions you deleted are not expressed
# here so unless you clear all partitions first, this is
# not guaranteed to work
zerombr yes                                                  8
clearpart --all --initlabel                                  9
part / --fstype ext3 --size=1 --grow --ondisk=dasda         10
~~#clearpart --linux~~                                          11
~~#part /boot --fstype ext3 --size=100 --ondisk=dasda~~
~~#part pv.11 --size=100 --grow --ondisk=dasde~~
~~#part pv.10 --size=100 --grow --ondisk=dasdd~~
~~#part pv.9 --size=100 --grow --ondisk=dasdc~~
~~#part pv.8 --size=100 --grow --ondisk=dasdb~~
~~#part / --fstype ext3 --size=100 --grow --ondisk=dasda~~
~~#volgroup install_vg --pesize=32768 pv.8 pv.9 pv.10 pv.11~~
~~#logvol /nfs --fstype ext3 --name=install_lv --vgname=install_vg --size=9120~~

%packages                                                   12
@core
@base
@games
```

```
@base-x
device-mapper-multipath
-sysreport

%post                                                  13
echo /dev/dasdq1 swap swap defaults 0 0 >> /etc/fstab  14
echo /dev/dasdr1 swap swap defaults 0 0 >> /etc/fstab
```

The numbers in bold font on the command lines refer to the descriptions that follow.

**1** The IP address of the installation server and the path to install the tree.

**2** The IP address of the new Linux server.

**3** The host name of the new Linux server.

**4** Disable the firewall. This is only advisable if the server is *not* on an external network.

**5** Remove references to additional drives only available to the controller.

**6** Add this line so that the server automatically shuts down after kickstart.

**7** Skip entering an installation number. If you have one, enter it here instead of using **`--skip`**.

**8** Add this line to perform a non-interactive direct access storage device (DASD) format.

**9** Removes all existing partitions. Uncomment this line and add **`--initlabel`**.

**10** Defines the root partition. The **`--size 1`** and **`--grow`** options specify all of the 100 disk.

**11** Previously defined Logical Volume Manager (LVM) used for install tree. *Delete* these lines.

**12** Specifies the packages to install. The `@core` entry is *required* for a minimal install.

**13** Add a section for the post-installation script.

**14** Add entries for virtual disk (VDISK) swap spaces in `/etc/fstab`.

> **Note:** The **`zerombr`** and **`clearpart`** options are chosen because the DASD is not yet formatted for Linux use. For future kickstarts to the same DASD volumes, they will already be formatted. Thus, you can save a substantial amount of time by only recreating the file systems and not reformatting the DASD. In this case you would use the following kickstart options:
>
> ```
> zerombr no
> clearpart --all
> ```

3. Finally, add the path to the kickstart folder to `/etc/exports`

```
# vi /etc/exports
/nfs/rhel5.2/            *(ro,sync)
/nfs/virt-cookbook-RH5.2 *(ro,sync)
/nfs/ks                  *(ro,sync)
```

4. Restart the NFS service on the controller. The **`showmount -e`** command should show the exported file systems:

```
# service nfs restart
Shutting down NFS mountd: [  OK  ]
Shutting down NFS daemon: [  OK  ]
Shutting down NFS quotas: [  OK  ]
Shutting down NFS services:  [  OK  ]
Starting NFS services:  [  OK  ]
Starting NFS quotas: [  OK  ]
```

```
Starting NFS daemon: [  OK  ]
Starting NFS mountd: [  OK  ]
# showmount -e
Export list for virtc530.itso.ibm.com:
/nfs/ks                 *
/nfs/rhel5.2            *
/nfs/virt-cookbook-RH5.2 *
```

## 10.2 Configure the LINUX02 user for kickstart

Earlier you should have created the user ID LINUX02, and at this point you configure it for kickstart. `LINUX02` must have its own parameter and configuration files, which are based on the `RHEL52` user ID.

**`LOGOFF`** from `MAINT` and logon to `LNXMAINT`. Copy the parameter and configuration files from `RHEL52` to `LINUX02` as follows:

```
==> copy rhel52 parm-rh5 d linux02 = =
==> copy rhel52 conf-rh5 d linux02 = =
```

Edit the `LINUX02 PARM-RH5` file. Because this is a non-interactive installation, the vnc options are no longer required. The **`ks=`** line directs the installer to obtain the kickstart file from the installation server. **`RUNKS=1`** is required for kickstarts, and the **`cmdline`** option prevents the installer's text-based user interface from opening on the 3270 console:

```
==> x linux02 parm-rh5 d
ramdisk_size=40000 root=/dev/ram0 ro ip=off
CMSDASD=191 CMSCONFFILE=linux02.conf-rh5
ks=nfs:9.12.5.30:/nfs/ks/linux02-ks.cfg
RUNKS=1 cmdline
====> file
```

Next, edit the `LINUX02 CONF` file, and change the DASD range and networking information:

```
==> x linux02 conf-rh5 d
DASD=100-10f,300-30f
HOSTNAME=<virtc532.itso.ibm.com>
NETTYPE=qeth
IPADDR=<9.12.5.32>
...
====> file
```

## 10.3 Kickstart the LINUX02 user

Perform the following steps to kickstart the `LINUX02` user:

1. **`Logoff`** from `LNXMAINT` and logon to `LINUX02`. When asked to IPL from disk 100, reply `n`:

   ```
   LOGON LINUX02
   ...
   Do you want to IPL Linux from minidisk 100? y/n
   n
   ```

2. Add more memory for the install process. Temporarily modify the storage up to 512 MB with the **`DEFINE STORAGE`** command. Then **`IPL CMS`** and again answer `n` to the question of IPLing Linux:

```
==> def stor 512m
```

```
00: STORAGE = 512M
00: Storage cleared - system reset.
==> ipl cms
...
Do you want to IPL Linux from minidisk 100? y/n
n
```

Verify that you have a 512MB virtual machine:

```
==> q v stor
00: STORAGE = 512M
```

This change is for the duration of the user ID session. When you logoff and log back on this user ID, the storage will revert to 256 MB.

3. Run **rhel52 exec** to initiate the kickstart. You see some initial kernel messages, followed by the file system format and Red Hat Package Manager (RPM) package installation.

   **Note:** Towards the end of the kickstart, it is normal to see some unrecognized characters on the window. This is because the 3270 console cannot display the progress meter during the post-nstallation phase. To automatically clear the 3270 console and avoid multiple windows of unreadable messages, issue the **#cp term more 0 0** command before running **RHEL52 EXEC**.

   ```
   ==> rhel52
   ...
   Kernel command line: ramdisk_size=40000 root=/dev/ram0 ro ip=off
                        CMSDASD=191 CMSCONFFILE=LINUX02.CONF
                        ks=nfs:9.12.5.30:/nfs/ks/linux02-ks.cfg
                        RUNKS=1 cmdline
   ...
   ```

4. The first time that kickstart is run, the installer must format the DASD for Linux use. It is normal to see error messages of the following format if the DASD you are using has never been formatted. In subsequent kickstart installs, you should not see these errors:

   ```
   end_request: I/O error, dev dasda, sector 0
   Buffer I/O error on device dasda, logical block 0
   Please wait while formatting drive dasda...
   ```

   **Note:** The VDISK swap device is configured in the %post section of the kickstart file.

5. At the end of the kickstart, IPL the 100 disk to make any changes to your RHEL 5.2 golden image:

   ```
    /mnt/sysimage/dev done
    /mnt/sysimage done
   you may safely reboot your system
   ==> #cp ipl 100
   00: zIPL v1.5.3 interactive boot menu
   00:  0. default (linux)
   00:  1. linux
   ...
   ```

Now you have installed Linux onto the virtual server using kickstart. This process can be repeated for other Linux guests. Here we showed a minimal installation with kickstart, but you can completely customize the kickstart file to install different packages based on your requirements. For more information regarding kickstart options, see the documentation located at http://www.redhat.com/docs/manuals/enterprise/ From there, click **Installation Guide**, then click **28. Kickstart Installations**.

# 11

# Servicing Linux with Red Hat Network

This chapter describes Red Hat Network (RHN) and its ability to manage the virtual servers. Using the **yum** command, the virtual servers can be updated when Red Hat errata are released. You can also use **yum** to install new packages with automatic dependency resolution. RHN is accessed by the following link:

http://rhn.redhat.com/

The following sections describe how to configure a Linux guest for **yum**, and manage the guest through RHN:

- Registering your system with RHN
- Installing and updating packages using yum
- Managing your Linux guest through RHN
- Updating a system without Internet access

## 11.1 Registering your system with RHN

This section assumes you have already obtained a valid entitlement for RHEL 5 on System z, or have completed the steps to obtain an evaluation copy. To receive a free 90-day evaluation, visit:

http://www.redhat.com/rhel/details/eval

Select the link named **Red Hat Enterprise Linux AS for IBM eServer™ zSeries and IBM S/390®** and create an account. Before using **yum** for the first time, you must import the Red Hat GPG key and register your Linux guest with RHN. Use the commands below, substituting your RHN user name, password, and host name of the Linux guest.

```
# rpm --import /usr/share/rhn/RPM-GPG-KEY
# rhnreg_ks --username=myuser --password=mypw --profilename=linux01.itso.ibm.com
```

## 11.2 Installing and updating packages using yum

Now that your system is registered with RHN, you can use **yum** to keep the system updated. You can download and install the latest version of a package by running **yum** with the RPM package name. You can also specify multiple packages on the command line separated by spaces. The **yum install** command installs the package if it is not present, and the **yum upgrade** command updates to the latest version if it is already installed. If a package has any dependencies, **yum** automatically downloads and installs them for you.

Update the cpp package to get the latest security fixes:

```
# rpm -q cpp
cpp-4.1.1-30
# yum upgrade cpp
Loading "rhnplugin" plugin
Loading "installonlyn" plugin
Setting up Upgrade Process
Setting up repositories
rhel-s390x-server-5-beta  100% |=========================|  950 B    00:00
...
=============================================================================
 Package                 Arch       Version          Repository        Size
=============================================================================
Updating:
 cpp                     s390x      4.1.1-43.el5     RHEL5             2.6 M

Transaction Summary
=============================================================================
Install      0 Package(s)
Update       1 Package(s)
Remove       0 Package(s)

Total download size: 2.6 M
Is this ok [y/N]: y
Downloading Packages:
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Updating  : cpp                            ######################### [1/2]
  Cleanup   : cpp                            ######################### [2/2]
```

```
Updated: cpp.s390x 0:4.1.1-43.el5
Complete!
```

Now query the `cpp` package and you should see that it has been updated.

```
# rpm -q cpp
cpp-4.1.1-43.el5
```

To update every installed package on the system, run:

```
# yum upgrade
```

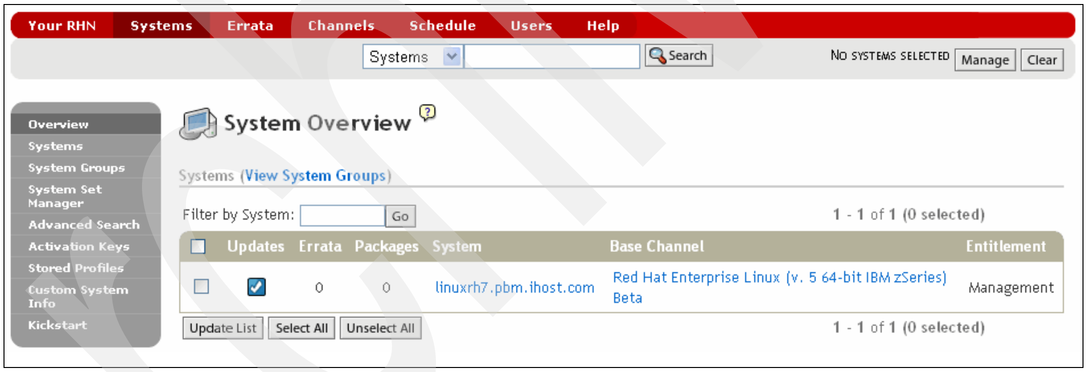
For more information about the **`yum`** command see the yum(8) man page.

## 11.3 Managing your Linux guest through RHN

You can also manage the packages on this Linux guest through the Web interface at:

http://rhn.redhat.com/

When you first log in to RHN, you see the system you registered under the **Systems** tab. If there is a red exclamation point next to your system, it means there is errata waiting to be applied. The number of relevant errata and the corresponding number of packages are visible to the left of the system name. Click the number beneath Errata or Packages to get a detailed list. If there is a blue check-mark, then the system is fully updated.



System Overview

Systems (View System Groups)

Filter by System:

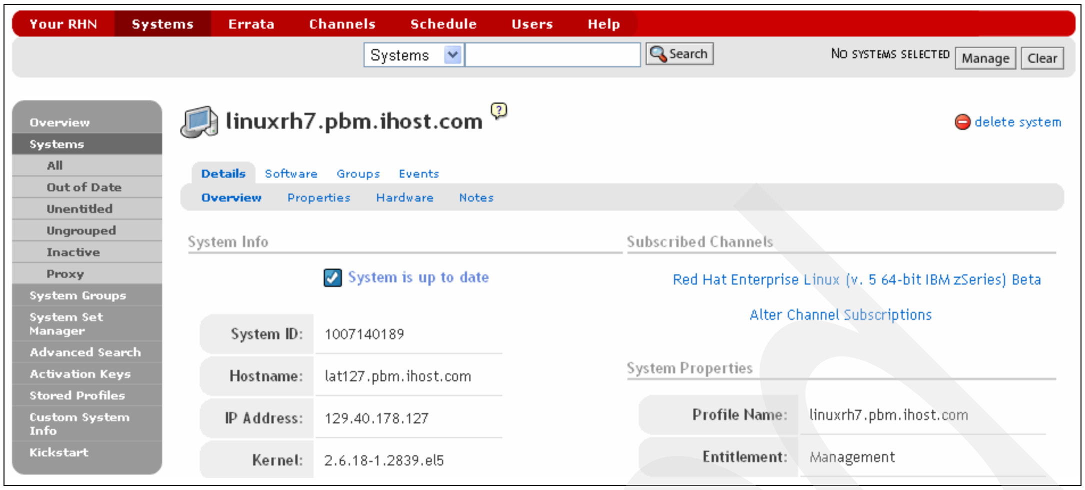
1 - 1 of 1 (0 selected)

| □ | Updates | Errata | Packages | System | Base Channel | Entitlement |
|---|---|---|---|---|---|---|
| □ | ☑ | 0 | 0 | linuxrh7.pbm.ihost.com | Red Hat Enterprise Linux (v. 5 64-bit IBM zSeries) Beta | Management |

1 - 1 of 1 (0 selected)

*Figure 11-1 RHN system overview*

Next, click the link that is the system name. This brings you to a detailed overview, where you can see the system properties as Figure 11-2 on page 156 shows. Click the **Packages** tab to view all packages installed on this system. From this tab, you can also update, remove, or install new packages onto the system.



linuxrh7.pbm.ihost.com

System Info

☑ System is up to date

| | |
|---|---|
| System ID: | 1007140189 |
| Hostname: | lat127.pbm.ihost.com |
| IP Address: | 129.40.178.127 |
| Kernel: | 2.6.18-1.2839.el5 |

Subscribed Channels

Red Hat Enterprise Linux (v. 5 64-bit IBM zSeries) Beta

System Properties

| | |
|---|---|
| Profile Name: | linuxrh7.pbm.ihost.com |
| Entitlement: | Management |

*Figure 11-2 RHN system details*

For more information about managing your systems through RHN, including usage guides and frequently asked questions, see:

http://rhn.redhat.com/help

## 11.4 Updating a system without Internet access

The **yum** command communicates with RHN using secure Web traffic. If your Linux guest is not connected to the Internet, or is behind a firewall that prevents access to RHN, you can still use all of the **yum** functionality.

### 11.4.1 Configuring the server

This process uses the controller as a **yum** repository. The controller exports the RHEL 5.2 install tree over NFS. Your Linux guest uses the automounter to access this install tree, which **yum** uses instead of RHN.

For more information about exporting the install tree over NFS, refer to 7.2.9, "Turning on the NFS server" on page 115. The following sections assume the install tree is exported over NFS as /nfs/rhel5.2/.

### 11.4.2 Configuring the client

On the Linux guest, you will first configure the automounter to mount the installation tree from the controller on demand. The automounter will automatically mount a remote directory when it is accessed, and automatically unmount it after a period of inactivity. Edit the file /etc/auto.master and add the following line at the bottom:

```
# vi /etc/auto.master
...
/nfs    /etc/auto.controller
```

Next, create the file /etc/auto.controller and add the following line, substituting the IP address of your controller:

```
# vi /etc/auto.controller
rhel5.2         -ro,hard,intr   <9.12.5.30>:/nfs/rhel5.2
```

Create the /nfs directory. Restart the **autofs** service to pick up the configuration changes, then list the contents of the automounted directory:

```
# mkdir /nfs
# service autofs restart
Stopping automount: [  OK  ]
Starting automount: [  OK  ]
# ls /nfs/rhel5.2
EULA            README-or.html         RELEASE-NOTES-ja.html
eula.en_US      README-pa.html         RELEASE-NOTES-ko.html
...
```

Now that the installation tree is accessible, you will configure **yum** to use it. Create a file named rhel5.repo in the /etc/yum.repos.d/ directory:

```
# vi /etc/yum.repos.d/rhel5.2.repo
[RHEL5.2]
name=Red Hat Enterprise Linux 5.2
baseurl=file:///nfs/rhel5.2/Server
```

Finally, you need to import the RPM GPG key so that **yum** knows you are installing official Red Hat packages. The Red Hat GPG key is located in the install tree. Import the key with:

```
# rpm --import /nfs/rhel5/RPM-GPG-KEY
```

**Note:** Red Hat signs each RPM with a private GPG key, which is compared to your public key each time a package is installed. This method ensures that the RPM is a genuine, unaltered package. When installing an RPM, if you ever see a message similar to the following, it means that either the correct GPG key has not been imported, or the package itself has been altered by a third party:

```
Header V3 DSA signature: NOKEY, key ID 897da07a
```

You are now ready to use **yum** to install or upgrade an RPM package. Refer to 11.2, “Installing and updating packages using yum” on page 154 for an explanation of **yum** syntax.

### 11.4.3 Updating packages on the server

The RPMs are located in the /Server subdirectory of the install tree. Beneath this directory, there is a subdirectory named /repodata which contains repository files used by **yum**. Whenever a new set of RPMs is added to the /Server directory, you must rebuild the repository data with the **createrepo** command.

On the controller, first install the **createrepo** RPM with:

```
# yum install createrepo
...
Running Transaction
  Installing: createrepo                ######################### [1/1]

Installed: createrepo.noarch 0:0.4.4-2.fc6
Complete!
```

Next, back up the current repository and create a new one with the **createrepo** command:

```
# cd /nfs/rhel5.2/Server
# mv repodata repodata.orig
# createrepo /nfs/rhel5.2/Server
96/2480 - system-config-network-1.3.97-1.el5.noarch.rpm
...
```

It should take about five minutes to build the new repository data, depending on the number of RPMs.

You now have a new repository based on the RPMs in `/nfs/rhel5.2/Server`. Repeat this procedure any time an RPM (or set of RPMs) change. You can also use this to create multiple RPM repositories, based on anything you like. For example, you can have a repository named "test" containing the latest packages from RHN, as well as a "production" repository based on an official RHEL update. You can then create a `.repo` file on the client pointing to either repository, based on the role of that client.

For example, on the server:

```
# cd /nfs/rhel5.2
# cp -a Server Server.production
# cp -a Server Server.test
```

Now, any time you update an RPM in `/nfs/rhel5.2/Server.test/`, run **createrepo** on that directory. On the client, in the `/etc/yum.repos.d/` directory you can have:

```
/etc/yum.repos.d/rhel5.2.production.repo:
[RHEL5.2]
name=RHEL 5.2 production
baseurl=file:///nfs/rhel5.2/Server.production/
```

Or you can have:

```
/etc/yum.repos.d/rhel5.test.repo:
[RHEL5.2]
name=RHEL 5.2 test
baseurl=file:///nfs/rhel5.2/Server.test/
```

Note that this method requires maintaining separate copies of each directory of RPMs, which can fill the `/nfs` directory on the controller quickly. Red Hat provides a separate product named Satellite which maintains a single tree, while managing the RPM version differences. The Satellite provides other features similar to RHN, as well; for more information, see:

https://www.redhat.com/rhn/rhndetails/architecture/

12

# Cloning open source virtual servers

This chapter describes how to clone and customize the following Linux virtual servers:

- Creating a virtual Web server
- Creating a virtual LDAP server
- Creating a virtual file and print server
- Creating a virtual application development server

Conceptual discussions and detailed explanations about these types of servers is beyond the scope of this book. Instead, use the following sections as a reference to get the servers quickly installed and configured.

# 12.1 Creating a virtual Web server

The example in this section uses the `LINUX01` user ID to create a virtual Web server. You should have a vanilla virtual server cloned to the user ID `LINUX01` as described in Chapter 9, “Configuring RHEL 5.2 for cloning” on page 129.

## 12.1.1 Installing Apache RPMs

**SSH** into the IP address of the new `LINUX01` server. Install the following Apache RPMs using the **`yum -y install`** command. The **`-y`** flag prevents the `Is this OK?` question from being presented.

```
# yum -y install httpd httpd-manual
...
=============================================================================
 Package                 Arch       Version          Repository        Size
=============================================================================
Installing:
 httpd                   s390x      2.2.3-6.el5      RHEL5             1.1 M
 httpd-manual            s390x      2.2.3-6.el5      RHEL5             831 k
Installing for dependencies:
 apr                     s390x      1.2.7-10         RHEL5             122 k
 apr-util                s390x      1.2.7-3          RHEL5              78 k
 postgresql-libs         s390x      8.1.4-1.1        RHEL5             197 k

Transaction Summary
=============================================================================
Install      5 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total download size: 2.3 M
Downloading Packages:
Running Transaction Test
warning: httpd-manual-2.2.3-6.el5: Header V3 DSA signature: NOKEY, key ID 897da07a
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: apr                        ######################### [1/5]
  Installing: postgresql-libs            ######################### [2/5]
  Installing: apr-util                   ######################### [3/5]
  Installing: httpd                      ######################### [4/5]
  Installing: httpd-manual               ######################### [5/5]

Installed: httpd.s390x 0:2.2.3-6.el5 httpd-manual.s390x 0:2.2.3-6.el5
Dependency Installed: apr.s390x 0:1.2.7-10 apr-util.s390x 0:1.2.7-3
postgresql-libs.s390x 0:8.1.4-1.1
Complete!
```

Verify that the RPMs were installed:

```
# rpm -qa | grep httpd
httpd-manual-2.2.3-11.el5_1.3
httpd-2.2.3-11.el5_1.3
```

Before starting the Apache Web server, use the **`chkconfig`** command to set the service to start at boot time:

```
# chkconfig --list httpd
```

```
httpd           0:off    1:off    2:off    3:off    4:off    5:off    6:off
# chkconfig httpd on
# chkconfig --list httpd
httpd           0:off    1:off    2:on     3:on     4:on     5:on     6:off
```

## 12.1.2 Testing Apache

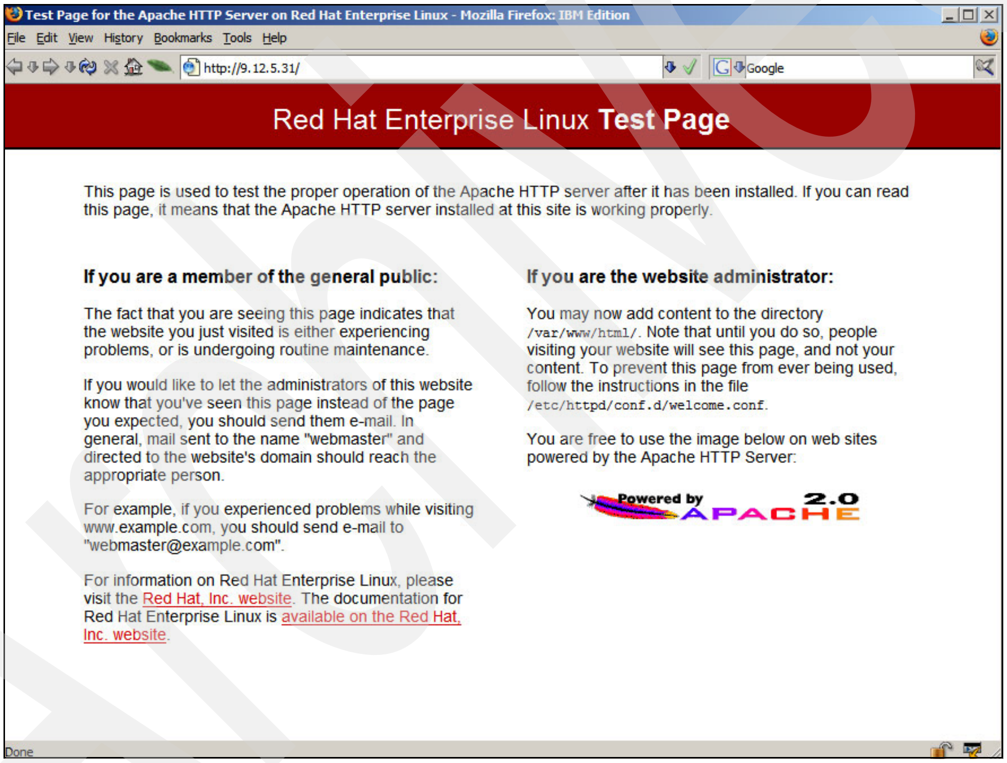
Start the Apache Web server to verify that it is installed successfully:

```
# service httpd start
Starting httpd: [  OK  ]
```

To verify that Apache is installed correctly, after it has been started, bring up a Web browser and point it to the server. For example, the virtual server running on LINUX01 can be reached with the following URL:

```
http://<9.12.5.31>/
```

Use the test page shown in Figure 12-1 to verify that the Web server is working.


Test Page for the Apache HTTP Server on Red Hat Enterprise Linux - Mozilla Firefox: IBM Edition
File Edit View History Bookmarks Tools Help
http://9.12.5.31/
Google
Done

Red Hat Enterprise Linux **Test Page**

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

**If you are a member of the general public:**

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

For information on Red Hat Enterprise Linux, please visit the Red Hat, Inc. website. The documentation for Red Hat Enterprise Linux is available on the Red Hat, Inc. website.

**If you are the website administrator:**

You may now add content to the directory /var/www/html/. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file /etc/httpd/conf.d/welcome.conf.

You are free to use the image below on web sites powered by the Apache HTTP Server:

*Figure 12-1 Apache test page*

If you get an error in starting Apache, look in the log file /var/log/httpd/error-log for clues. If Apache started successfully but you cannot reach the test page from a browser, try accessing it using the IP address rather than the DNS name.

## 12.1.3 Turning on a firewall

RHEL 5.2 comes with an IP tables firewall which is disabled after a default installation. Your enterprise may have firewalls in front of Web servers, but you may also want a firewall running on this virtual server. This section describes how to quickly enable an IP tables firewall and configure it to allow Web traffic through.

Verify that the firewall is off using the **chkconfig --list** command. The service name is **iptables**:

```
# chkconfig --list iptables
iptables        0:off   1:off   2:off   3:off   4:off   5:off   6:off
```

Turn on the firewall at boot time with the **chkconfig** command, and for this session with the **service** command:

```
# chkconfig iptables on
# service iptables start
Applying iptables firewall rules: [  OK  ]
Loading additional iptables modules: ip_conntrack_netbios_ns [  OK  ]
```

Go back to your browser and click **refresh**. You should get an error that the server is not responding. This is because packets for ports for http: and https: (80 and 443) are dropped by default.

To allow Web traffic through, you can modify the file /etc/sysconfig/iptables. First make a backup copy, add two rules (shown in bold font) to allow these ports, and then save your changes:

```
# cd /etc/sysconfig
# cp iptables iptables.orig
# vi iptables
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A FORWARD -j RH-Firewall-1-INPUT
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp --icmp-type any -j ACCEPT
-A RH-Firewall-1-INPUT -p 50 -j ACCEPT
-A RH-Firewall-1-INPUT -p 51 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp --dport 5353 -d 224.0.0.251 -j ACCEPT
-A RH-Firewall-1-INPUT -p udp -m udp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 631 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

Restart the firewall to pick up the new rules:

```
# service iptables restart
Flushing firewall rules: [  OK  ]
Setting chains to policy ACCEPT: filter [  OK  ]
Unloading iptables modules: [  OK  ]
Applying iptables firewall rules: [  OK  ]
Loading additional iptables modules: ip_conntrack_netbios_ns [  OK  ]
```

Go back to your browser and click **refresh** again. You should not get an error this time. You should now have a firewall that allows Web traffic.

### 12.1.4 Configuring SSL for Apache

Use the Secure Sockets Layer (SSL) to encrypt data between the client (browser) and the server. This is done by specifying an https prefix in the URL which uses port 443, rather than using the conventional http prefix which uses port 80.

To use SSL, the mod_ssl package is required. You can demonstrate that SSL communications do not work by changing `http` to `https` in your browser:

```
https://<9.12.5.31>/
```

Click **reload** in your browser; you should receive a communications error.

#### Installing the SSL Apache module

RHEL 5.2 makes it very easy to install and enable SSL support for Apache. Install the mod_ssl RPM using the **`yum -y install`** command:

```
# yum -y install mod_ssl
...
Install      2 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total download size: 212 k
Is this ok [y/N]: y
Downloading Packages:
Running Transaction Test
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: distcache                    ######################### [1/2]
  Installing: mod_ssl                      ######################### [2/2]

Installed: mod_ssl.s390x 1:2.2.3-6.el5
Dependency Installed: distcache.s390x 0:1.4.5-14.1
Complete!
```

Verify that the RPM was added:

```
# rpm -qa | grep ssl
openssl-0.9.8b-10.el5
openssl-0.9.8b-10.el5
mod_ssl-2.2.3-11.el5_1.3
```

Restart the Web server:

```
# service httpd restart
Stopping httpd: [  OK  ]
Starting httpd: [  OK  ]
```

Go back to your browser and click **restart** again. This time you should get a warning about a self-signed certificate. This is acceptable, because for a production Web site you will want a certificate signed by a certificate authority.

### 12.1.5 Populating your Web site

You can begin to put your Web pages in the directory `/var/www/html/` which is the default Web root.

### 12.1.6 Apache resources

The following Web sites contain additional information about Apache:

http://www.samspublishing.com/articles/article.asp?p=30115&seqNum=4
http://www.sitepoint.com/article/securing-apache-2-server-ssl
http://www.securityfocus.com/infocus/1786

## 12.2 Creating a virtual LDAP server

The Lightweight Directory Access Protocol (LDAP) is commonly implemented using the OpenLDAP package, which comes standard with most Linux distributions. Among other directory functions, OpenLDAP allows for centralized login authentication and user and group ID resolution.

In this section you will install Linux manually and set up login authentication to a new virtual LDAP server. Then you will go back to the virtual Web server you just created and point it to the new LDAP server.

The following steps are documented in this section:

- Cloning a Linux virtual server
- Installing the OpenLDAP server
- Configuring the OpenLDAP server
- Migrating existing users to LDAP
- Configuring an LDAP client
- Adding new user accounts

### 12.2.1 Cloning a Linux virtual server

Start an SSH session as root to the controller and clone the golden image to the LINUX02 user ID:

```
# clone rhel52 linux02 100-101
Invoking CP command: QUERY rhel52
Invoking CP command: QUERY linux02

This will copy disks from rhel52 to linux02
Host name will be:  virtc532.itso.ibm.com
IP address will be: 9.12.5.32
Do you want to continue? (y/n): y
...
Booting linux02
Successfully cloned rhel52 to linux02
```

You should now have a fresh RHEL 5.2 system running on LINUX02.

### 12.2.2 Installing the OpenLDAP server

Start an SSH session to the IP address of the new virtual server running on LINUX02. Use the **yum** command to install the OpenLDAP client and server RPMs:

```
# yum -y install openldap-clients openldap-servers
...
=============================================================================
 Package                 Arch       Version          Repository        Size
```

```
=============================================================================
Installing:
 openldap-clients        s390x      2.3.27-4         RHEL5            189 k
 openldap-servers        s390x      2.3.27-4         RHEL5            2.3 M
Installing for dependencies:
 libtool-ltdl            s390x      1.5.22-6.1       RHEL5             37 k

Transaction Summary
=============================================================================
Install      3 Package(s)
Update       0 Package(s)
Remove       0 Package(s)

Total download size: 2.5 M
Downloading Packages:
Running Transaction Test
warning: openldap-clients-2.3.27-4: Header V3 DSA signature: NOKEY, key ID 897da07a
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: libtool-ltdl                 ######################### [1/3]
  Installing: openldap-clients             ######################### [2/3]
  Installing: openldap-servers             ######################### [3/3]

Installed: openldap-clients.s390x 0:2.3.27-4 openldap-servers.s390x 0:2.3.27-4
Dependency Installed: libtool-ltdl.s390x 0:1.5.22-6.1
Complete!
```

### 12.2.3 Configuring the OpenLDAP server

A detailed description of LDAP is beyond the scope of this book. Instead, brief configuration recommendations are given in this section.

There are two important configuration values that must be chosen.

- The suffix or base distinguished name of the LDAP Domain Information Tree (DIT) - the most common suffix is to use your company’s DNS name.
- The LDAP administrator or root name and password.

Choose an administrative password and run the **slappasswd** command, which displays an encrypted version of it. The output of this command will be used in a configuration file, so you may want to make a copy of it.

```
# slappasswd
New password: <lnx4vm>
Re-enter new password: <lnx4vm>
{SSHA}4FiGwLm+cy+I96TyiWMn4evNXSXa5aJ2
```

The OpenLDAP server configuration file is /etc/openldap/slapd.conf file. Make a backup copy of it then modify the original. First set the suffix and root user distinguished name (rootdn):

```
# cd /etc/openldap
# cp slapd.conf slapd.conf.orig
# vi slapd.conf
...
database        bdb
# suffix                "dc=my-domain,dc=com"
# rootdn                "cn=Manager,dc=my-domain,dc=com"
suffix "dc=itso,dc=ibm,dc=com"
```

```
rootdn "cn=ldaproot,dc=itso,dc=ibm,dc=com"
...
```

Set the **rootpw** line to a value of the encrypted password that was the output of the **slappasswd** command:

```
...
# rootpw                secret
# rootpw                {crypt}ijFYNcSNctBYg
rootpw {SSHA}4FiGwLm+cy+I96TyiWMn4evNXSXa5aJ2
...
```

Save the slapd.conf file. Your LDAP server should now be minimally configured.

### Start the LDAP service

Start LDAP at boot time using the **chkconfig** command and, for this session, using the **service** command:

```
# chkconfig ldap on
# service ldap start
Checking configuration files for slapd:  config file testing succeeded         [  OK  ]
Starting slapd: [  OK  ]
```

Query the LDAP database using the **ldapsearch** command. The **-x** flag specifies that simple authentication is used:

```
# ldapsearch -x
# extended LDIF
#
# LDAPv3
# base <> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# search result
search: 2
result: 32 No such object
```

The result shows that the LDAP directory can be searched, but that it is empty. This is expected because no data has been added to it.

## 12.2.4 Migrating existing users to LDAP

A common method of populating the LDAP database is using LDAP Directory Interchange Format (LDIF) files. Red Hat includes a set of migration scripts with the OpenLDAP server RPM, written by PADL Software Pty Ltd (http://padl.com). These scripts allow you to convert users and groups from the /etc/ directory to LDIF format. The scripts are in the /usr/share/openldap/migration/ directory. Before converting the /etc/group and /etc/passwd files, add a user that will be used to test LDAP authentication. In this example, ldapuser1 is added using the **useradd** command and the password is set using the **passwd** command:

```
# useradd ldapuser1
# passwd ldapuser1
Changing password for user ldapuser1.
New UNIX password: <lnx4vm>
Retype new UNIX password: <lnx4vm>
passwd: all authentication tokens updated successfully.
```

The PADL tools have a shared configuration file named `migrate_common.ph`. Make a backup copy of this file and modify it. There are two lines to be changed which set the domain name and suffix (or root) of LDAP tree. In this example, the suffix is `itso.ibm.com`:

```
# cd /usr/share/openldap/migration
# cp migrate_common.ph migrate_common.ph.orig
# vi migrate_common.ph
...
# Default DNS domain
$DEFAULT_MAIL_DOMAIN = "<itso.ibm.com>";

# Default base
$DEFAULT_BASE = "<dc=itso,dc=ibm,dc=com>";
...
```

The first tool conversion Perl script is **`migrate_base.pl`**. Redirect the output to a file named `accounts.ldif`. This will be the file that you use to populate the LDAP server. After you run it, take a look at the first few lines of the LDIF file using the **`head`** command to give you an idea of the format:

```
# ./migrate_base.pl > accounts.ldif
# head -9 accounts.ldif
dn: dc=ibm,dc=com
dc: ibm
objectClass: top
objectClass: domain

dn: dc=itso,dc=ibm,dc=com
dc: itso
objectClass: top
objectClass: domain
```

**Important:** There appears to be a bug in `migrate_base.pl` when the suffix has more than two components. This will later result in an error when you try to import the LDIF file:

```
slapadd: line 5: database (dc=itso,dc=ibm,dc=com) not configured to hold
"dc=ibm,dc=com"
```

If your suffix (Base DN) has more than two components, you can work around the bug by deleting the entry or entries in the LDIF file with fewer components than your suffix. In this example, itso.ibm.com has three components, so the first entry (first four lines and one blank line) in the LDIF file must be deleted. Then the **`slapadd`** command should succeed.

The next two scripts migrate the `/etc/passwd` and `/etc/group` files. Append the output of both of these to the `accounts.ldif` file using the **>>** redirection operator:

```
# ./migrate_passwd.pl /etc/passwd >> accounts.ldif
# ./migrate_group.pl /etc/group >> accounts.ldif
```

Search for the string ldapuser1 in the `accounts.ldif` file:

```
# grep ldapuser1 accounts.ldif
dn: uid=ldapuser1,ou=People,dc=itso,dc=ibm,dc=com
uid: ldapuser1
cn: ldapuser1
homeDirectory: /home/ldapuser1
dn: cn=ldapuser1,ou=Group,dc=itso,dc=ibm,dc=com
cn: ldapuser1
```

The output shows that the user was migrated to the LDIF file.

Now it is time to populate the LDAP server. This can be done while the server is not running using the **slapadd -l** command. First stop the LDAP service, then add the data in the LDIF file:

```
# service ldap stop
Stopping slapd: [  OK  ]
# slapadd -l accounts.ldif
bdb_db_open: Warning - No DB_CONFIG file found in directory /var/lib/ldap: (2)
Expect poor performance for suffix dc=itso,dc=ibm,dc=com.
```

The database is stored in the directory `/var/lib/ldap/`. For the OpenLDAP server to start, all files in this directory must be owned by the `ldap` user and group. This can be accomplished using the **chown** command:

```
# chown ldap:ldap /var/lib/ldap/*
```

Set the LDAP server:

```
# service ldap start
Checking configuration files for slapd:  bdb_db_open: Warning - No DB_CONFIG file found
in directory /var/lib/ldap: (2)
config file testing succeeded
[  OK  ]
Starting slapd: [  OK  ]
```

The message `Expect poor performance for suffix dc=itso,dc=ibm,dc=com.` is left as an exercise for the reader. Test that you can query the LDAP directory for `ldapuser1` using the **ldapsearch** command. The **-b** flag sets the base of the search (it is necessary because the default suffix of `dc=itso,dc=ibm,dc=com` has not been configured into the LDAP client; later it will not be needed):

```
# ldapsearch -x -b dc=itso,dc=ibm,dc=com uid=ldapuser1
# extended LDIF
#
# LDAPv3
# base <dc=itso,dc=ibm,dc=com> with scope subtree
# filter: uid=ldapuser1
# requesting: ALL
#

# ldapuser1, People, itso.ibm.com
dn: uid=ldapuser1,ou=People,dc=itso,dc=ibm,dc=com
uid: ldapuser1
...
# numEntries: 1
```

You should now have an OpenLDAP server installed, configured, and populated with users and groups.

## Deleting duplicate users

You now have a duplicate set of users and groups in both the `/etc/` file system and in LDAP. There are different ways that you can manage this duplication. For this section, the following changes are recommended:

- Delete ldapuser1 from file system: this user was designed to be the first LDAP uid so it should not also be in the file system
- Delete root from LDAP: having uid of 0 (root) in LDAP is may not be a good security design.

Delete the ldapuser1 from the file system with the **userdel** command, and delete root from LDAP using the **ldapdelete** command:

```
# userdel ldapuser1
# ldapdelete -x -D cn=ldaproot,dc=itso,dc=ibm,dc=com -W \
uid=root,ou=People,dc=itso,dc=ibm,dc=com
Enter LDAP Password: <lnx4vm>
```

Now ldapuser1’s credentials are only in LDAP and the root password is only in the local file system.

## 12.2.5 Configuring an LDAP client

You are now ready to configure a system to authenticate users using the new LDAP server. You will first go to a different virtual server, running on the LINUX01 user ID, and configure it to point to this LDAP server.

Start an SSH session to the Web server running on LINUX01.

Invoke the command **authconfig-tui**. Use the **Tab** key to move between fields, the **space bar** to change selections and the **Enter** key to select:

```
# authconfig-tui
```

```
+----------------¦ Authentication Configuration +-----------------+
¦                                                                 ¦
¦  User Information        Authentication                         ¦
¦  [*] Cache Information   [*] Use MD5 Passwords                  ¦
¦  [ ] Use Hesiod          [*] Use Shadow Passwords               ¦
¦  [*] Use LDAP            [*] Use LDAP Authentication            ¦
¦  [ ] Use NIS             [ ] Use Kerberos                       ¦
¦  [ ] Use Winbind         [ ] Use SMB Authentication             ¦
¦                          [ ] Use Winbind Authentication         ¦
¦                          [*] Local authorization is sufficient  ¦
¦                                                                 ¦
¦            +--------+                     +------+              ¦
¦            ¦ Cancel ¦                     ¦ Next ¦              ¦
¦            +--------+                     +------+              ¦
¦                                                                 ¦
¦                                                                 ¦
+-----------------------------------------------------------------+
```

Under User Information, select **Cache Information** and **Use LDAP**. Under Authentication, select **Use LDAP Authentication** and **Local authorization is sufficient**. Press **Enter** when you have tabbed to the **Next** button.

```
+-----------------¦ LDAP Settings +-----------------+
¦                                                   ¦
¦          [ ] Use TLS                              ¦
¦  Server: ldap://<9.12.5.32>/____________________  ¦
¦ Base DN: <dc=itso,dc=ibm,dc=com>________________  ¦
¦                                                   ¦
¦         +------+              +----+              ¦
¦         ¦ Back ¦              ¦ Ok ¦              ¦
¦         +------+              +----+              ¦
¦                                                   ¦
¦                                                   ¦
¦                                                   ¦
+---------------------------------------------------+
```

On the next window, set the Server value to point to the LDAP server. In this example, it is `ldap://9.12.5.32/`. Set the `Base DN` to your suffix value. In this example it is `dc=itso,dc=ibm,dc=com`. Press **OK**.

Your LDAP client should now be pointing to the LDAP server. Test it using the **`id ldapuser1`** command:

```
# id ldapuser1
uid=500(ldapuser1) gid=500(ldapuser1) groups=500(ldapuser1)
context=root:system_r:unconfined_t:s0-s0:c0.c1023
```

Authentication should also allow LDAP. Test this by starting another SSH session to the LDAP server and logging in as ldapuser1:

```
login as: ldapuser1
ldapuser1@9.12.5.32's password:
```

When you get an SSH session as `ldapuser1`, this shows that authentication is working.

## 12.2.6 Adding new user accounts

There are different front-ends to LDAP administration, however, there is no industry standard. A detailed discussion about front-end tools is beyond the scope of this book. However, it is useful to be able to add users using line commands. This section describes how to create a new LDIF file and add a new user.

Go back to the directory `/usr/share/openldap/` and view the LDIF file you created earlier, `accounts.ldif` using the **`less`** command. Search for the user ldapuser1 using the **`/`** subcommand:

```
# cd /usr/share/openldap/migration
# less accounts.ldif
/ldapuser1
```

Before:

```
dn: uid=ldapuser1,ou=People,dc=itso,dc=ibm,dc=com
uid: ldapuser1
cn: ldapuser1
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
shadowLastChange: 13515
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 500
gidNumber: 500
homeDirectory: /home/ldapuser1
```

Copy and paste the lines for that user to a new file `ldapuser.ldif`. Change the name ldapuser1 to the desired new name (ldapuser2, in this example). Also increment the uidNumber field (to 501, in this example)

```
# vi ldapuser.ldif
```

After:

```
dn: uid=ldapuser2,ou=People,dc=itso,dc=ibm,dc=com
uid: ldapuser2
cn: ldapuser2
```

```
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
shadowLastChange: 13515
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 501
gidNumber: 500
homeDirectory: /home/ldapuser2
```

You now have created a new LDIF file containing the entries for a new LDAP user with a unique UID. Now you can add the user to the LDAP database using the following **ldapadd** command:

```
# ldapadd -x -D cn=ldaproot,dc=itso,dc=ibm,dc=com -W -f ldapuser.ldif
Enter LDAP Password:

adding new entry "cn=ldapuser2,ou=Group,dc=itso,dc=ibm,dc=com"
```

### Set a password

You can set the new user's password using the following **ldappasswd** command:

```
# ldappasswd -x -D cn=ldaproot,dc=itso,dc=ibm,dc=com -W -s lnx4vm \
uid=ldapuser2,ou=People,dc=itso,dc=ibm,dc=com
Enter LDAP Password: <lnx4vm>

Result: Success (0)
```

You can test logging into a virtual server with the new user's credentials. If you successfully configured the virtual server running on LINUX01 to point to the LDAP server, try a new SSH session with the new user. You should be able to successfully login:

```
login as: ldapuser2
ldapuser2@9.12.5.32's password:
Could not chdir to home directory /home/ldapuser2: No such file or directory
```

The issue of not having a home directory is addressed in 13.3, "Centralizing home directories for LDAP users" on page 187.

## 12.3 Creating a virtual file and print server

Samba allows Windows clients to map Linux file systems as shared drives. Samba can also act as a "middleman" between Windows clients and a Linux print server. The recommended Linux print server is the Common UNIX Printing System. (CUPS). A detailed description of the configuration of CUPS is beyond the scope of this book. However, this section describes how the necessary RPMs are installed.

The following steps are discussed in this section:

- Cloning a Linux virtual server
- Installing the necessary RPMs
- Configuring the Samba configuration file
- Adding a Samba user
- Starting Samba at boot time
- Testing your changes

## 12.3.1 Cloning a Linux virtual server

From the controller, clone a basic virtual server. In this example, the user ID `LINUX03` is used.

```
# clone rhel52 linux03 100-101
Invoking CP command: QUERY rhel52
Invoking CP command: QUERY linux02

This will copy disks from rhel52 to linux02
Host name will be: virtc533.itso.ibm.com
IP address will be: 9.12.5.33
Do you want to continue? (y/n): y
...
Booting linux03
Successfully cloned rhel52 to linux03
```

Start an SSH session to the new virtual server.

## 12.3.2 Installing the necessary RPMs

Add the following RPMs using the **`yum -y`** command:

```
# yum -y install samba-common samba samba-client
...
=============================================================================
 Package                 Arch       Version          Repository        Size
=============================================================================
Installing:
 samba                   s390x      3.0.23c-2        RHEL5              16 M
 samba-client            s390x      3.0.23c-2        RHEL5             4.6 M
 samba-common            s390       3.0.23c-2        RHEL5             9.2 M
 samba-common            s390x      3.0.23c-2        RHEL5             9.1 M
Installing for dependencies:
 popt                    s390       1.10.2-36.el5    RHEL5              69 k

Transaction Summary
=============================================================================
Install      5 Package(s)
Update       0 Package(s)
Remove       0 Package(s)
Total download size: 39 M
Downloading Packages:
Running Transaction Test
warning: samba-client-3.0.23c-2: Header V3 DSA signature: NOKEY, key ID 897da07a
Finished Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing: samba-common                ######################### [1/5]
  Installing: popt                        ######################### [2/5]
  Installing: samba-client                ######################### [3/5]
  Installing: samba                       ######################### [4/5]
  Installing: samba-common                ######################### [5/5]

Installed: samba.s390x 0:3.0.23c-2 samba-client.s390x 0:3.0.23c-2 samba-common.s390
0:3.0.23c-2 samba-common.s390x 0:3.0.23c-2
Dependency Installed: popt.s390 0:1.10.2-36.el5
Complete!
```

Confirm that the RPMs were added:

```
# rpm -qa | grep samba
samba-common-3.0.28-0.el5.8
samba-client-3.0.28-0.el5.8
samba-3.0.28-0.el5.8
samba-common-3.0.28-0.el5.8
```

## 12.3.3 Configuring the Samba configuration file

The one configuration file for Samba is `/etc/samba/smb.conf`. It is easy to add an SMB share that will be made available by the Samba server. A good test directory is `/usr/share/doc/` because it contains a significant amount of useful Linux documentation. The following example will create a file *share* named `sharedoc`:

```
# cd /etc/samba
# cp smb.conf smb.conf.orig
# vi smb.conf    // add three lines at the bottom of the file:
...
[sharedoc]
  comment = RHEL 5.2 on System z documentation
  path = /usr/share/doc/
```

You can verify the syntax of your changes with the testparm command:

```
# testparm smb.conf
Load smb config files from smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[sharedoc]"
Loaded services file OK.
Server role: ROLE_STANDALONE
Press enter to see a dump of your service definitions
...
```

This change will create an SMB share named **`sharedoc`** consisting of the contents of the directory `/usr/share/doc` and below.

## 12.3.4 Adding a Samba user

The default method that Samba uses to determine user credentials is to look in the `/etc/samba/smbpasswd` file. That user must first exist in the Linux file system (`/etc/passwd`, `/etc/shadow`, and so on).

To create a new Samba user, the **`smbpasswd -a`** command is used. First use the **`useradd`** and **`passwd`** commands to add a user locally. In this example, the user sambauser1 is used:

```
# id sambauser1
id: sambauser1: No such user
# useradd sambauser1
# passwd sambauser1
Changing password for sambauser1.
New password: <lnx4vm>
Re-enter new password: <lnx4vm>
Password changed
```

Add the user sambauser1 to the `smbpasswd` file using the **`smbpasswd -a`** command:

```
# smbpasswd -a sambauser1
New SMB password: <lnx4vm>
```

```
Retype new SMB password: <lnx4vm>

startsmbfilepwent_internal: file /etc/samba/smbpasswd did not exist. File successfully
created.
account_policy_get: tdb_fetch_uint32 failed for field 1 (min passwd length), returning 0
...
Added user sambauser1.
```

You can see that the last **smbpasswd** command added `sambauser1` to the file `smbpasswd`:

```
# cat smbpasswd
sambauser1:502:2E6F0C45D305054CAAD3B435B51404EE:74154D10CC18CFC98F1ED128DB8764DA:[U
]:LCT-45A3AAB2:
```

This method of maintaining Samba users, groups and passwords is useful for a small number of users. For a larger number of users, merging Samba and LDAP is recommended. It is not a simple as pointing the virtual file and print server at the virtual LDAP server as described in 12.2, “Creating a virtual LDAP server” on page 164 because the Samba schema must first be added to LDAP. Detailed information about this topic are beyond the scope of this book.

## 12.3.5 Starting Samba at boot time

Samba can be started for the current session using the **service** command and at boot time using the **chkconfig** command:

```
# service smb start
Starting SMB services: [  OK  ]
Starting NMB services: [  OK  ]
# chkconfig smb on
```

Samba should now be running and configured to start at boot time.

## 12.3.6 Testing your changes

You can verify that Samba is running using the following **service** command:

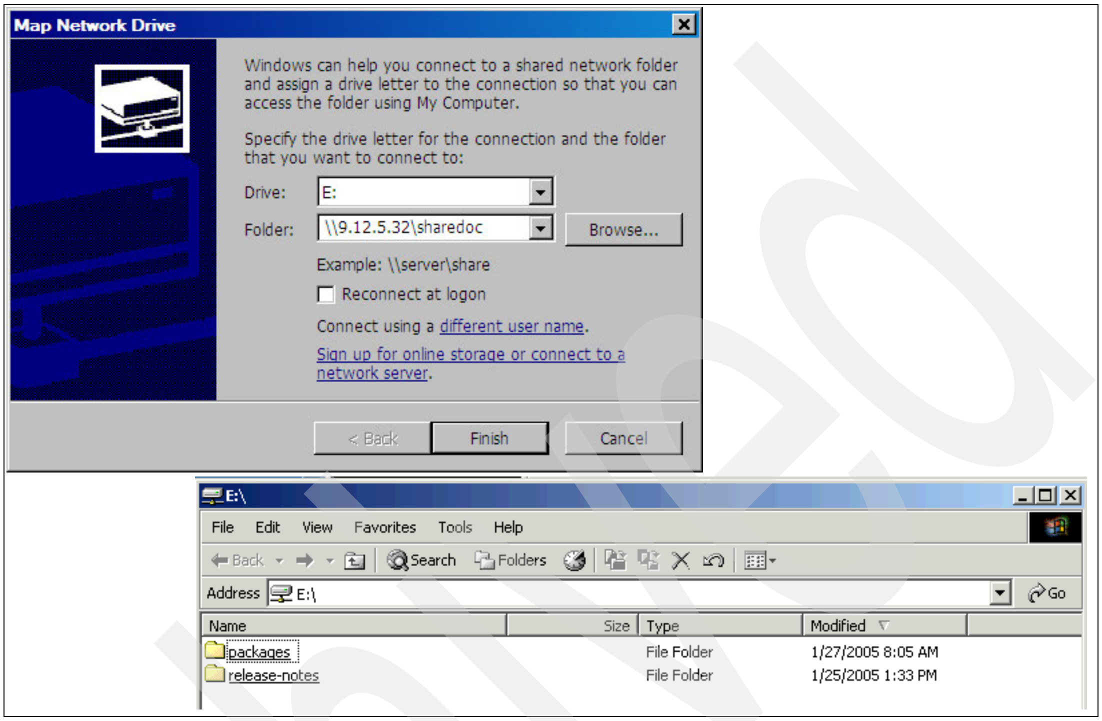
```
# service smb status
smbd (pid 6987 6982) is running...
nmbd (pid 6985) is running...
```

You can verify the shares that are available using the following **smbclient** command:

```
# smbclient -U sambauser1 -L localhost
Password:

Domain=[LAT123] OS=[Unix] Server=[Samba 3.0.23c-2]

        Sharename       Type      Comment
        ---------       ----      -------
        sharedoc        Disk      RHEL 5.2 on System z documentation
        IPC$            IPC       IPC Service (Samba Server)
        sambauser1      Disk      Home Directories
Domain=[LAT123] OS=[Unix] Server=[Samba 3.0.23c-2]
...
```

You can test getting a Samba share from a Windows desktop. Go to any Windows Explorer window (such as My Computer) and select **Tools** -> **Map Network Drive**. Use the Universal Naming Convention (UNC) to specify the Samba server and share name as shown in the upper left corner of Figure 12-2. In this example the UNC is **\\9.12.5.33\sharedoc**.

You may have to click **different user name** if the user or password on the new Samba server is different from the Windows system you are connecting from. Then click **Finish**. If all the steps were correct, you should see the files in a new Explorer window as shown in the bottom right corner of Figure 12-2.



Windows can help you connect to a shared network folder and assign a drive letter to the connection so that you can access the folder using My Computer.

Specify the drive letter for the connection and the folder that you want to connect to:

Drive: E:

Folder: \\9.12.5.32\sharedoc

Example: \\server\share

□ Reconnect at logon

Connect using a different user name.

Sign up for online storage or connect to a network server.

Address E:\

| Name | Size | Type | Modified |
|---|---|---|---|
| packages | | File Folder | 1/27/2005 8:05 AM |
| release-notes | | File Folder | 1/25/2005 1:33 PM |

*Figure 12-2 Mapping a network drive to the Samba server*

You should now have Samba configured and running with one new share available.

If you prefer a DOS command line, you can also link to the share using the following **`net use`** command:

```
c:\>net use y: \\9.12.5.33\sharedoc
The password is invalid for \\9.12.5.33\sharedoc.

Enter the user name for '9.12.5.33': sambauser1
Enter the password for 9.12.5.33:
The command completed successfully.
```

You can detach the share using the following **`net use`** command:

```
c:\>net use y: /delete
y: was deleted successfully.
```

## 12.3.7 Configuring printing

Configuring printing is more complex and a description of this procedure is beyond the scope of this book. For details about this topic, refer to the IBM Redpaper *Printing with Linux on zSeries Using CUPS and Samba*, REDP-3864, which is available on the Web at:

http://www.redbooks.ibm.com/abstracts/redp3864.html

# 12.4 Creating a virtual application development server

Most Linux distributions come with a basic set of application development tools, making Linux one of the most versatile development systems. These basic tools are ideal for projects of any size.

The development languages used in implementation range from scripting languages such as Python or Tcl, to compiled languages such as C/C++ and Java™. There is software available on Linux to help form a development system for developers to create integrated applications, and MySQL™ and Apache are among them. A popular open source Web platform is LAMP, which stands for the open source software and programming languages used to make up the platform: Linux, Apache, MySQL, Python or PHP.

- From the controller clone a virtual server to the user ID LINUX04.

```
# clone rhel52 linux04 100-101
...
```

- SSH in to the new virtual server.
- Note that before installing the development tools, the root file system is about half full:

```
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1           2.3G  1.1G  1.1G  49% /
tmpfs                 124M     0  124M   0% /dev/shm
```

- You can use the **yum -y groupinstall** command to install the groups named development-tools and development-libs. This will add about 181 packages, which requires a number of minutes to complete:

```
# yum -y groupinstall development-tools development-libs
...
Transaction Summary
=============================================================================
Install    181 Package(s)
Update       0 Package(s)
Remove       0 Package(s)
...
Complete!
```

- Your application development server is now ready to use. You may choose to add or remove different packages. Use **df -h** command to show that your system should still have about 30% free space:

```
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1           2.3G  1.5G  635M  71% /
tmpfs                 124M     0  124M   0% /dev/shm
```

## 12.4.1 Additional resources

The following Web sites are resources for additional information about application development topics:

### Scripting languages

http://www.perl.com/
http://www.python.org/
http://www.php.net/
http://www.tcl.tk/
http://www.freeos.com/guides/lsst/

## C/C++

http://gcc.gnu.org/onlinedocs/gcc/
http://en.wikipedia.org/wiki/GNU_Compiler_Collection#External_links
http://vertigo.hsrl.rutgers.edu/ug/make_help.htmsll
http://www.gnu.org/software/make/manual/html_chapter/make_toc.html

## Java

http://www-130.ibm.com/developerworks/java/
http://java.sun.com/
http://csdl.ics.hawaii.edu/~johnson/613f99/modules/04/jar-files.html
http://java.sun.com/j2se/1.3/docs/tooldocs/solaris/jdb.html

## Linux kernel development

http://www.kernel.org/pub/linux/docs/lkml/#blkd

## Web development

http://www.onlamp.com/
http://cgi.resourceindex.com/
http://www.perl.com/

# 13

# Miscellaneous recipes

This chapter has the following sections of miscellaneous tasks that you might want to perform:

- Adding a logical volume
- Extending an existing logical volume
- Centralizing home directories for LDAP users
- Rescuing a Linux system

# 13.1 Adding a logical volume

There are times when you require more disk space than a single direct access storage device (DASD) volume provides. For example, if you want to have a shared `/home/` directory, you will want it to be of sufficient size. When this is the case, you can use the Logical Volume Manager (LVM) to combine multiple DASD volumes into one logical volume.

The following process describes how to create a logical volume, or extend an existing logical volume with additional DASD on a Linux guest. The overall steps in creating a logical volume are as follows:

- Add minidisks to the z/VM directory entry and IPL Linux.
- Bring the new DASD online.
- Format and partition the DASD.
- Create the logical volume and file system.
- Update the file system table.
- Make the change persistent.

In the following sections, we discuss these steps in more detail.

## 13.1.1 Adding minidisks to the z/VM directory entry

A summary of the overall steps is given here. Specific details of these steps on z/VM are beyond the scope of this document.

- Determine the labels of the volumes that will be added.
- Add minidisk statements to define minidisks (at virtual addresses 103 and 104 of size 6678 cylinders, in this example) to the appropriate Linux user ID definition in the `USER DIRECT` file.
- Create the `USER DISKMAP` file to verify the disk layout.
- Bring the changes online with the **`DIRECTXA`** command.
- Modify the Linux system to recognize the new minidisks.
- Shut down the system and log off the z/VM user ID. The LINUX02 user ID is used in the following example (this puts the logical volume on the LDAP server).
- Log on to the user ID and verify that the new minidisks are available.

## 13.1.2 Making the new minidisks available

Before you shut down, make a backup copy then modify the `/etc/modprobe.conf` file to include the new minidisks at virtual addresses 103 and 104:

```
# cd /etc
# cp modprobe.conf modprobe.conf.orig
# vi modprobe.conf
alias eth0 qeth
options dasd_mod dasd=100-104
```

Save the file. To effect the change, you must build a new initial RAMdisk with the following **`mkinitrd`** command, then run **`zipl`** to write the changes to the boot record:

```
# mkinitrd -v -f /boot/initrd-$(uname -r).img $(uname -r)
Creating initramfs
...
Adding module jbd
Adding module ext3
Adding module dasd_mod with options dasd=100-104
Adding module dasd_eckd_mod
Adding module dasd_fba_mod
# zipl
Using config file '/etc/zipl.conf'
Building bootmap in '/boot/'
Building menu 'rh-automatic-menu'
Adding #1: IPL section 'linux' (default)
Preparing boot device: dasda (0100).
Done.
# shutdown -h now
...
```

When your system comes back up, start an SSH session to it. Use the **lsdasd** command to verify that the new minidisks have been recognized:

```
# lsdasd
0.0.0100(ECKD) at ( 94: 0) is dasda : active at blocksize 4096, 600840 blocks, 2347 MB
0.0.0101(FBA ) at ( 94: 4) is dasdb : active at blocksize 512, 524288 blocks, 256 MB
0.0.0102(FBA ) at ( 94: 8) is dasdc : active at blocksize 512, 1048576 blocks, 512 MB
0.0.0103(ECKD) at ( 94:12) is dasdd : active at blocksize 4096, 1202040 blocks, 4695 MB
0.0.0104(ECKD) at ( 94:16) is dasde : active at blocksize 4096, 1202040 blocks, 4695 MB
```

## 13.1.3 Formatting and partitioning the minidisks

You can format the minidisks sequentially, but you could also use the following bash **for** loop to put two **dasdfmt** jobs in the background and format both in parallel, as shown ere:

```
# for i in d e
> do
>   dasdfmt -b 4096 -y -f /dev/dasd$i &
> done
[1] 2713
[2] 2714
```

When the jobs are finished, use the **fdasd** command with the **-a** flag to create a single partition from each minidisk:

```
# fdasd -a /dev/dasdd
auto-creating one partition for the whole disk...
writing volume label...
writing VTOC...
rereading partition table...
# fdasd -a /dev/dasde
auto-creating one partition for the whole disk...
writing volume label...
writing VTOC...
rereading partition table...
```

The minidisks are now ready for you to use in the logical volume. If you are creating a new logical volume, refer to 13.1.4, “Create the logical volume and file system” on page 182. If you are extending an existing logical volume, skip ahead to 13.2, “Extending an existing logical volume” on page 185.

## 13.1.4 Create the logical volume and file system

The overall steps involved in creating a logical volume are:

- Create physical volumes from the two DASD.
- Create a single volume group.
- Create a single logical volume.
- Make a file system from the logical volume.

Figure 13-1 shows a block diagram of the logical volume manager (LVM) reflecting this example.

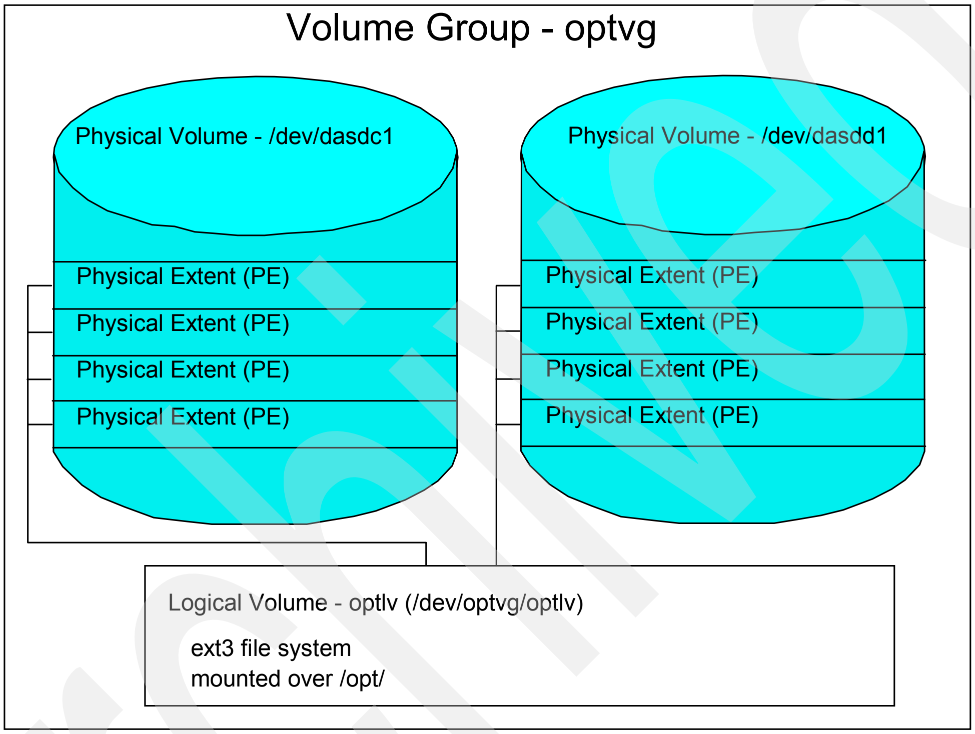


*Figure 13-1 Logical volume manager (LVM) block diagram*

### *Creating physical volumes from the two DASDs*

The **pvcreate** command initializes DASD for use by LVM. Initialize the two new DASD partitions. Verify with the **pvdisplay** command:

```
# pvcreate /dev/dasdd1 /dev/dasde1
  Physical volume "/dev/dasdd1" successfully created
  Physical volume "/dev/dasde1" successfully created

# pvdisplay
--- NEW Physical volume ---
  PV Name               /dev/dasdd1
  VG Name
  PV Size               4.59 GB
  Allocatable           NO
  PE Size (KByte)       0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               eksaUw-HvSX-S9m8-a5u0-RBOn-fMH7-y6fkRb
  --- NEW Physical volume ---
  PV Name               /dev/dasde1
  VG Name
  PV Size               4.59 GB
  Allocatable           NO
  PE Size (KByte)       0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               y6lMf2-PkCk-w46g-lSkl-3M9d-7vUZ-rtvaJX
```

### *Creating a single volume group*

The **vgcreate** command can be used to create a volume group named homevg from the two DASDs. Use the **vgdisplay** command to verify:

```
# vgcreate homevg /dev/dasdd1 /dev/dasde1

# vgdisplay
  --- Volume group ---
VG Name               homevg
  System ID
  Format                lvm2
  Metadata Areas        2
  Metadata Sequence No  1
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                0
  Open LV               0
  Max PV                0
  Cur PV                2
  Act PV                2
  VG Size               9.16 GB
  PE Size               4.00 MB
  Total PE              2346
  Alloc PE / Size       0 / 0
  Free  PE / Size       2346 / 9.16 GB
  VG UUID               IHAq2g-ME3w-b6LY-E3hA-raNk-bFOG-Poprma
```

In this example, there are 2346 free physical extents.

### *Creating a single logical volume*

The **lvcreate** command is used to create a logical volume. The **-l 2346** flag specifies to use all free extents, in this example. The **-n homelv** specifies the name of the new logical volume. The last argument **homevg** specifies the name of the volume group from which the logical volume will be created. Use the **lvdisplay** command to verify:

```
# lvcreate -l 2346 -n homelv homevg
  Logical volume "homelv" created
# lvdisplay
--- Logical volume ---
  LV Name                /dev/homevg/homelv
  VG Name                homevg
  LV UUID                roBwTM-sxkF-AdLn-sePN-KnzT-n36r-7lkIxk
  LV Write Access        read/write
  LV Status              available
  # open                 0
  LV Size                9.16 GB
  Current LE             2346
  Segments              2
  Allocation            inherit
  Read ahead sectors    0
  Block device          253:0
```

### *Making a file system from the logical volume*

Now you have a logical volume. Use the **mke2fs** command to create a file system out of it. The **-j** flag adds a journal so it will be of type ext3:

```
# mke2fs -j /dev/homevg/homelv
mke2fs 1.36 (05-Feb-2005)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
402400 inodes, 803840 blocks
40192 blocks (5.00%) reserved for the super user
...
```

The file system created from the logical volume is now ready to be mounted.

## 13.1.5 Updating the file system table

You could now mount the file system manually. However, if you add the mount to the file system table file, `/etc/fstab`, you can effectively test the change by using the **mount** command with only one argument. Add one line to the file:

```
# cd /etc
# vi fstab
LABEL=/                 /                       ext3    defaults        1 1
/dev/dasdb1             swap                    swap    defaults        0 0
/dev/dasdc1             swap                    swap    defaults        0 0
/dev/homevg/homelv      /home                   ext3    defaults        0 0
...
```

Mount the `/home/` file system with one argument. Use the **ls** command to verify that there is no data in the logical volume except the `lost+found/` directory. Use the **df -h** command to verify that it is mounted.

```
# mount /home
# ls /home
lost+found/
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1           2.3G  1.2G 1016M  53% /
tmpfs                 124M     0  124M   0% /dev/shm
/dev/mapper/homevg-homelv
                      9.1G  149M  8.5G   2% /home
```

### Moving data from existing file system

You may have noticed that there may be some data in the directory which will serve as the mount point. To complete the addition of the logical volume, we recommend that you move any data from the existing directory to the new logical volume. First unmount the logical volume and view the contents of the `/home/` directory:

```
# umount /home
# ls -F /home
ldapuser1/
```

In this example, there is only one subdirectory. You can move the contents from the root file system to the new logical volume a number of different ways. Following is one method:

```
# cd /home
# mkdir /tmp/home
# mv * /tmp/home
# cd ..
# mount /home
# cd /home
# mv /tmp/home/* .
# ls -F
ldapuser1/  lost+found/
```

Even though you tested mounting the file system using reading the `/etc/fstab` file, you may want to test a reboot to verify that the logical volume is successfully mounted.

## 13.2 Extending an existing logical volume

This section describes the process of adding a new minidisk to an existing LVM. This is useful when your logical volume has run out of space. In this example, the **`vgdisplay`** command shows that the existing volume group is full.

```
# vgdisplay
--- Volume group ---
  VG Name               homevg
...
  Total PE              2346
  Alloc PE / Size       2346 / 9.16 GB
  Free  PE / Size       0 / 0
  VG UUID               IHAq2g-ME3w-b6LY-E3hA-raNk-bFOG-Poprma
```

First, repeat the steps at the beginning of this section to add a new minidisk. In this example, a minidisk at virtual address 105 is added that is 3338 cylinders in size.

### Create a new RAMdisk to enable the new minidisk

Edit the /etc/modprobe.conf file and add the new minidisk at address 105 to the existing range:

```
# vi /etc/modprobe.conf
alias eth0 qeth
options dasd_mod dasd=100-105
```

Use the **`mkinitrd`** command to rebuild the initial RAMdisk, then use the **`zipl`** command to rewrite the boot sector on the root partition:

```
# mkinitrd -v -f /boot/initrd-$(uname -r).img $(uname -r)
...
# zipl
...
```

Reboot the system to test the changes:

```
# reboot
...
```

When your system comes back, verify that the new minidisk is available using the **`lsdasd`** command:

```
# lsdasd
...
```

```
0.0.0100(ECKD) at ( 94: 0) is dasda : active at blocksize 4096, 600840 blocks, 2347 MB
0.0.0101(FBA ) at ( 94: 4) is dasdb : active at blocksize 512, 524288 blocks, 256 MB
0.0.0102(FBA ) at ( 94: 8) is dasdc : active at blocksize 512, 1048576 blocks, 512 MB
0.0.0103(ECKD) at ( 94:12) is dasdd : active at blocksize 4096, 600840 blocks, 2347 MB
0.0.0104(ECKD) at ( 94:16) is dasde : active at blocksize 4096, 600840 blocks, 2347 MB
0.0.0105(ECKD) at ( 94:20) is dasdf : n/f
```

Format the minidisk using the **dasdfmt** command and make a single partition using the **fdasd -a** command as you did before:

```
# dasdfmt -b 4096 -y -f /dev/dasdf
Finished formatting the device.
Rereading the partition table... ok
# fdasd -a /dev/dasdf
auto-creating one partition for the whole disk...
writing volume label...
writing VTOC...
rereading partition table...
```

### Creating a physical volume

Use the **pvcreate** command to create a physical volume from the minidisk:

```
# pvcreate /dev/dasdf1
  Physical volume "/dev/dasdf1" successfully created
```

### Extending the volume group

Use the **vgextend** command to extend the volume group into the new physical volume. Then, use **vgdisplay** to verify that the volume group has free space:

```
# vgextend homevg /dev/dasdf1
  Volume group "homevg" successfully extended
# vgdisplay
--- Volume group ---
  VG Name               homevg
  ...
  Total PE              2932
  Alloc PE / Size       2346 / 9.16 GB
  Free  PE / Size       586 / 2.29 GB
  VG UUID               IHAq2g-ME3w-b6LY-E3hA-raNk-bFOG-Poprma
```

Note there are 586 new free physical extents (PEs) in this example.

### Extend the logical volume and the file system

Now that you have free space in the volume group, you can increase the size of the existing logical volume with the **lvextend** command. The **-l** option specifies the number extents to add. Finally, use the **resize2fs** command, which increases the size of the file system while it is mounted. You can use the **df** command to show the file system size before and after you extend it, as the following example shows:

```
# df -h | grep home
/dev/mapper/homevg-homelv
                      9.1G  149M  8.5G   2% /home
# lvextend -l +586 /dev/homevg/homelv
  Extending logical volume homelv to 11.45 GB
  Logical volume homelv successfully resized
# resize2fs -p /dev/homevg/homelv
resize2fs 1.39 (29-May-2006)
Filesystem at /dev/homevg/homelv is mounted on /home; on-line resizing required
Performing an on-line resize of /dev/homevg/homelv to 3002368 (4k) blocks.
The filesystem on /dev/homevg/homelv is now 3002368 blocks long.
```

```
# df -h | grep home
/dev/mapper/homevg-homelv
                       12G  152M   11G   2% /home
```

You have now added the new minidisk as free space to the existing logical volume. Test your changes by rebooting your system:

```
# reboot
...
```

## 13.3 Centralizing home directories for LDAP users

If you have completed 12.2, “Creating a virtual LDAP server” on page 164, you have a working LDAP authentication server that provides centralized login authentication.

If you have completed 13.1, “Adding a logical volume” on page 180, you have a larger file system mounted over `/home/` for storing user data.

Now you can bring these together with the automount service to have a centralized `/home` file system. In this fashion, users can log in to all virtual servers using the same credentials and be able to access the same data. Further, you can modify the golden image so that all newly cloned virtual servers will inherit this setup. Figure 13-2 is a block diagram

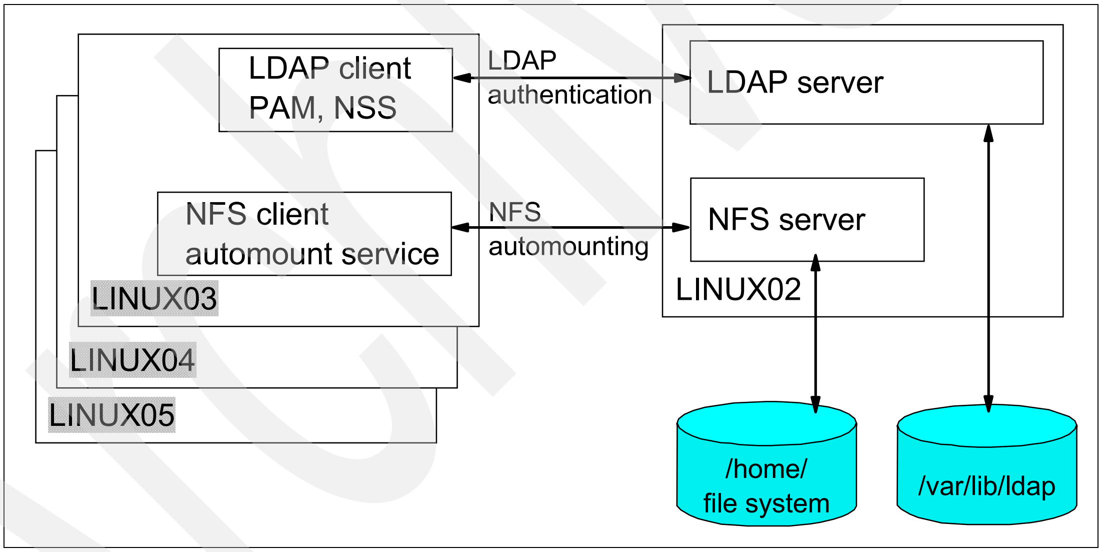


*Figure 13-2 Block diagram of centralized LDAP authentication and automounted /home file system*

### 13.3.1 Configuring the NFS server

Configure `LINUX02` to export `/home` over NFS.

Start an SSH session to the virtual server running on `LINUX02`. Add one line to the `/etc/exports` file:

```
# vi /etc/exports
/home           *(rw,sync)
```

Set the NFS service to start at boot time using the **`chkconfig`** command, and for this session using the **`service`** command:

```
# chkconfig nfs on
# service nfs start
Starting NFS services:  [  OK  ]
Starting NFS quotas: [  OK  ]
Starting NFS daemon: [  OK  ]
Starting NFS mountd: [  OK  ]
```

Test mounting the newly exported file system locally:

```
# mount localhost:/home /mnt
# ls /mnt
ldapuser1  lost+found
```

You now have `/home/` available for NFS mounting.

### 13.3.2 Configuring the golden image for LDAP and automount

In this section you will shut down the controller and boot the golden image. You will then configure it for LDAP authentication and automount of `/home/` to the virtual server running on `LINUX02`.

Start a 3270 connect to the controller, shut it down, and IPL the golden image:

```
Red Hat Enterprise Linux Server release 4.92 (Tikanga)
Kernel 2.6.18-1.2839.el5 on an s390x

virtc530 login: root
Password:

Last login: Fri Dec 15 09:49:10 from 9.56.60.150
#shutdown -h now
...
==> ipl 100
...
==> #cp disc
```

#### Configuring LDAP

You can modify the authentication settings using the command **authconfig-tui** (you can also use **authconfig** from a graphical environment such as VNC). Use the **Tab** key to move between fields, the **Spacebar** to change selections and the **Enter** key to select:

```
# authconfig-tui

   +----------------¦ Authentication Configuration +-----------------+
   ¦                                                                 ¦
   ¦  User Information        Authentication                         ¦
   ¦  [*] Cache Information   [*] Use MD5 Passwords                  ¦
   ¦  [ ] Use Hesiod          [*] Use Shadow Passwords               ¦
   ¦  [*] Use LDAP            [*] Use LDAP Authentication            ¦
   ¦  [ ] Use NIS             [ ] Use Kerberos                       ¦
   ¦  [ ] Use Winbind         [ ] Use SMB Authentication             ¦
   ¦                          [ ] Use Winbind Authentication         ¦
   ¦                          [*] Local authorization is sufficient  ¦
   ¦                                                                 ¦
   ¦            +--------+                      +------+             ¦
   ¦            ¦ Cancel ¦                      ¦ Next ¦             ¦
   ¦            +--------+                      +------+             ¦
   ¦                                                                 ¦
   ¦                                                                 ¦
   ¦                                                                 ¦
   +-----------------------------------------------------------------+
```

Under User Information, select **Cache Information** and **Use LDAP**. Under Authentication, select **Use LDAP Authentication** and **Local authorization is sufficient**. Press **Enter** when you have tabbed to the **Next** button.

```
+-----------------¦ LDAP Settings +-----------------+
¦                                                   ¦
¦          [ ] Use TLS                              ¦
¦  Server: ldap://<9.12.5.32>/____________________  ¦
¦ Base DN: <dc=itso,dc=ibm,dc=com>________________  ¦
¦                                                   ¦
¦         +------+                +----+            ¦
¦         ¦ Back ¦                ¦ Ok ¦            ¦
¦         +------+                +----+            ¦
¦                                                   ¦
¦                                                   ¦
+---------------------------------------------------+
```

Set the Server value to point to the LDAP server. In this example, it is `ldap://9.12.5.32/`. Set the `Base DN` to your suffix value. In this example, it is `dc=itso,dc=ibm,dc=com`. Press **OK**.

Your LDAP client should now be pointing to the LDAP server. Test it using the **`id ldapuser1`** command:

```
# id ldapuser1
uid=500(ldapuser1) gid=500(ldapuser1) groups=500(ldapuser1)
context=root:system_r:unconfined_t:s0-s0:c0.c1023
```

Authentication should also allow LDAP. Test this by starting another SSH session to the LDAP server and logging in as ldapuser1:

```
login as: ldapuser1
ldapuser1@9.12.5.32's password:
```

When you get an SSH session as `ldapuser1`, this shows that authentication is working.

## Configuring automount

Now SSH to the golden image to configure the automounter. Add an entry to the bottom of `/etc/auto.master` for `/home`:

```
# vi /etc/auto.master
/nfs    /etc/auto.controller
/home   /etc/auto.home
```

Create the file `/etc/auto.home` and add one line to it:

```
# vi /etc/auto.home
*       <9.12.5.32>:/home/&
```

The asterisk (*) is used as a wildcard character to represent the local subdirectory. The ampersand (&) is used as the remote system name or the remote subdirectory. Whatever is in the local directory name field replaces the ampersand character.

Finally, restart **`autofs`** and test logging in as `ldapuser1` with **`su`**. You will see that /home/ldapuser1 is now mounted from the LDAP server:

```
# service autofs restart
Stopping automount: [  OK  ]
Starting automount: [  OK  ]
# su - ldapuser1
$ pwd
/home/ldapuser1
$ mount | grep ldapuser1
```

```
9.12.5.32:/home/ldapuser1 on /home/ldapuser1 type nfs (rw,addr=9.12.5.32)
```

You have now configured the automounter to mount a remote LDAP user's home directory at login.

Shut down the golden image, and IPL the controller (`200`). Clone a new virtual server using the **`clone.sh`** script. Start an SSH session to the new virtual server and LDAP authentication, and the automounting of `/home/` using NFS should work.

## 13.4 Rescuing a Linux system

This section describes how to boot your Linux server into different modes for troubleshooting purposes. It covers booting Linux into single user mode, and also entering a rescue environment when you require more advanced troubleshooting.

### 13.4.1 Entering single user mode

Single user mode is helpful when you need to recover the root password, or if you are having problems while booting Linux into the default runlevel. To enter single user mode, first IPL your Linux server from the 3270 console. You will see a message similar to the one shown here:

```
zIPL v1.3.2 interactive boot menu
0. default (linux)
1. linux
Note: VM users please use '#cp vi vmsg <input>'
Please choose (default will boot in 15 seconds):
```

Use the **`#cp vi vmsg`** command to boot the desired menu option (zero (0), in this example), followed by the number 1 for single user mode:

```
==> #cp vi vmsg 0 1
```

After some initial kernel boot messages, you should see:

```
INIT: Entering runlevel: 1
Entering non-interactive startup
Telling INIT to go to single user mode.
INIT: Going single user
INIT: Sending processes the TERM signal
INIT: Sending processes the KILL signal
sh-3.00#
```

In single user mode, you are logged in as the root user. You can use the **`passwd`** command to set the root password. All of the file systems in `/etc/fstab` are mounted, but networking has not been started. To exit single user mode, you can type **`reboot`**, or enter **`init 3`** to continue booting normally.

### 13.4.2 Entering a rescue environment

If you encounter errors mounting the root file system, or have other problems that prevent you from entering single user mode, you can enter a *rescue* environment. This environment loads a Linux image in memory, and does not attempt to mount the root file system.

To enter a rescue environment, initiate an interactive Linux installation. Perform the following steps to enter a rescue environment on the `LINUX03` user ID:

1. **`Logon`** to `LNXMAINT`. Copy the `RHEL5 EXEC` file to a new file named `RESCUE EXEC`, and copy the user's `PARM-RH5` file to a new file (`LINUX03 RESCUE`, in this example):

   ```
   ==> copy rhel52 exec d rescue = =
   ==> copy linux03 parm-rh5 d = rescue =
   ```

2. Next, edit `RESCUE EXEC` to point to the new `LINUX03 RESCUE` file:

   ```
   ==> xedit RESCUE EXEC
   ...
   00005 'PUN RHEL52       KERNEL * (NOH'
   00006 'PUN' userid() '  RESCUE * (NOH'
   00007 'PUN RHEL52       INITRD * (NOH'
   ...
   ==> file
   ```

3. Edit the `LINUX03 RESCUE` file, replacing any kickstart or VNC lines with the **`rescue`** command line option:

   ```
   ==> xedit LINUX03 RESCUE
   ramdisk_size=40000 root=/dev/ram0 ro ip=off
   CMSDASD=191 CMSCONFFILE= LINUX03.CONF-RH5
   rescue
   ```

4. Now **`logoff`** from LNXMAINT and logon to LINUX03. Respond `no` to the IPL from DASD question, and run `RESCUE EXEC`.

   ```
   ==> rescue
   Linux version 2.6.18-92.el5 (brewbuilder@spark.z900.redhat.com) (gcc version 4.1
   .2 20071124 (Red Hat 4.1.2-41)) #1 SMP Tue Apr 29 13:16:58 EDT 2008
   We are running under VM (64 bit mode)
   Detected 4 CPU's
   Boot cpu address  0
   Built 1 zonelists.  Total pages: 524288
   Kernel command line: ramdisk_size=40000 root=/dev/ram0 ro ip=off
                        CMSDASD=191 CMSCONFFILE=Z12.conf
                        rescue
   ...
   ```

   The install process directs you to telnet or SSH to the IP address of your Linux server to begin the first stage of the installation.

5. Use SSH to connect and log in as root. After choosing the language, the rescue environment will prompt you for the location of the rescue image, which is located in the install tree on the controller.

6. Choose **NFS image**, then enter the IP address or host name of the controller and the path `/nfs/rhel5.2`

7. The rescue image will search for your Linux installation and will prompt you to mount the partitions it finds. When prompted, select **Continue** to mount the partitions. After this, you will be presented with a shell prompt. Because you chose to mount your root partition, it will be automatically mounted to the `/mnt/sysimage` directory.

   You can use the **`chroot`** command to run an interactive bash shell from the `/mnt/sysimage` directory. To do this, run the following command:

   ```
   sh-3.2# chroot /mnt/sysimage /bin/bash
   ```

   The **`/bin/bash`** argument instructs **`chroot`** to run the bash shell after changing the root directory to `/mnt/sysimage`. From the **`chroot`** shell, the system appears almost exactly as it would if it were running (the only difference is that, because **`udev`** is not running, there may be no device nodes under /dev). If you need to rewrite the master boot record (MBR), run **zipl** from the chroot shell.

To exit the shell, type: `exit`.

If the rescue image cannot find your partition, or if you chose **Skip** when prompted to search for the existing file systems, you can mount the file systems manually.

First, bring the LVM logical volumes online with the following two commands:

```
sh-3.2# lvm vgscan
Reading all physical volumes.  This may take a while...
  Found volume group "system_vg" using metadata type lvm2
sh-3.2# lvm vgchange -a y
  5 logical volume(s) in volume group "system_vg" now active
```

**Note:** You can use the **lvm <command>** to troubleshoot any LVM problems in rescue mode. For a list of possible commands, run:

```
sh-3.2:# lvm help
```

When the LVM is online, create a mount point, and then mount the file systems.

```
sh-3.2# mkdir /mnt/sysimage
sh-3.2# mount /dev/system_vg/root_lv /mnt/sysimage/
sh-3.2# mount /dev/system_vg/tmp_lv /mnt/sysimage/tmp
sh-3.2# mount /dev/system_vg/opt_lv /mnt/sysimage/opt
sh-3.2# mount /dev/system_vg/var_lv /mnt/sysimage/var
sh-3.2# mount /dev/system_vg/usr_lv /mnt/sysimage/usr
```

Lastly, mount the /boot file system located on dasda1:

```
sh-3.2# mount /dev/dasda1 /mnt/sysimage/boot
```

8. When finished, type: `exit` to leave the chroot shell, then type: exit again to leave rescue mode.

# 14

# Monitoring z/VM and Linux

This chapter briefly describes how to monitor z/VM and Linux. For more detailed information about z/VM performance and monitoring, see Chapter 11, “Monitoring performance and capacity”, in the *Getting Started With Linux*, SC24-6096, which is available on the Web at:

http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/Shelves/hcsh2a70

There are a number of z/VM monitoring tools such as the CA VM:Monitor, the IBM z/VM Performance Toolkit, and the IBM Tivoli OMEGAMON® XE for z/VM and Linux. The IBM z/VM Performance Toolkit is briefly described in this section.

For more information about the CA VM:Monitor, see:

http://www.ca.com/

This chapter describes the following topics:

- Using INDICATE and other commands
- The z/VM Performance Toolkit
- Monitoring Linux
- Viewing Linux data in the Performance Toolkit

## 14.1 Using INDICATE and other commands

z/VM has many commands to monitor the state of the system. **`CP INDICATE`** is the most commonly used, and there are other commands that are addressed.

### 14.1.1 Using the INDICATE command

z/VM has basic commands such as **`INDICATE`**. This command can be useful if there are no other tools with more extended function present on the system, such as when undertaking a completely new system installation. (However, command level tools should not be relied upon for formal, long-term reporting on production systems.)

There are many **`INDICATE`** parameters that can be included as command line options. Use the command **`HELP INDICATE`** for a basic understanding, and then press **F11** for help on each parameter.

#### *INIDICATE LOAD*

If no parameter is specified, **`INDICATE LOAD`** is the default option. There are two “flavors” of this, depending on whether the issuing user ID has privilege class G or privilege class E. Class G users can use **`INDICATE`** to display recent contention for system resources, as well as to display environment characteristics and measurements of resources used by their virtual machine.

The output from user ID with class E privilege (for example, `MAINT`, `OPERATOR`) is shown here:

```
==> ind load
1  AVGPROC-038% 03
2  XSTORE-000021/SEC MIGRATE-0001/SEC
3  MDC READS-000068/SEC WRITES-000001/SEC HIT RATIO-099%
4  PAGING-0031/SEC STEAL-000%
5  Q0-00006(00000)                          DORMANT-00357
6  Q1-00001(00000)            E1-00000(00000)
7  Q2-00001(00000) EXPAN-002 E2-00000(00000)
8  Q3-00034(00000) EXPAN-002 E3-00000(00000)
9
10 PROC 0000-038%          PROC 0001-038%
11 PROC 0002-038%
12
13 LIMITED-00000
```

The numbers in bold font refer to the descriptions that follow:

The **`INDICATE LOAD`** command gives a snapshot of current system performance. Except for the counts of virtual machines in various queues and the limited list, the values you see here are a smoothed average over the past 4 minutes. Areas where z/VM performance analysts tend to focus are the following:

**1** AVGPROC gives the overall processor utilization (38%, in this example). The number following it is the number of online processors (3, in this example). The individual processor utilization is shown on lines **10** and **11**. Examine these to see if they are somewhat balanced. There are cases where an imbalance is acceptable. This would include very low utilization scenarios, or cases where there are not enough users ready to run virtual processors to keep the physical processors busy. One of the processors will be a Master, all of the others Alternate, and some imbalance may result from performing these functions.

2 Paging to expanded storage is displayed here. Most z/VM systems on z9 class machines can sustain several thousands of this type of paging operations a second without any problems. The MIGRATE rate is the number of pages per second being moved from expanded storage out to paging space on DASD. A healthy system will have a MIGRATE rate significantly lower than the XSTORE rate, probably being measures in hundreds rather than thousands. The higher values seen tend to build up over time, and are sustained over periods of intense system activity; however, there are times when the MIGRATE value may spike for brief periods of time.

3 Minidisk cache (MDC) statistics are displayed here. The effectiveness of MDC can be judged by the combination of the `READS` rate and the `HIT RATIO`. If both are high, then a large number of physical I/Os are avoided due to the MDC feature. For a system that has an appreciably high I/O rate, composed of reads plus writes, and a high proportion of reads, and a good hit ratio for those reads (tending to 90% or greater), the real, physical I/O avoidance can be very high (as high as 50% in some cases has been observed). Conversely, however, a high `HIT RATIO` with a low value for the READS rate should not be taken as desirable (a 100% hit ratio, when doing only one I/O per second, is effectively meaningless).

4 More storage (memory) management is displayed here. The `PAGING` rate is important. Higher values will often impact performance. This can be at least partially offset by increasing the number of page volumes, but a more thorough examination of this problem is advisable whenever it arises.The `STEAL` percentage is often misleading. This is basically the percentage of pages taken from guests that z/VM believes are non-dormant. Because some guests have periodic timers going off, they appear to be active to z/VM even when relatively idle. Pages taken from these guests are still considered to be stolen. So there are scenarios where a system only has a user set comprising active guests, in which case all pages taken would be considered stolen. Bearing this in mind, if a high `STEAL` value is observed, the paging rate needs to be checked. If the paging rate is relatively low, then the `STEAL` value is not important.

On lines 5 through 8, you also see a series of counters that represent the users in various queues. The z/VM scheduler classifies work into three different classes (1 through 3), and a special additional class labelled zero. So the Column of Qx values and Ex represent the virtual machines in the dispatch list and the eligible list. The most important value here to validate is that there are no virtual machines in the Eligible list: E1, E2, E3; this implies z/VM has stopped dispatching some virtual machines to avoid over-committing resources. Such a system would require further investigation, possibly leading to some tuning work, or even the addition of some hardware in extreme cases. You can ignore the values in parenthesis.

### *INDICATE QUEUES EXP*

Another useful command to help you determine the state of the system is **`INDICATE QUEUES EXP`**. Following is an example:

```
==> ind q exp
DATAMGT1      Q3 AP  00000537/00000537 .... -2.025 A02
BITNER        Q1 R00 00000785/00000796 .I.. -1.782 A00
EDLLNX4       Q3 PS  00007635/00007635 .... -1.121 A00
TCPIP         Q0 R01 00004016/00003336 .I.. -.9324 A01
APCTEST1      Q2 IO  00003556/00003512 .I.. -.7847 A01
EDLWRK20      Q3 AP  00001495/00001462 .... -.6996 A01
EDL           Q3 IO  00000918/00000902 .... -.2409 A01
EDLWRK11      Q3 AP  00002323/00002299 .... -.0183 A00
EDLWRK18      Q3 IO  00001052/00000388 .... -.0047 A00
EDLWRK4       Q3 AP  00004792/00002295 ....  .0055 A01
EDLWRK8       Q3 AP  00004804/00004797 ....  .0089 A02
EDLWRK16      Q3 AP  00002378/00002378 ....  .0170 A02
EDLWRK2       Q3 AP  00005544/00002956 ....  .0360 A00
EDLWRK12      Q3 AP  00004963/00002348 ....  .0677 A01
```

```
EDLWRK6      Q3 IO  00000750/00000302 ....  .0969 A02
EDLWRK3      Q3 AP  00005098/00005096 ....  .0999 A02
EDLWRK17     Q3 AP  00004786/00004766 ....  .1061 A01
EDLWRK9      Q3 AP  00002372/00002334 ....  .1107 A02
EDLWRK5      Q3 IO  00002376/00002376 ....  .1205 A01
EDLWRK14     Q3 AP  00002426/00002323 ....  .1238 A02
EDLLIB19     Q3 IO  00001226/00001100 ....  .1309 A02
EDLWRK19     Q3 AP  00002322/00002298 ....  .1705 A00
EDLWRK15     Q3 AP  00002839/00002781 ....  .2205 A02
EDLWRK1      Q3 AP  00002969/00002935 ....  .2491 A02
```

This is another class E command, and it displays the virtual processors associated with a given user ID (a single virtual machine may have multiple virtual processors), along with what queue (dispatch list, eligible list, limit list) they are in and what state they are in. This is a snapshot in time.

Check this output to make sure there are no virtual machines in the eligible list. Normal virtual processors in the dispatch list will be $Q_x$ (x=1,2,3). Eligible list would be marked as $E_x$.

The third column in the example displays the state of the virtual processor. This information can give you an idea of how the virtual processors might be constrained. Virtual processors that are actually running in the snapshot period are marked with RNN, where NN is the processor number they are on. An R without a number means the virtual processor is ready to run, but there is no processor available. (Note that the virtual machine that issues the **INDICATE** command will always be one of the running machines.) Other states are documented in the help information for the **IND Q EXP** command.

You can ignore the remaining columns unless detailed analysis is required, or IBM support requests it. Also, always remember that is just a snapshot in time and repeating this command over time can often provide a more accurate picture of your z/VM system; a single snapshot cannot be regarded as indicative.

## 14.1.2 Using other basic commands

This section briefly mentions other basic commands that are especially useful. All examples are shown from the MAINT user ID. The results will be different for users with fewer privileges.

### *Getting help*

To get help on the system, use the **HELP** command. Here are some useful help commands:

```
==> help             // for basic help
==> help menus       // for menu of all z/VM help menus
==> help cp menu     // for a menu of all CP commands
==> help cpquery     // for a menu of all CP QUERY command
==> help cpset       // for a menu of all CP SET commands
```

### *Determining who is logged on*

To see who is logged on to the system, use the **QUERY NAMES** command. For example:

```
==> q n
LINUX06  - DSC , LINUX04  - DSC , LINUX03  - DSC , LINUX07  - DSC
LINUX01  - DSC , SLES9    - DSC , FTPSERVE - DSC , DTCVSW2  - DSC
DTCVSW1  - DSC , TCPIP    - DSC , OPERSYMP - DSC , DISKACNT - DSC
EREP     - DSC , OPERATOR - DSC , MAINT    -L0005
VSM     - TCPIP
```

### *Determining storage or memory*

To see how much central and expanded storage (memory) are installed and allocated to a system, use the **QUERY STORAGE** and **QUERY XSTOR** commands. For example:

```
==> q stor
STORAGE = 4G CONFIGURED = 4G INC = 256M STANDBY = 0  RESERVED = 0
==> q xstor
XSTORE= 2048M online= 2048M
XSTORE= 2048M userid= SYSTEM usage= 0% retained= 0M pending= 0M
XSTORE MDC min=0M, max=0M, usage=0%
XSTORE= 2048M userid=  (none)  max. attach= 2048M
```

### *Determining processors or CPUs*

To see how many processors (CPs, IFLs, CPUs) you have allocated at system level, use the **QUERY PROCESSORS** command. For example:

```
==> q proc
PROCESSOR 00 MASTER CP
PROCESSOR 01 ALTERNATE CP
PROCESSOR 02 ALTERNATE CP
PROCESSOR 03 ALTERNATE CP
PROCESSOR 04 STANDBY CP
PROCESSOR 05 STANDBY CP
PROCESSOR 06 STANDBY CP
PROCESSOR 07 STANDBY CP
```

### *Determining software level*

To determine what level of CP your system is at, use the **QUERY CPLEVEL** command. For example:

```
==> q cplevel
z/VM Version 5 Release 4.0, service level 0801 (64-bit)
Generated at 04/08/08 18:17:39 EDT
IPL at 05/18/08 10:17:32 EDT
```

### *Determining system cylinder allocation*

The **QUERY ALLOC MAP** command shows you the system allocation of spool, paging and directory space. For example:

```
==> q alloc map
EXTENT     EXTENT                         % ALLOCATION
VOLID  RDEV      START        END  TOTAL IN USE   HIGH USED TYPE
------ ---- ---------- ---------- ------ ------ ------ ---- -------------
MVD850 D850          1         20     20      1      2   5% DRCT ACTIVE
MVD851 D851          1       3338 600840  58332  63360   9% SPOOL
MVD852 D852          1       3338 600840      0      0   0% PAGE
MPD855 D855          0       3338 601020      0      0   0% PAGE
MPD950 D950          0       3338 601020      0      0   0% PAGE
MPD951 D951          0       3338 601020      0      0   0% PAGE
MPDA50 DA50          0       3338 601020      0      0   0% PAGE
MPDB50 DB50          0       3338 601020     13     45   1% PAGE
```

### *Determining DASD, OSA and virtual resources*

The **QUERY DASD** and **QUERY DASD FREE** commands will show you what DASD is assigned to the system and what DASD is free to be assigned. Similarly, the **QUERY OSA** and **QUERY OSA FREE** commands report on OSA resources. Finally, the **QUERY VIRTUAL ALL** command can be useful.

The following list gives the short form of these commands, without associated output:

```
==> q da
==> q da free
==> q osa
==> q osa free
==> q v all
```

## 14.2 z/VM Performance Toolkit

To use the z/VM Performance Toolkit, the product must be licensed. Only configure the product if you have licensed it.

For more detailed information about the toolkit, refer to the following documentation:

- *z/VM Performance Toolkit Guide*, SC24-6156, and *z/VM Performance Toolkit Reference*, SC24-6157, which are available on the Web starting at the z/VM 5.4 bookshelf:

  http://www-03.ibm.com/systems/z/os/zos/bkserv/zvmpdf/#zvm54

  Search for **Toolkit** on that page.
- *The Program Directory for Performance Toolkit for VM*, GI10-0785-00

  http://www.vm.ibm.com/progdir/5vmptk30.pdf
- IBM Redbooks publication *Linux on IBM zSeries and S/390: Performance Toolkit for VM*, SG24-6059

  http://www.redbooks.ibm.com/abstracts/sg246059.html

The following sections provide a brief explanation of how to set up and use the IBM Performance Toolkit.

### 14.2.1 Configuring the z/VM Performance Toolkit

The Performance Toolkit is installed with z/VM. Configuration is described in the Program Directory. Following is a summary of how to turn it on. As mentioned, only configure the product if you have licensed it.

1. Query which priced products are enabled using the `QUERY PRODUCT` command:

   ```
   ==> q product
   Product  State    Description
   5VMDIR40 Disabled 00/00/00.00:00:00.$BASEDDR DIRECTORY MAINTENANCE FL 540
   5VMPTK40 Disabled 00/00/00.00:00:00.$BASEDDR PERFORMANCE TOOLKIT FOR VM
   5VMRAC40 Disabled 00/00/00.00:00:00.$BASEDDR RACF for VM
   5VMRSC40 Disabled 00/00/00.00:00:00.$BASEDDR RSCS Networking Version 5 Release 4
    Modification 0
   ```

2. To enable The z/VM Performance Toolkit, `logon` to `MAINT` and enter the following command:

   ```
   ==> service perftk enable
   VMFSRV2760I SERVICE processing started
   ...
   VMFSUT2760I VMFSUFTB processing started
   VMFSUT2760I VMFSUFTB processing completed successfully
   VMFSRV2760I SERVICE processing completed successfully
   ```

   You should see a few windows of messages scroll by and finally the success messages shown above. This will enable the Performance Toolkit for the current z/VM session.

3. At IPL time, the SYSTEM CONFIG file is modified by having a line appended to the end. Verify this with the following commands:

```
==> acc cf1 f
==> x system config f
====> bot
====> -2
====> pre off
...
PRODUCT PRODID 5VMPTK40 STATE ENABLED DESCRIPTION '05/22/08.10:08:55.MAINT    PE
RFKIT Minidisk Install and Service'
```

The Performance Toolkit is now enabled.

## 14.2.2 Configuring Web Browser support

After the product is enabled, the TCPIP profile must be modified to add browser capabilities for the Performance Toolkit. The following example describes how to set the port to 80, which is the default for a Web browser.

1. Logon to TCPMAINT. Edit the <vmlinuxa> TCPIP D file (assuming you modified this file name earlier; the default name is PROFILE TCPIP) and search for the string reserve ports. This is where z/VM TCP/IP ports are reserved.

```
==> x <vmlinuxa> tcpip d
====> /port
```

2. Add the following lines under the PORT entries:

```
...
PORT
  20   TCP FTPSERVE  NOAUTOLOG ; FTP Server
  21   TCP FTPSERVE            ; FTP Server
  23   TCP INTCLIEN            ; TELNET Server
; 25   TCP SMTP                ; SMTP Server
; 53   TCP NAMESRV             ; Domain Name Server
; 53   UDP NAMESRV             ; Domain Name Server
; 67   UDP BOOTPD              ; BootP Server
; 67   UDP DHCPD               ; DHCP Server
; 69   UDP TFTPD               ; TFTPD (Trivial FTP) Server
  80   TCP PERFSVM             ; Performance Toolkit
; 111  TCP PORTMAP             ; Portmap Server
...
```

Save your changes. The TCPIP user ID needs to be recycled in order for your changes to take effect. You can issue **FORCE** and **XAUTOLOG TCPIP** from a console. Alternatively, if you are in a position to reIPL the system, you can do that (shutdown reipl iplparms cons=sysc).

3. When the system comes back, logon to TCPMAINT and verify success by issuing the **NETSTAT** command. Verify that the service PERFSVM is in a Listen state.

```
==> netstat
VM TCP/IP Netstat Level 540       TCP/IP Server Name: TCPIP

Active IPv4 Transmission Blocks:

User Id  Conn    Local Socket           Foreign Socket         State
---- --  ----    ----- ------           ------- ------         -----
FTPSERVE 1001    *..FTP-C               *..*                   Listen
INTCLIEN 1009    *..TELNET              *..*                   Listen
PERFSVM  1011    *..80                  *..*                   Listen
```

```
Active IPv6 Transmission Blocks: None
```

In this case, the `PERFSVM` service was shown to be listening on port 80.

### 14.2.3 Configuring PERFSVM

The `PERFSVM` user ID is the Performance Toolkit service machine.

1. Logon to `PERFSVM`. If you successfully enabled the product, you should be put into a Performance Toolkit session and see the following text at the top of the window:

   ```
   FCX001                Performance Toolkit for VM
   FCXBAS500I Performance Toolkit for VM FL530 BASE
   Monitor event started -- recording is activated
   Monitor sample started -- recording is activated
   ```

(You may also receive the message `FCXPMN446E Incomplete monitor data: SAMPLE CONFIG size too small`. For an explanation of how to resolve this issue, refer to 14.5, “Common Performance Toolkit issue” on page 209.)

2. Press **PF12** twice to get to a CMS prompt.
3. Copy the `PROFILE XEDIT` from the `MAINT 191` disk so XEDIT sessions will have a common interface among user IDs.

   a. Use the **`VMLINK`** command to both link the disk read-only and access it as the highest available file mode. The default read password is **read**; however, if you changed your passwords as described in 4.9.3, “Changing passwords in USER DIRECT” on page 61, then the password will be `lnx4vm`.

   ```
   ==> vmlink maint 191
   ENTER READ PASSWORD:
   lnx4vm
   DMSVML2060I MAINT 191 linked as 0120 file mode Z
   ```

   b. Copy the `PROFILE XEDIT` to the A disk:

   ```
   ==> copy profile xedit z = = a
   ```

4. Copy the default configuration files, which are on the `PERFSVM` D disk, to your A disk:

   ```
   ==> copy * * d = = a
   ```

5. The main configuration file is `FCONX $PROFILE`. Edit that file and search for the string VMCF. This should take you to line 173, where the next four lines are comments starting with an asterisk (*). Make the following changes:

   – Uncomment the second line and the fourth line by changing `*C` to **`FC`**.
   – Change port 81 to 80 on the fourth line.
   – Add the text `FC MONCOLL LINUXUSR ON TCPIP TCPIP` after the fourth line. This tells the Performance Toolkit to collect Linux performance data.

   The modified lines should look as follows. Save your changes with the **`FILE`** subcommand:

   ```
   ==> x fconx $profile a
   ====> /vmcf
   *    Following command activates VMCF data retrieval interface
   FC MONCOLL VMCF ON
   *    Following command activates Internet interface
   FC MONCOLL WEBSERV ON TCPIP TCPIP 80
   FC MONCOLL LINUXUSR ON TCPIP TCPIP
   *    Following command activates Internet interface with SSL
   *C MONCOLL WEBSERV ON SSL TCPIP TCPIP 81 IDTEST RACF
   ...
   ```

```
====> file
```

6. Create a remote data retrieval authorization file with your z/VM system identifier; replace `<vmlinuxa>` with your z/VM system name:

   ```
   ==> x fconrmt authoriz
   ====> a 2
   <vmlinuxa>  PERFSVM  S&FSERV
   <vmlinuxa>  MAINT    DATA CMD EXCPMSG
   ```

7. Create a system identification file; replace `<vmlinuxa>` with your z/VM system name:

   ```
   ==> x fconrmt systems
   ====> a
   <vmlinuxa>   PERFSVM  z/VM5.4 N   FCXRES00
   ```

8. Create a Linux system definition file. Add the TCP/IP addresses of your Linux system(s). The following example shows adding five Linux virtual servers:

   Use port 8803 for Linux performance data:

   ```
   ==> x fconx linuxusr a
   LINUX00 <9.12.5.30>:8803
   LINUX01 <9.12.5.31>:8803
   LINUX02 <9.12.5.32>:8803
   LINUX03 <9.12.5.33>:8803
   LINUX04 <9.12.5.34>:8803
   ```

9. Edit the `PROFILE EXEC` file and uncomment the five `MONITOR SAMPLE` statements and the two `MONITOR EVENT` statements. Also, the size of the of the `SAMPLE CONFIG` and `EVENT CONFIG` may need to be increased:

   ```
   ==> x profile exec a
   ```

   Before:

   ```
   ...
   /*** Once you have PERFKIT enabled and running uncomment the      ***/
   /*** following comments                                           ***/
   /* 'CP MONITOR SAMPLE ENABLE PROCESSOR'    */
   /* 'CP MONITOR SAMPLE ENABLE STORAGE'      */
   /* 'CP MONITOR SAMPLE ENABLE USER ALL'     */
   /* 'CP MONITOR SAMPLE ENABLE I/O ALL'      */
   /* 'CP MONITOR SAMPLE ENABLE NETWORK'      */
   /* 'CP MONITOR SAMPLE ENABLE APPLDATA ALL' */
   /* 'CP MONITOR EVENT  ENABLE STORAGE'      */
   /* 'CP MONITOR EVENT  ENABLE I/O ALL'      */

   'PERFKIT'                    /* Invoke the PERFKIT module   @FC012BD*/

   Exit
   ```

   After:

   ```
   ...
   /*** Once you have PERFKIT enabled and running uncomment the      ***/
   /*** following comments                                           ***/
   'CP MONITOR SAMPLE ENABLE PROCESSOR'
   'CP MONITOR SAMPLE ENABLE STORAGE'
   'CP MONITOR SAMPLE ENABLE USER ALL'
   'CP MONITOR SAMPLE ENABLE I/O ALL'
   'CP MONITOR SAMPLE ENABLE NETWORK'
   'CP MONITOR SAMPLE ENABLE APPLDATA ALL'
   'CP MONITOR EVENT  ENABLE STORAGE'
   'CP MONITOR EVENT  ENABLE I/O ALL'
   ```

```
/* Increase the size of the SAMPLE CONFIG and EVENT CONFIG */
'CP MONITOR SAMPLE CONFIG SIZE 1200'
'CP MONITOR EVENT CONFIG SIZE 350'

'PERFKIT'                    /* Invoke the PERFKIT module   @FC012BD*/

Exit
====> file
```

You should now be ready to run the Performance Toolkit.

## 14.2.4 Starting the z/VM Performance Toolkit

To start the Performance Toolkit, enter the following command from the PERFSVM user ID:

```
===> perfkit
FCX001                 Performance Toolkit for VM                Autoscroll 12
FCXBAS500I Performance Toolkit for VM FL540 25Feb08
FCXAPP530I Connected to *IDENT for resource FCXRES00
FCXAPF530I Connected to *IDENT for resource FCXSYSTM
FCXTCP571I Connected to TCP/IP server TCPIP on path 0003
FCXTCR571I Connected to TCP/IP server TCPIP on path 0004
FCXAPP527I User PERFSVM connected on path 0007
FCXAPC535I Connected to resource FCXRES00 on path 0006, for S&F-Coll
FCXTCP575I WebServer host IP address is 9.12.5.22:00080
FCXTCR575I TCPRequest host IP address is 9.12.5.22:41780
FCXTCR590I TCPRequest interface activated
FCXTCP590I WebServer interface activated
```

Disconnect from PERFSVM now.

```
Command ===> disc
```

The Performance Toolkit should now be configured and running. It is now collecting basic data. However, to get relevant data from Linux images, they must be configured appropriately as described later.

## 14.2.5 Using the z/VM Performance Toolkit

The Performance Toolkit can be used using a Web browser or 3270 interface.

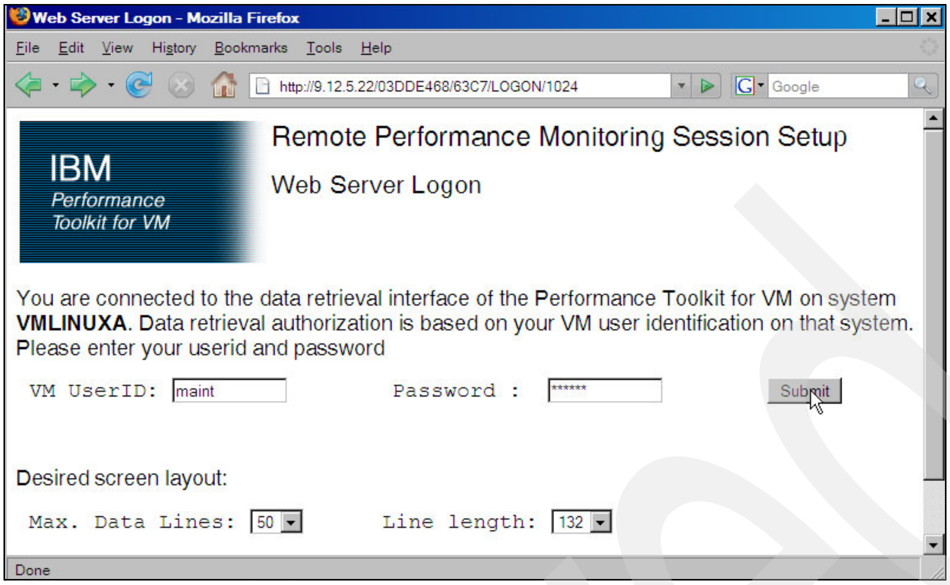
### Using a Web browser interface

To use the Web-enabled Performance Toolkit, perform the following steps:

1. Point a browser to your z/VM system at port 80. For example:

   ```
   http://9.12.5.22
   ```

2. You should see your system on the Web Session Setup window. Click it and you will be presented with the Web Server Logon window; see Figure 14-1 on page 203.


Web Server Logon - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://9.12.5.22/03DDE468/63C7/LOGON/1024
Google
IBM
Performance
Toolkit for VM
Done

Remote Performance Monitoring Session Setup

Web Server Logon

You are connected to the data retrieval interface of the Performance Toolkit for VM on system **VMLINUXA**. Data retrieval authorization is based on your VM user identification on that system. Please enter your userid and password

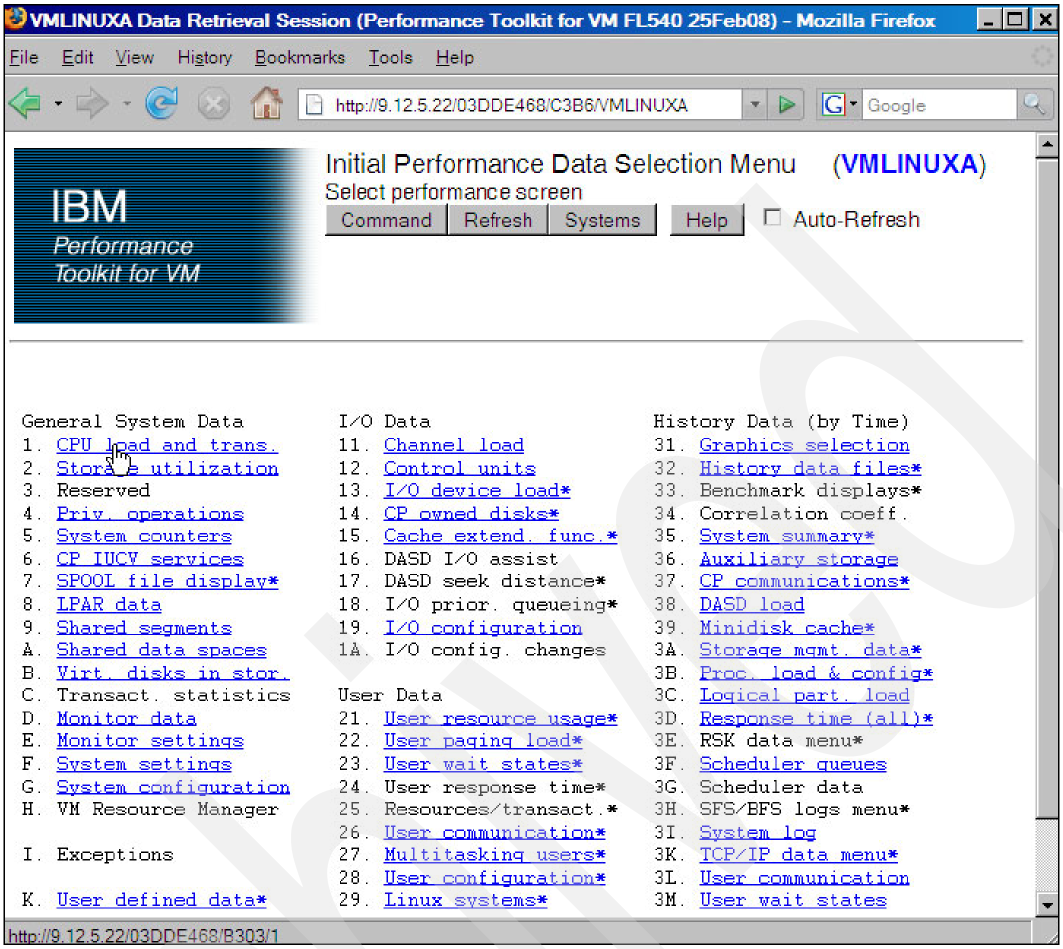
VM UserID: maint Password : ****** Submit

Desired screen layout:

Max. Data Lines: 50 Line length: 132

*Figure 14-1 Performance Toolkit logon window*

3. Enter any valid user ID and password (in this example, `MAINT`).
4. You should see the Central Monitoring System Load Overview with your system name on the left side.
5. Click your system name and you should see the Initial Performance Data Selection Menu window, as shown in Figure 14-2 on page 204.


VMLINUXA Data Retrieval Session (Performance Toolkit for VM FL540 25Feb08) - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://9.12.5.22/03DDE468/C3B6/VMLINUXA
Google
IBM
Performance Toolkit for VM
Command
Refresh
Systems
Help
□ Auto-Refresh
http://9.12.5.22/03DDE468/B303/1

Initial Performance Data Selection Menu (VMLINUXA)

Select performance screen

```
General System Data           I/O Data                     History Data (by Time)
1. CPU load and trans.        11. Channel load             31. Graphics selection
2. Storage utilization        12. Control units            32. History data files*
3. Reserved                   13. I/O device load*         33. Benchmark displays*
4. Priv. operations           14. CP owned disks*          34. Correlation coeff.
5. System counters            15. Cache extend. func.*     35. System summary*
6. CP IUCV services           16. DASD I/O assist          36. Auxiliary storage
7. SPOOL file display*        17. DASD seek distance*      37. CP communications*
8. LPAR data                  18. I/O prior. queueing*     38. DASD load
9. Shared segments            19. I/O configuration        39. Minidisk cache*
A. Shared data spaces         1A. I/O config. changes      3A. Storage mgmt. data*
B. Virt. disks in stor.                                    3B. Proc. load & config*
C. Transact. statistics       User Data                    3C. Logical part. load
D. Monitor data               21. User resource usage*     3D. Response time (all)*
E. Monitor settings           22. User paging load*        3E. RSK data menu*
F. System settings            23. User wait states*        3F. Scheduler queues
G. System configuration       24. User response time*      3G. Scheduler data
H. VM Resource Manager        25. Resources/transact.*     3H. SFS/BFS logs menu*
                              26. User communication*      3I. System log
I. Exceptions                 27. Multitasking users*      3K. TCP/IP data menu*
                              28. User configuration*      3L. User communication
K. User defined data*         29. Linux systems*           3M. User wait states
```

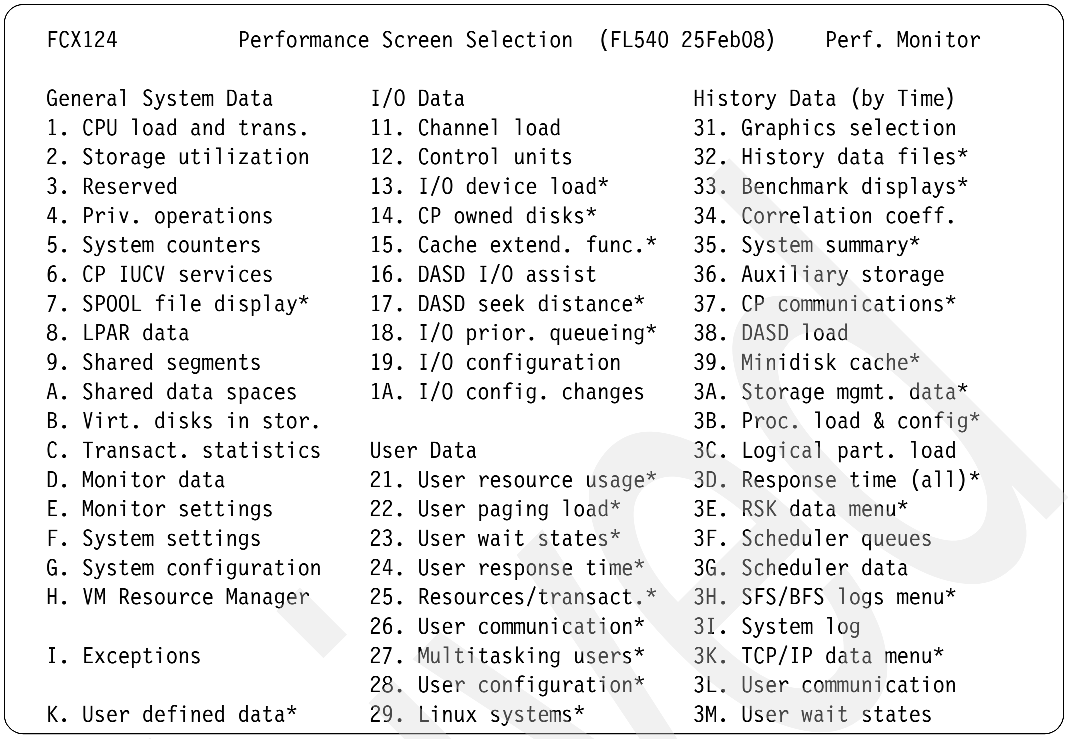
*Figure 14-2 Browser interface to the Performance Toolkit*

## Using a 3270 interface

**Logon** to PERFSVM. Run the **PROFILE EXEC** and you should be put into the Performance Toolkit for z/VM environment. The subcommand **monitor** should present the following window.

```
==> profile
FCXBAS500I Performance Toolkit for VM FL530
Monitor event started -- recording is activated
Monitor sample started -- recording is activated
...
FCX001                  Performance Toolkit for VM
  FCXBAS500I Performance Toolkit for VM FL530
  HCPMOF6229E Monitor event collection is already active.
  HCPMOG6229E Monitor sample collection is already active.

Command ==> monitor
```

You will see the window displayed in Figure 14-3 on page 205.



```
FCX124         Performance Screen Selection  (FL540 25Feb08)    Perf. Monitor

General System Data        I/O Data                  History Data (by Time)
1. CPU load and trans.     11. Channel load          31. Graphics selection
2. Storage utilization     12. Control units         32. History data files*
3. Reserved                13. I/O device load*      33. Benchmark displays*
4. Priv. operations        14. CP owned disks*       34. Correlation coeff.
5. System counters         15. Cache extend. func.*  35. System summary*
6. CP IUCV services        16. DASD I/O assist       36. Auxiliary storage
7. SPOOL file display*     17. DASD seek distance*   37. CP communications*
8. LPAR data               18. I/O prior. queueing*  38. DASD load
9. Shared segments         19. I/O configuration     39. Minidisk cache*
A. Shared data spaces      1A. I/O config. changes   3A. Storage mgmt. data*
B. Virt. disks in stor.                              3B. Proc. load & config*
C. Transact. statistics    User Data                 3C. Logical part. load
D. Monitor data            21. User resource usage*  3D. Response time (all)*
E. Monitor settings        22. User paging load*     3E. RSK data menu*
F. System settings         23. User wait states*     3F. Scheduler queues
G. System configuration    24. User response time*   3G. Scheduler data
H. VM Resource Manager     25. Resources/transact.*  3H. SFS/BFS logs menu*
                           26. User communication*   3I. System log
I. Exceptions              27. Multitasking users*   3K. TCP/IP data menu*
                           28. User configuration*   3L. User communication
K. User defined data*      29. Linux systems*        3M. User wait states
```

*Figure 14-3 Performance Toolkit 3270 Interface Main Menu window*

### Drilling down into report windows

You should now be able to use the active report windows. To drill down into these windows, move the cursor to any of the titles that are active (active titles display the number or letter in white; inactive titles are in green). Some of the more useful report windows to drill down into are:

```
21. User resource usage
22. User paging load
23. User wait states
28. User configuration
29. Linux systems
```

For example, to drill down into the Linux systems submenu window, enter the following command:

```
Command ===> 29
```

Then type `S` over the in the row corresponding to the report you want to see.

## 14.3 Monitoring Linux

Measurements can show resource consumption of the Linux guest as measured and dispatched by the VM host. It is also possible to measure performance data from within the Linux guest itself.

To monitor Linux performance data at this level, a data gatherer process must be running within each Linux guest you want to monitor. There are different ways of gathering this data. Data can be gathered in the kernel and as a user application. Novell SUSE SLES9, SLES10 and Red Hat RHEL 4 and RHEL 5 have been enabled for the kernel to gather performance data. There is a package called the Linux RMF™ PM Data Gatherer (also called rmfpms) that runs as a user application. Both of these data gatherers work in conjunction with the IBM z/VM Performance Toolkit.

## 14.3.1 Monitoring Linux with rmfpms

As a user application, the Linux RMF PM Data Gatherer (rmfpms) can be used. Currently it is not part of an IBM product and is intended for evaluation purposes only. A description of rmfpms is as follows:

> rmfpms is a modular data gatherer for Linux. The gathered data can be analyzed using the RMF PM client application. The performance data is accessible through XML over HTTP so you can easily exploit it in your own applications.

The following Web site is a starting point:

http://www-03.ibm.com/servers/eserver/zseries/zos/rmf/rmfhtmls/pmweb/pmlin.html

To download the data gatherer, scroll down and look for the following text and links:

```
* 31 bit data gatherer (kernel24 - 630 KB, kernel26 - 1040 KB).
* 64 bit data gatherer (kernel24 - 650 KB, kernel26 - 666 KB).
```

You can download the appropriate gatherer using a browser, or if you have access to the Internet, you can use an FTP client. You will want one of two files, depending on whether you have a 31-bit or 64-bit kernel:

```
rmfpms_s390_kernel26.tgz - for 31-bit distributions
rmfpms_s390x_kernel26.tgz - for 64-bit distributions
```

If you cannot access Internet directly from your Linux image, download the tar file to your workstation and copy it to Linux. Following is an example of downloading the tar file for 64-bit distributions directly from the Internet. Get an SSH session on the controller or any other virtual server.

1. Change directory to /usr/local/src/:

   ```
   # cd /usr/local/src
   ```

2. Download the appropriate tar file with the **wget** command. For a 64-bit SLES 9 and for SLES 10, use:

   ```
   # wget ftp://ftp.software.ibm.com/eserver/zseries/zos/rmf/rmfpms_s390x_kernel26.tgz
   --17:26:26--
   ftp://ftp.software.ibm.com/eserver/zseries/zos/rmf/rmfpms_s390x_kernel26.tgz
              => `rmfpms_s390x_kernel26.tgz'
   Resolving ftp.software.ibm.com... 9.17.252.40
   Connecting to ftp.software.ibm.com[9.17.252.40]:21... connected.
   Logging in as anonymous ... Logged in!
   ...
   100%[===================================>] 1,097,125    538.93K/s
   17:26:32 (537.12 KB/s) - `rmfpms_s390x_kernel26.tgz' saved [1097125]
   ```

3. For RHEL 4 and RHEL 5, use the following **wget** command:

   ```
   # wget ftp://ftp.software.ibm.com/eserver/zseries/zos/rmf/rmfpms_s390x_kernel26_rhel4.tgz
   ```

4. Untar the file with the **tar** command and change to the rmfpms/ directory:

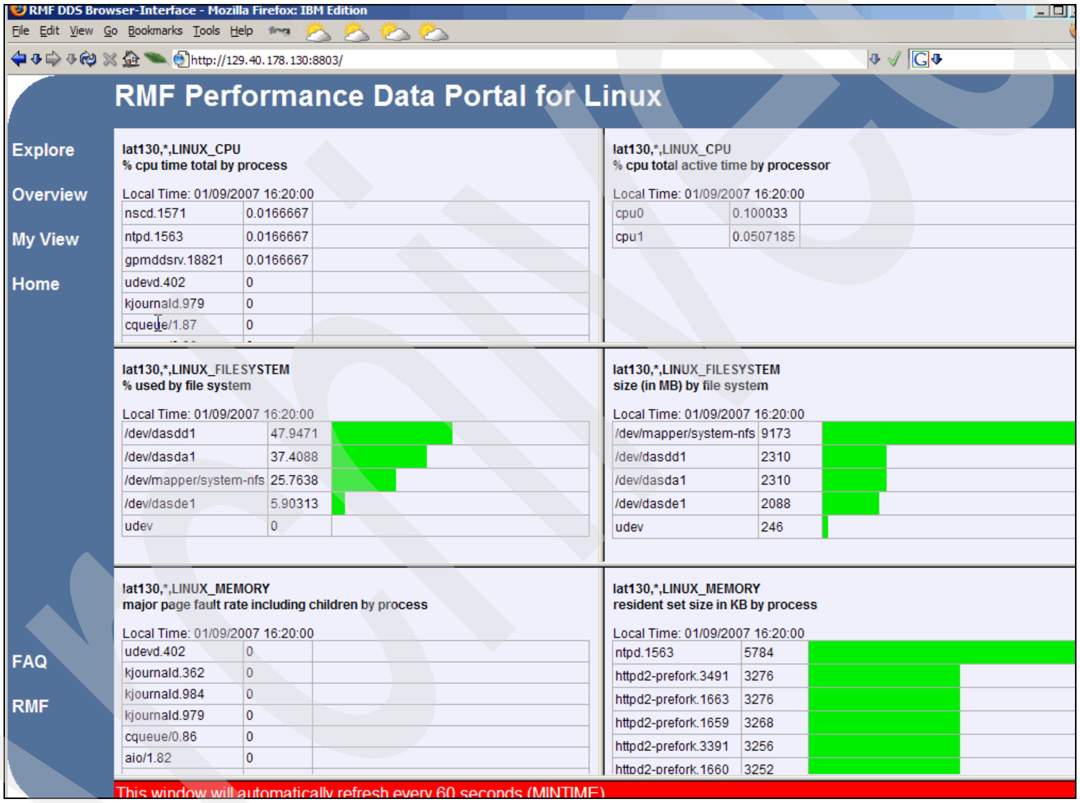
   ```
   # tar xzf rmfpms_s390x_kernel26.tgz
   # ls
   rmfpms  rmfpms_s390x_kernel26.tgz
   # cd rmfpms/
   ```

```
# ls
.rmfpms_config            README           bin  enable_autostart
.rmfpms_config_autostart  autostart_rmfpms doc  rmfpms
```

5. You should now be able to start **rmfpms** in the bin/ directory with the following command:

```
# bin/rmfpms start
Creating /root/rmfpms/.rmfpms ...
Starting performance gatherer backends ...
 DDSRV: RMF-DDS-Server/Linux-Beta (Sep  8 2007) started.
 DDSRV: Functionality Level=2.339
 DDSRV: Reading exceptions from gpmexsys.ini and gpmexusr.ini.
DDSRV: Server will now run as a daemon process.
done!
```

6. After it is running, you can view the performance data from a browser pointing to the Linux image and port 8803, as shown in Figure 14-4.


RMF DDS Browser-Interface - Mozilla Firefox: IBM Edition
File Edit View Go Bookmarks Tools Help
http://129.40.178.130:8803/
RMF Performance Data Portal for Linux
Explore
Overview
My View
Home
FAQ
RMF

lat130,*,LINUX_CPU
% cpu time total by process

Local Time: 01/09/2007 16:20:00

| | |
|---|---|
| nscd.1571 | 0.0166667 |
| ntpd.1563 | 0.0166667 |
| gpmddsrv.18821 | 0.0166667 |
| udevd.402 | 0 |
| kjournald.979 | 0 |
| cqueue/1.87 | 0 |

lat130,*,LINUX_CPU
% cpu total active time by processor

Local Time: 01/09/2007 16:20:00

| | |
|---|---|
| cpu0 | 0.100033 |
| cpu1 | 0.0507185 |

lat130,*,LINUX_FILESYSTEM
% used by file system

Local Time: 01/09/2007 16:20:00

| | |
|---|---|
| /dev/dasdd1 | 47.9471 |
| /dev/dasda1 | 37.4088 |
| /dev/mapper/system-nfs | 25.7638 |
| /dev/dasde1 | 5.90313 |
| udev | 0 |

lat130,*,LINUX_FILESYSTEM
size (in MB) by file system

Local Time: 01/09/2007 16:20:00

| | |
|---|---|
| /dev/mapper/system-nfs | 9173 |
| /dev/dasdd1 | 2310 |
| /dev/dasda1 | 2310 |
| /dev/dasde1 | 2088 |
| udev | 246 |

lat130,*,LINUX_MEMORY
major page fault rate including children by process

Local Time: 01/09/2007 16:20:00

| | |
|---|---|
| udevd.402 | 0 |
| kjournald.362 | 0 |
| kjournald.984 | 0 |
| kjournald.979 | 0 |
| cqueue/0.86 | 0 |
| aio/1.82 | 0 |

lat130,*,LINUX_MEMORY
resident set size in KB by process

Local Time: 01/09/2007 16:20:00

| | |
|---|---|
| ntpd.1563 | 5784 |
| httpd2-prefork.3491 | 3276 |
| httpd2-prefork.1663 | 3276 |
| httpd2-prefork.1659 | 3268 |
| httpd2-prefork.3391 | 3256 |
| httpd2-prefork.1660 | 3252 |

This window will automatically refresh every 60 seconds (MINTIME)

*Figure 14-4   Browser view of rmfpms interface*

You can also register Linux images with the Performance Toolkit, as described in 14.4, “Viewing Linux data in the Performance Toolkit” on page 208.

## 14.3.2 Monitoring Linux performance data from the kernel

To monitor Linux performance data directly from the kernel, the both of the following requirements must be true:

1. The APPLMON option must be set in the user directory.
2. Applmon data monitoring must be built into the kernel.

The first requirement should be true because the OPTION APPLMON was set for the controller and for Linux user IDs in earlier sections.

For the second requirement, SLES 9, SLES 10 and RHEL 5 now have this function built in. Details of this function are described in Chapter 15, “Linux monitor stream support for z/VM” in *Device Drivers, Features, and Commands* documentation for the October 2005 stream, which is available on the Web at:

http://www-128.ibm.com/developerworks/linux/linux390/october2005_documentation.html

A brief explanation of how to use this built-in monitoring function is given here.

1. There are three modules built into the kernel but not loaded, by default: `appldata_mem`, `appldata_os` and `appldata_net_sum`. You can verify that they are not loaded with the **lsmod** and **grep** commands:

```
# lsmod | grep appldata
```

2. There is no output, which means that no modules with the string `appldata` are loaded. Load those modules now using the **modprobe** command and verify they have been loaded:

```
# modprobe appldata_mem
# modprobe appldata_os
# modprobe appldata_net_sum
```

3. Now if you repeat the **lsmod** command, you should see the following:

```
# lsmod | grep appldata
appldata_net_sum      20064  0
appldata_os           22536  0
appldata_mem          20576  0
```

4. The directory in the virtual `/proc/` file system where the monitoring variables exist is `/proc/sys/appldata/`. In this directory there are five files:

| | |
|---|---|
| `timer` | Controls whether any data gathering is in effect |
| `interval` | Sets the interval, in milliseconds, that samples will be taken |
| `mem` | Controls the memory data gathering module |
| `os` | Controls the CPU data gathering module |
| `net_sum` | Controls the net data gathering module |

5. To turn on the built-in kernel monitoring, use the **echo** command to send a non-zero value into four of the five monitoring variables in the `/proc/` virtual file system:

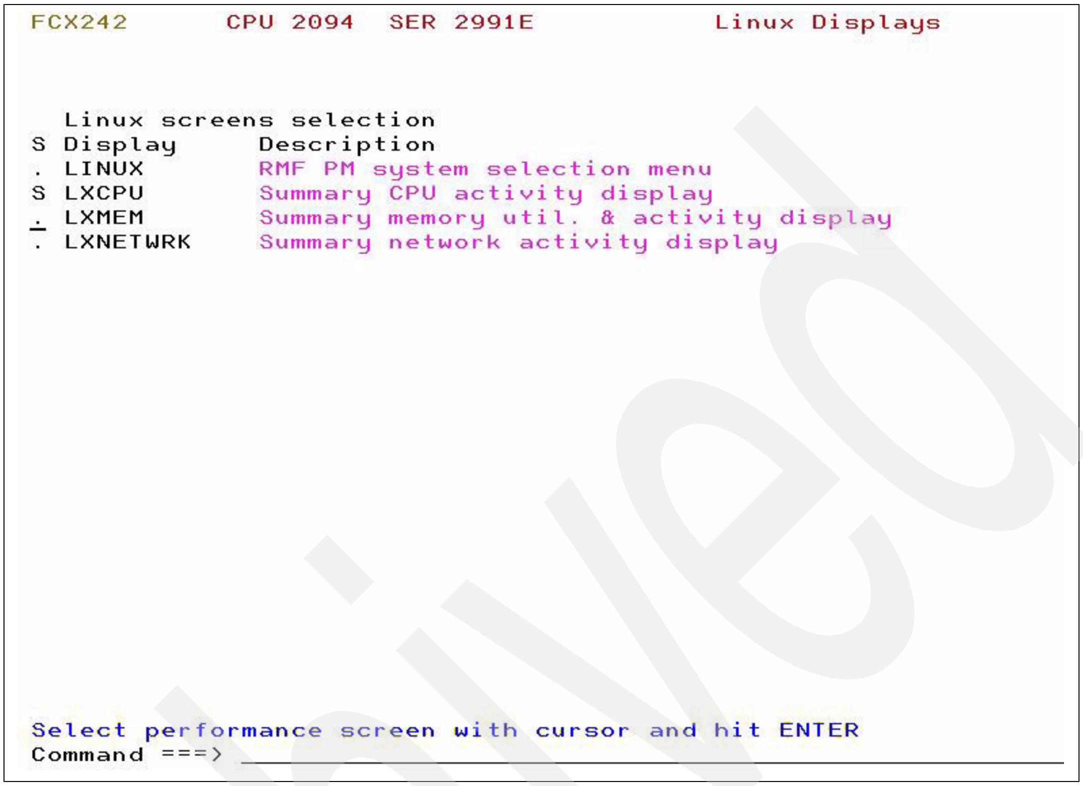
```
# echo 1 > /proc/sys/appldata/timer
# echo 1 > /proc/sys/appldata/mem
# echo 1 > /proc/sys/appldata/os
# echo 1 > /proc/sys/appldata/net_sum

# echo 20000 > /proc/sys/appldata/interval
```

Built-in kernel monitoring should now be turned on.

> **Note:** We do not recommend keeping Linux data gatherers running in all images all the time, because this may cause a serious performance problem. Instead, start gatherers only for specific periods of time and only when you have a performance problem to investigate. Also, as Linux monitoring data is captured, the Performance Toolkit’s minidisk space can fill up relatively quickly.

## 14.4 Viewing Linux data in the Performance Toolkit

After the system has had some time to collect data, you should be able to use the Performance Toolkit to view Linux performance data. There should be entries in the `FCONX LINUXUSR` file as completed in 14.2.2, “Configuring Web Browser support” on page 199. To view that data, drill down into menu 29, `Linux systems`. This can be done either from the browser interface or the 3270 interface as shown in Figure 14-5.



```
FCX242       CPU 2094   SER 2991E           Linux Displays


  Linux screens selection
S Display      Description
. LINUX        RMF PM system selection menu
S LXCPU        Summary CPU activity display
_ LXMEM        Summary memory util. & activity display
. LXNETWRK     Summary network activity display











Select performance screen with cursor and hit ENTER
Command ===> ____________________________________________
```

*Figure 14-5 Linux Guest Systems sub menu*

Type S over the period on the left side of the submenu window in the row corresponding to the report you want to see. You should see a new report window with the Linux guest systems CPU overview.

## 14.5 Common Performance Toolkit issue

When you browse performance toolkit windows, check the Performance Toolkit console log exceptions. If you get `FCXPMN446E Incomplete monitor data: DCSS size too small` or `FCXPMN446E Incomplete monitor data: SAMPLE CONFIG size too small` you have to act; otherwise, data in windows may be wrong or missing (that is, your data may be suspect).

To enlarge SAMPLE CONFIG or EVENT CONFIG, you may use or add the following commands to the PERFSVM PROFILE EXEC:

```
'CP MONITOR SAMPLE CONFIG SIZE xxx'
'CP MONITOR EVENT CONFIG SIZE yyy'
```

To enlarge DCSS, here is a hint. First, determine where the current MONDCSS is located by entering:

```
q nss name mondcss map
```

This results in output that is similar to:

```
FILE FILENAME FILETYPE MINSIZE BEGPAG ENDPAG TYPE CL #USERS PARMREGS VMGROUP
7271 MONDCSS CPDCSS N/A 03300 06FFF SC R 00007 N/A N/A
```

In this example, it starts at x3300 and ends at x6FFF (this is a PAGE boundary). You also need to issue **`q nss all map`** to determine where the new MONDCSS can be located with no overlap.

**Important:** Check carefully to make sure that the new MONDCSS does not overlap any other NSS. If it does, PERKIT may not be able to use it.

One way to test whether the MONDCSS segment can be loaded is to issue SEGMENT LOAD MONDCSS before you start PERFKIT. If it returns any error (for example, DMSDCS343E), there may be an overlap and you have to redefine the MONDCSS location.

Often, you can simply raise the ending location of the current MONDCSS to solve this problem.

First, delete the old MONDCSS by issuing:

```
PURGE NSS NAME MONDCSS
```

If any users are currently connected to the MONDCSS, you need to log off those users. You can determine who is connected by issuing **`Q MONITOR`**.

Next, create a new MONDCSS:

```
CP DEFSEG MONDCSS 03300-09FFF SC RSTD
```

Save the new MONDCSS:

```
CP SAVESEG MONDCSS
```

**Recommendation:** There is a relationship between the PERFSVM virtual machine storage size and the MONDCSS segment size, as described here.

- The PERFSVM virtual machine storage size should be of ample size to contain the entire MONDCSS segment, plus some CMS pointer tables.
- Or, the MONDCSS must be located outside the size of the PERFSVM virtual machine storage size (the preferred solution).

Then you can restart PERFKIT. If you receive any messages from the PERFKIT that the MONDCSS SAMPLE CONFIG size is too small, you have to enlarge it as previously explained. The same would apply to the MONITOR EVENT CONFIG SIZE, if you receive a PERFKIT message pertaining to that.

If you continue to receive MONDCSS SIZE TOO SMALL messages, you will need to restart this process over again, increasing the total size of the MONDCSS segment.

RESTART the PERFKIT and monitor the startup for at least 2 monitor interval timeframes to ensure things are working and there are no error type messages.

A

# References

This appendix describes the location and content of z/VM configuration files, and provides quick reference sheets for the XEDIT and vi text editors.

# z/VM configuration files

z/VM differs from Linux in regard to the location and number of configuration files. In Linux, there are many configuration files and most of them are in or under the `/etc/` directory. On z/VM, there are relatively few configuration files. However, they are on many different minidisks.

Table 1 summarizes the location and content of z/VM configuration files.

*Table 1   z/VM configuration files*

| File | Location | Description |
|---|---|---|
| SYSTEM CONFIG | MAINT CF1 | This is the operating system’s main configuration file. It defines the system name, the CP volumes, user volumes and other settings. |
| USER DIRECT | MAINT 2CC | This file defines the user directory. All user IDs or virtual machines known to the system are defined here (assuming a directory maintenance product is not being used). |
| <System_ID> TCPIP | TCPMAINT 198 | This file defines the resources for the primary z/VM TCP/IP stack, including TCP/IP address, OSA resources, subnet mask and gateway. It is initially created by the IPWIZARD tool as `PROFILE TCPIP`. |
| SYSTEM DTCPARMS | TCPMAINT 198 | This file is created to define the TCP/IP stacks on the system. It is initially created by the IPWIZARD tool. |
| TCPIP DATA | TCPMAINT 592 | This file defines the DNS server, the domain name and some other settings. It is initially created by the IPWIZARD tool. |
| PROFILE EXEC | AUTOLOG1 191 | This file is a REXX EXEC that is run when the system starts up. It is analogous to the /etc/inittab file in Linux. |

# Quick reference sheets

This section contains quick reference sheets for the XEDIT and vi editors.

## XEDIT quick reference sheet

XEDIT has line commands which are typed on the command line (===>) and prefix commands which are typed over the line numbers on the left side of the window.

### Line commands

```
a                      Add a line
a<n>                   Add ‘n’ lines
c/<old>/<new>/ <n> <m> Search for string ‘old’ and replace it with ‘new’ for ‘n’ lines
    below the current line and ‘m’ times on each line. ‘*’ can be used for ‘n’ and ‘m’
/<string>              Search for ‘string’ from the current line
-/<string>             Search backwards for ‘string’
all /<string>/         Show all occurences of ‘string’ and hide other lines
bottom                 Move to the bottom of the file
top                    Move to the top of the file
down <n>               Move down ‘n’ lines
up <n>                 Move up ‘n’ lines
file                   Save the current file and exit XEDIT
ffile                  Save the current file and exit but don’t warn of overwrite
save                   Save the current file but don’t exit
```

| | |
|---|---|
| quit | Exit XEDIT if no changes have been made |
| qquit | Exit XEIDT even if changes have not been saved |
| left <n> | Shift ‘n’ characters to the left |
| right <n> | Shift ‘n’ characters to the right |
| get <file> | Copy file and insert past the current line |
| :<n> | Move to line ‘n’ |
| ? | Display last command |
| = | Execute last command |
| x <file> | Edit ‘file’ and put it into the XEDIT “ring” |
| x | Move to the next file in the ring |

### Prefix commands

| | |
|---|---|
| a | Add one line |
| a<n> | Add 'n' lines |
| c | Copies one line |
| cc | Copies a block of lines |
| d | Deletes one line |
| dd | Deletes a block of lines |
| f | Line after which a copy (c) or a move (m) is to be inserted |
| p | Line before which a copy (c) or a move (m) is to be inserted |
| i | Insert a line |
| i<n> | Insert 'n' lines |
| m | Move one line |
| mm | Move a block of lines |
| " | Replicate a line |
| "<n> | Replicate a line 'n' times |
| "" | Replicate a block of lines |

## vi quick reference sheet

Following is a small, but commonly used, subset of vi commands.The vi editor has three modes:

1. Input mode - using the Insert key, **i**, **o** (add a line below), **0** (add a line above) and other commands puts you in this mode. When you are in this mode, you will see the text `--INSERT--` in the last line.
2. Command mode - 'Esc' gets you out of input mode and into command mode.

| | |
|---|---|
| i | brings you back to input mode |
| dd | deletes a line and puts it in the buffer |
| <n>dd | delete <n> lines |
| x | delete a character |
| dw | delete a word |
| p | add the buffer past the current location |
| P | add the buffer before the current location |
| o | add a line and go into insert mode |
| /string - search for string | |
| n | do the last command again (this can be powerful) |
| jkl; | cursor movement |
| A | add text at the end of the line |
| <nn>G | go to line <nn> |
| G | go to the last line in the file |
| yy | yank a line (copy into buffer) |
| <n>yy | yank n lines |

3. Command line mode - pressing the colon (:) key brings you to this mode.

| | |
|---|---|
| :wq | save (write & quit) |
| :q! | quit and discard changes |

```
:<nn>     go to line number <nn>
:r <file> read <file> into the current file
:1,$s/old/new/g globally replace <old> with <new>
:help     give help
```

B

# z/VM source code

This appendix lists the source code associated with this book.

## Obtaining and using the Web material

The pdf of this book is available on the Internet at the following address:

`http://www.redbooks.ibm.com/abstracts/sg247493.html`

The files associated with this book are in a gzipped tar file at the following address:

`ftp://www.redbooks.ibm.com/redbooks/SG247493`

Download the tar file to your NFS server and use it as is described in 7.2.1, “Copying files to the controller” on page 109. After untarring the file, you will have a directory named `virt-cookbook-S10SP2`.

Under that directory are the following files and directory:

| | |
|---|---|
| `README.txt` | The main README file |
| **`clone.sh`** | The script to clone the golden image to a target user ID |
| `vm/` | A directory containing files used on z/VM |

## z/VM REXX EXECs and XEDIT macros

This section lists z/VM REXX EXECs and XEDIT macros.

### The CPFORMAT EXEC

Following is the code for the EXEC that formats multiple disks using **CPFMTXA** (described in 4.6.1, “Formatting the paging volumes” on page 48):

```
/*+------------------------------------------------------------------+*/
/*| EXEC: CPFORMAT - wrapper around CPFMTXA to format many DASD      |*/
/*|  retVal: 0 - success                                             |*/
/*|          1 - help was asked for or given                         |*/
/*|          2 - user is not sure                                    |*/
/*|          3 - DASD (minidisk) range is not valid                  |*/
/*|          4 - at least one DASD (minidisk) is reserved to MAINT   |*/
/*+------------------------------------------------------------------+*/
/* For details on how this EXEC is used, see one of the two  books:
"z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES 10 SP2"
on the Web at: http://www.redbooks.ibm.com/abstracts/SG247493.html
 -or-
"z/VM and Linux on IBM System z: The Virtualization Cookbook for RHEL 5.2"
on the Web at: http://www.redbooks.ibm.com/abstracts/SG247492.html */

/*-------------------------------------------------------------------
THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT
LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT,
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES
-------------------------------------------------------------------*/
firstChar = 'M' /* change this for an LPAR ID other than 'M' */
parse upper arg dasds "AS " type
if ((dasds = '') | (dasds = '?')) then call help
labelPrefix = getLabelPrefix(firstChar type)
numDasd = parseDasd(dasds)
answer = areYouSure(type)
if (answer = 'Y') then  /* the user is sure */
do
  formatted = ""
  retVal = doFormat(labelPrefix numDasd type)
  call doReport retVal
end
else
  retVal = 2
exit retVal

/*+------------------------------------------------------------------+*/
help: procedure
/*+------------------------------------------------------------------+*/
  parse source . . fn .
  say ''
  say 'Synopsis:'
  say ''
  say '  Format one or a range of DASD as page, perm, spool or temp disk space'
  say '  The label written to each DASD is V<t><xxxx> where:'
  say '    <t> is type - P (page), M (perm), S (spool) or T (Temp disk)'
  say '    <xxxx> is the 4 digit address'
  say ''
  say 'Syntax is:'
  say "                                             .-PAGE-."
  say "   >>--CPFORMAT--.-rdev--------------.--AS---+-PERM-+---------><"
  say "                 | <---------------< |       '-SPOL-'"
  say "                 '-rdev1-rdev2-------'  "
  say ''
exit 1

/*+------------------------------------------------------------------+*/
areYouSure: procedure
/*| Show minidisks, ask are you sure                                 |*/
/*|  parm 1: type - PERM, PAGE, or SPOL                              |*/
/*|  retVal: firstChar - LPAR identifier, 'V' by default             |*/
/*+------------------------------------------------------------------+*/
  arg type
  say ''
  say 'WARNING - this will destroy data!'
  say 'ARE YOU SURE you want to format the DASD as' type 'space (y/n)?'
  parse upper pull answer
return  substr(answer, 1, 1) /* from areYouSure */

/*+------------------------------------------------------------------+*/
getLabelPrefix: procedure
/*| Return first two chararcters of label                            |*/
/*|  parm 1: firstChar - LPAR identifier, 'V' by default             |*/
/*|  retVal: the two character label prefix                          |*/
/*+------------------------------------------------------------------+*/
  arg firstChar type
  select
  when (type = PERM) then
    labelPrefix = firstChar||'M' /* for VM Minidisk */
  when (type = PAGE) then
    labelPrefix = firstChar||'P' /* for VM Page */
  when (type = SPOL) then
    labelPrefix = firstChar||'S' /* for VM Spool */
  otherwise
    do
      say 'Error: "AS" must be present, type must be PERM, PAGE or SPOL'
      call help
    end /* otherwise */
  end /* select */
return labelPrefix /* from getLabelPrefix */

/*+------------------------------------------------------------------+*/
parseDasd: procedure expose dasdList.
/*| parse all dasd into an array verifying all are attached          |*/
/*|  parm 1: dasds - the list of dasd passed in                      |*/
/*|  retVal: number of DASD in dasdList                              |*/
/*+------------------------------------------------------------------+*/
  arg dasds
  numDasd = 0
  say ''
  say 'Format the following DASD:'
  do while (dasds <> '')
```

```
    parse upper var dasds dasd dasds
    dashPos = pos('-', dasd)
    if (dashPos = 0) then  /* there is just one DASD */
    do
      numDasd = numDasd + 1
      dasdList.numDasd = dasd
      'CP Q MDISK' dasdList.numDasd 'LOCATION'
      if (rc <> 0) then
      do
        say 'Return code from Q MDISK =' rc
        say 'Are all DASD ATTached?'
        exit 3
      end
      call checkReserved(dasdList.numDasd)
    end /* do */
    else /* process the range of DASD */
    do
      startRange = substr(dasd, 1, dashPos - 1)
      endRange = substr(dasd, dashPos + 1, length(dasd) - dashPos)
      do i = x2d(startRange) to x2d(endRange)
        numDasd = numDasd + 1
        dasdList.numDasd = d2x(i)
        'CP Q MDISK' dasdList.numDasd 'LOCATION'
        if (rc <> 0) then
        do
          say 'Return code from Q MDISK =' rc
          exit 3
        end
        call checkReserved(dasdList.numDasd)
      end /* do i */
    end /* else */
  end /* do while */
return numDasd /* from parseDasd */

/*+------------------------------------------------------------------+*/
doFormat: procedure expose dasdList. formatted
/*| Format all DASD specified using CPFMTXA                          |*/
/*|  parm 1: labelPrefix - the two character label prefix           |*/
/*|  parm 2: numDasd - number of DASD in the array dasdList          |*/
/*|  parm 3: type - the type of DASD format                          |*/
/*|  retVal: 0 = success                                             |*/
/*+------------------------------------------------------------------+*/
  arg labelPrefix numDasd type
  'CP TERM MORE 1 1'
  do i = 1 to numDasd
    label = getLabel(labelPrefix dasdList.i)
    retVal = formatOne(dasdList.i type label)
    if (retVal ^= 0) then
    do
      say "Error from CPFMTXA on DASD" label "rc =" retVal
      leave /* error - abort! */
    end
    formatted = formatted label
  end /* do i = */
  'CP TERM MORE 50 10'
return retVal /* from doFormat */

/*+------------------------------------------------------------------+*/
checkReserved: procedure
/*| Try copying an already formatted DASD then relabelling it       |*/
/*|  parm 1: source                                                    |*/
/*|  parm 2: target                                                    |*/
/*|  parm 3: label                                                     |*/
/*+-------------------------------------------------------------------+*/
  arg dasd
  /* create a list of reserved dasd - this is somewhat hokey to be sure
     but it's better to be hokey than to format system minidisks! */
  resvd1 = "0122 0123 0124 0125 0190 0191 0193 0194 019D 019E 0201 02A2"
  resvd2 = "02A4 02A6 02C2 02C4 02CC 02D2 0319 03A2 03A4 03A6 03B2 03C2"
  resvd3 = "03C4 03D2 0400 0401 0402 0405 0490 0493 049B 049E 04A2 04A4"
  resvd4 = "04A6 04B2 04C2 04C4 04D2 0500 051D 05A2 05A4 05A6 05B2 05C2"
  resvd5 = "05C4 05D2 05E5 05E6 06A2 06A4 06A6 06B2 06C2 06C4 06D2 07A2"
  resvd6 = "07A4 07A6 07B2 07C2 07C4 07D2 0CF1 0CF2 0CF3"
  reserved = resvd1 resvd2 resvd3 resvd4 resvd5 resvd6
  if (index(reserved, dasd) <> 0) then /* MAINT minidisk - ABORT! */
  do
    say 'Minidisk' dasd 'is a reserved MAINT minidisk'
    say 'This must be formatted manually using a different vaddr'
    exit 4
  end /* if dasd is reserved */
return /* from checkReserved */


/*+-------------------------------------------------------------------+*/
doReport: procedure expose dasds formatted
/*| Report on the newly labelled DASD                                  |*/
/*|  parm 1: formatSuccess - 0=all is well, non-0= a format failed    |*/
/*|  retVal: 0 = success                                               |*/
/*+-------------------------------------------------------------------+*/
  arg formatSuccess
  if (formatSuccess ^= 0) then
    say 'Error was encountered! retVal from CPFMTXA =' formatSuccess
    if (formatted = '') then
      say "No DASD were successfully formatted"
    else
      say "DASD successfully formatted:" formatted
  'DETACH' dasds
  'ATTACH' dasds '*'
  say ''
  say 'DASD status after:'
  'CP Q MDISK' dasds 'LOCATION'
return 0 /* from doReport */

/*+-------------------------------------------------------------------+*/
formatOne: procedure
/*| Format a DASD via DDR                                              |*/
/*|  parm 1: disk - the vaddr to be formatted                          |*/
/*|  parm 2: type - PAGE, SPOL or PERM                                 |*/
/*|  parm 3: label - the six character label                           |*/
/*+-------------------------------------------------------------------+*/
  arg disk type label
  queue 'FORMAT'
  queue disk
  queue '0 END'
  queue label
  queue 'YES'
  queue type '0 END'
  queue 'END'
  'CPFMTXA'
  retVal = rc
```

```
return retVal /* from formatOne */

/*+------------------------------------------------------------------+*/
getLabel: procedure
/*| Compose the six character label of a minidisk                    |*/
/*|  parm 1: labelPrefix - first two characters of label            |*/
/*|  parm 2: disk - vaddr of length 1, 2, 3 or 4                     |*/
/*|  return: the 6 character label                                   |*/
/*+------------------------------------------------------------------+*/
  arg labelPrefix disk
  diskLen = length(disk)
  select
  when (diskLen = 1) then /* insert 3 zeros */
    label = labelPrefix||'000'||disk
  when (diskLen = 2) then /* insert 2 zeros */
    label = labelPrefix||'00'||disk
  when (diskLen = 3) then /* insert a zero */
    label = labelPrefix||'0'||disk
  otherwise /* it must be length 4 or query would have failed */
    label = labelPrefix||disk
  end /* select */
return label /* from getLabel */
```

## The CHPW540 XEDIT macro

Following is the code for the XEDIT macro that changes all passwords in the z/VM 5.4 USER DIRECT file:

```
/*+------------------------------------------------------------------+*/
/* CHPW540 XEDIT - change all passwords in z/VM 5.2 USER DIRECT file */
/*+------------------------------------------------------------------+*/
/* For details on how this macro is used, see the book:
"z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES 10 SP2"
on the Web at: http://www.redbooks.ibm.com/abstracts/SG247493.html */
/*------------------------------------------------------------------
THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT
LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT,
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES
-------------------------------------------------------------------*/

parse arg fn ft fm '(' options ')' newPass .
if (length(newPass) > 8) then
do
  say "Error: new password must be 8 characters or fewer"
  exit
end
say ''
say 'Changing all passwords to:' newPass
say ''

/* set some values */
'command set stay on'
```

```
'command set num on'
'command set nulls on'
'command set serial off'
'command set cmdline bottom'
'command set curline on 3'
'command set serial off'
'command set scale off'
'command set case m i'
'command set pre off'
'command set v 1 80'

/* change user ID passwords */
'command c/MAINT MAINT/MAINT' newPass'/*'
'command c/AVSVM AVSVM/AVSVM' newPass'/*'
'command c/TSAFVM TSAFVM/TSAFVM' newPass'/*'
'command c/GCS GCS/GCS' newPass'/*'
'command c/GCSXA GCSXA/GCSXA' newPass'/*'
'command c/AUDITOR AUDITOR/AUDITOR' newPass'/*'
'command c/AUTOLOG1 AUTOLOG1/AUTOLOG1' newPass'/*'
'command c/AUTOLOG2 AUTOLOG2/AUTOLOG2' newPass'/*'
'command c/BLDCMS  BLDCMS/BLDCMS' newPass'/*'
'command c/BLDNUC  BLDNUC/BLDNUC' newPass'/*'
'command c/BLDRACF  BLDRACF/BLDRACF' newPass'/*'
'command c/BLDSEG BLDSEG/BLDSEG' newPass'/*'
'command c/CMS1  CMS1/CMS1' newPass'/*'
'command c/CMSBATCH CMSBATCH/CMSBATCH' newPass'/*'
'command c/DISKACNT DISKACNT/DISKACNT' newPass'/*'
'command c/EREP EREP/EREP' newPass'/*'
'command c/IBMUSER IBMUSER/IBMUSER' newPass'/*'
'command c/LGLOPR  LGLOPR/LGLOPR' newPass'/*'
'command c/MIGMAINT MIGMAINT/MIGMAINT' newPass'/*'
'command c/MONWRITE MONWRITE/MONWRITE' newPass'/*'
'command c/OP1 OP1/OP1' newPass'/*'
'command c/OPERATNS OPERATNS/OPERATNS' newPass'/*'
'command c/OPERATOR OPERATOR/OPERATOR' newPass'/*'
'command c/OPERSYMP OPERSYMP/OPERSYMP' newPass'/*'
'command c/SYSADMIN SYSADMIN/SYSADMIN' newPass'/*'
'command c/SYSDUMP1 SYSDUMP1/SYSDUMP1' newPass'/*'
'command c/SYSMAINT SYSMAINT/SYSMAINT' newPass'/*'
'command c/SYSMON  SYSMON/SYSMON' newPass'/*'
'command c/VMRMADMN VMRMADMN/VMRMADMN' newPass'/*'
'command c/VMRMSVM VMRMSVM/VMRMSVM' newPass'/*'
'command c/VMSERVR VMSERVR/VMSERVR' newPass'/*'
'command c/VMSERVS VMSERVS/VMSERVS' newPass'/*'
'command c/VMSERVU VMSERVU/VMSERVU' newPass'/*'
'command c/VMUTIL VMUTIL/VMUTIL' newPass'/*'
'command c/VSMPROXY VSMPROXY/VSMPROXY' newPass'/*'
'command c/VSMREQIN VSMREQIN/VSMREQIN' newPass'/*'
'command c/VSMREQIU VSMREQIU/VSMREQIU' newPass'/*'
'command c/VSMSERVE VSMSERVE/VSMSERVE' newPass'/*'
'command c/VSMWORK1 VSMWORK1/VSMWORK1' newPass'/*'
'command c/VSMWORK2 VSMWORK2/VSMWORK2' newPass'/*'
'command c/VSMWORK3 VSMWORK3/VSMWORK3' newPass'/*'
'command c/ZVMMAPLX MAINT/ZVMMAPLX' newPass'/*'
'command c/5684042J 5684042J/5684042J' newPass'/*'
'command c/4OSASF40 4OSASF40/4OSASF40' newPass'/*'
'command c/OSADMIN1 OSADMIN1/OSADMIN1' newPass'/*'
'command c/OSADMIN2 OSADMIN2/OSADMIN2' newPass'/*'
'command c/OSADMIN3 OSADMIN3/OSADMIN3' newPass'/*'
'command c/OSAMAINT OSAMAINT/OSAMAINT' newPass'/*'
```

```
'command c/OSASF OSASF/OSASF' newPass'/*'
'command c/5VMRSC40 5VMRSC40/5VMRSC40' newPass'/*'
'command c/RSCS RSCS/RSCS' newPass'/*'
'command c/RSCSAUTH RSCSAUTH/RSCSAUTH' newPass'/*'
'command c/RSCSDNS RSCSDNS/RSCSDNS' newPass'/*'
'command c/XCHANGE XCHANGE/XCHANGE' newPass'/*'
'command c/5VMTCP40 5VMTCP40/5VMTCP40' newPass'/*'
'command c/TCPIP TCPIP/TCPIP' newPass'/*'
'command c/TCPMAINT TCPMAINT/TCPMAINT' newPass'/*'
'command c/ADMSERV ADMSERV/ADMSERV' newPass'/*'
'command c/DHCPD DHCPD/DHCPD' newPass'/*'
'command c/DTCVSW1  DTCVSW1/DTCVSW1' newPass'/*'
'command c/DTCVSW2  DTCVSW2/DTCVSW2' newPass'/*'
'command c/FTPSERVE FTPSERVE/FTPSERVE' newPass'/*'
'command c/IMAP IMAP/IMAP' newPass'/*'
'command c/IMAPAUTH IMAPAUTH/IMAPAUTH' newPass'/*'
'command c/LDAPSRV  LDAPSRV/LDAPSRV' newPass'/*'
'command c/LPSERVE LPSERVE/LPSERVE' newPass'/*'
'command c/MPROUTE MPROUTE/MPROUTE' newPass'/*'
'command c/NAMESRV NAMESRV/NAMESRV' newPass'/*'
'command c/NDBPMGR NDBPMGR/NDBPMGR' newPass'/*'
'command c/NDBSRV01 NDBSRV01/NDBSRV01' newPass'/*'
'command c/PORTMAP PORTMAP/PORTMAP' newPass'/*'
'command c/REXECD REXECD/REXECD' newPass'/*'
'command c/RXAGENT1 RXAGENT1/RXAGENT1' newPass'/*'
'command c/SMTP SMTP/SMTP' newPass'/*'
'command c/SNALNKA SNALNKA/SNALNKA' newPass'/*'
'command c/SNMPD SNMPD/SNMPD' newPass'/*'
'command c/SNMPQE SNMPQE/SNMPQE' newPass'/*'
'command c/SNMPSUBA SNMPSUBA/SNMPSUBA' newPass'/*'
'command c/SSLSERV SSLSERV/SSLSERV' newPass'/*'
'command c/TFTPD TFTPD/TFTPD' newPass'/*'
'command c/UFTD    UFTD/UFTD' newPass'/*'
'command c/VMKERB VMKERB/VMKERB' newPass'/*'
'command c/VMNFS VMNFS/VMNFS' newPass'/*'
'command c/X25IPI X25IPI/X25IPI' newPass'/*'
'command c/5VMDIR40 5VMDIR40/5VMDIR40' newPass'/*'
'command c/5VMRAC40 5VMRAC40/5VMRAC40' newPass'/*'
'command c/RACFSMF RACFSMF/RACFSMF' newPass'/*'
'command c/RACFVM RACFVM/RACFVM' newPass'/*'
'command c/RACMAINT RACMAINT/RACMAINT' newPass'/*'
'command c/5VMPTK40 5VMPTK40/5VMPTK40' newPass'/*'
'command c/PERFSVM  PERFSVM/PERFSVM' newPass'/*'
'command c/5VMHCD40 5VMHCD40/5VMHCD40' newPass'/*'
'command c/CBDIODSP CBDIODSP/CBDIODSP' newPass'/*'
'command c/LNXMAINT LNXMAINT/LNXMAINT' newPass'/*'

/* change mindisk passwords */
'command c/ALL      WRITE    MULTIPLE/ALL' newPass newPass'/*'
'command c/ALL      WTCPMAIN MTCPMAIN/ALL' newPass newPass'/*'
'command c/RADMSERV WADMSERV MADMSERV/'newPass newPass newPass'/*'
'command c/RAUDITOR WAUDITOR MAUDITOR/'newPass newPass newPass'/*'
'command c/RAUTOLOG WAUTOLOG MAUTOLOG/'newPass newPass newPass'/*'
'command c/RAVSOBJ  WAVSOBJ  MAVSOBJ/'newPass newPass newPass'/*'
'command c/RBATCH   WBATCH   MBATCH/'newPass newPass newPass'/*'
'command c/RCATALOG WCATALOG/'newPass newPass'/*'
'command c/RCONTROL WCONTROL/'newPass newPass'/*'
'command c/RCRRLOG1 WCRRLOG1/'newPass newPass'/*'
'command c/RDATA    WDATA/'newPass newPass'/*'
'command c/RDHCPD   WDHCPD   MDHCPD/'newPass newPass newPass'/*'
```

```
'command c/RDVF     WDVF     MDVF/'newPass newPass newPass'/*'
'command c/READ     WRITE    MULTIPLE/'newPass newPass newPass'/*'
'command c/READ     WRITE/'newPass newPass'/*'
'command c/RFTPSERV WFTPSERV MFTPSERV/'newPass newPass newPass'/*'
'command c/RGCS     WGCS     MGCS/'newPass newPass newPass'/*'
'command c/RIMAP    WIMAP    MIMAP/'newPass newPass newPass'/*'
'command c/RLOG1    WLOG1/'newPass newPass newPass'/*'
'command c/RLOG2    WLOG2/'newPass newPass newPass'/*'
'command c/RLPSERVE WLPSERVE MLPSERVE/'newPass newPass newPass'/*'
'command c/RMAINT   WMAINT   MMAINT/'newPass newPass newPass'/*'
'command c/RMPROUTE WMPROUTE MMPROUTE/'newPass newPass newPass'/*'
'command c/RNAMESRV WNAMESRV MNAMESRV/'newPass newPass newPass'/*'
'command c/RNDBPMGR WNDBPMGR MNDBPMGR/'newPass newPass newPass'/*'
'command c/RNDBSRVO WNDBSRVO MNDBSRVO/'newPass newPass newPass'/*'
'command c/RPORTMAP WPORTMAP MPORTMAP/'newPass newPass newPass'/*'
'command c/RREXECD  WREXECD  MREXECD/'newPass newPass newPass'/*'
'command c/RSERVER  WSERVER/'newPass newPass'/*'
'command c/RSMTP    WSMTP    MSMTP/'newPass newPass newPass'/*'
'command c/RSNALNKA WSNALNKA MSNALNKA/'newPass newPass newPass'/*'
'command c/RSNMPD   WSNMPD   MSNMPD/'newPass newPass newPass'/*'
'command c/RSNMPQE  WSNMPQE  MSNMPQE/'newPass newPass newPass'/*'
'command c/RSSLSERV WSSLSERV MSSLSERV/'newPass newPass newPass'/*'
'command c/RSYSMON  WSYSMON  MSYSMON/'newPass newPass newPass'/*'
'command c/RTCPIP   WTCPIP   MTCPIP/'newPass newPass newPass'/*'
'command c/RTCPMAIN WTCPMAIN MTCPMAIN/'newPass newPass newPass'/*'
'command c/RTFTPD   WTFTPD   MTFTPD/'newPass newPass newPass'/*'
'command c/RTSAFOBJ WTSAFOBJ MTSAFOBJ/'newPass newPass newPass'/*'
'command c/RUFTD    WUFTD    MUFTD/'newPass newPass newPass'/*'
'command c/RVMKERB  WVMKERB  MVMKERB/'newPass newPass newPass'/*'
'command c/RVMNFS   WVMNFS   MVMNFS/'newPass newPass newPass'/*'
'command c/RX25IPI  WX25IPI  MX25IPI/'newPass newPass newPass'/*'
'command c/R4TCPIP  W4TCPIP  M4TCPIP/'newPass newPass newPass'/*'
'command c/RDTCVSW1 WDTCVSW1 MDTCVSW1/'newPass newPass newPass'/*'
'command c/RDTCVSW2 WDTCVSW2 MDTCVSW2/'newPass newPass newPass'/*'
'command c/RCRRLOG2 WCRRLOG2/'newPass newPass'/*'
'command c/RLDAPSRV WLDAPSRV MLDAPSRV/'newPass newPass newPass'/*'
'command c/RSNMPSUB WSNMPSUB MSNMPSUB/'newPass newPass newPass'/*'
'command c/MR READ/'MR newPass'/*'
```

## The LABEL540 EXEC

Following is the code for the EXEC that changes the system labels of a z/VM 5.4 system:

```
/*+------------------------------------------------------------------+*/
/*| EXEC: LABEL540  wrapper around CPFMTXA to LABEL and ALLOC DASD   |*/
/*|  retVal: 0 - success                                             |*/
/*|          1 - help was asked for or given                         |*/
/*|          2 - user is not sure                                    |*/
/*|          3 - DASD (minidisk) range is not valid                  |*/
/*|          4 - at least one DASD (minidisk) is reserved to MAINT   |*/
/*+------------------------------------------------------------------+*/
/* For details on how this EXEC is used, see the book:
"z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES 10 SP2"
on the Web at: http://www.redbooks.ibm.com/abstracts/SG247493.html */
/*--------------------------------------------------------------------
THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT
LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT,
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR
```

```
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES
-------------------------------------------------------------------*/

/* Construct the two character label prefix */
firstChar = 'M' /* change this for an LPAR ID other than 'M' */
labelPrefix = firstChar'V'

/* Parse arguments */
parse upper arg res spl pag w01 w02 .
if (w02 = '') then call help


/* Construct the 5 labels */
resLabel = getLabel(labelPrefix res)
splLabel = getLabel(labelPrefix spl)
pagLabel = getLabel(labelPrefix pag)
w01Label = getLabel(labelPrefix w01)
w02Label = getLabel(labelPrefix w02)

/* Ask "Are you sure?" */
say 'The volumes are:'
'CP Q' res spl pag w01 w02
say ''
say 'The system volume labels will become:'
say resLabel splLabel pagLabel w01Label w02Label
say ''
say 'ARE YOU SURE you want to relabel the DASD (y/n)?'
parse upper pull answer
ansFirstChar = substr(answer, 1, 1)
if (ansFirstChar ^= 'Y') then exit 2

/* Label the 4 volumes: RES is 123, W01 is 124, W02 is 125, SPL is 122 */
'CP TERM MORE 1 1'
'CPFMTXA 123' resLabel 'LABEL'
'CPFMTXA 124' w01Label 'LABEL'
'CPFMTXA 125' w02Label 'LABEL'
'CPFMTXA 122' splLabel 'LABEL'

/* LINK the 540PAG volume which is $PAGE$ A03, label it, DETACH it */
'CP LINK $PAGE$ A03 A03 MR'
'CPFMTXA A03' pagLabel 'LABEL'
'CP DET A03'
'CP TERM MORE 50 10'
exit

/*+------------------------------------------------------------------+*/
help: procedure expose firstChar
/*+------------------------------------------------------------------+*/
  parse source . . fn .
  say ""
  say "Synopsis:"
  say ""
  say "Relabel 5 system volumes (540RES, 540W01, ...) to" firstChar"V<xxxx>"
  say "   where <xxxx> is the 4 digit address"
  say ""
  say "Syntax is:"
  say ""
  say "   >>---LABEL540--res--spl--pag--w01--w02------------------------><"
  say ""
  say " where res, spl, pag, w01 and w02 are 4 digit virtual addresses"
  say " of the volumes that z/VM 5.2 is installed onto"
  say ""
exit 1

/*+-------------------------------------------------------------------+*/
getLabel: procedure
/*| Compose the six character label of a minidisk                     |*/
/*|  parm 1: labelPrefix - first two characters of label             |*/
/*|  parm 2: disk - vaddr of length 1, 2, 3 or 4                      |*/
/*|  return: the 6 character label                                    |*/
/*+-------------------------------------------------------------------+*/
  arg labelPrefix disk
  if (DATATYPE(disk, 'X') = 0) then
  do
    say "Error:" disk "is not a hexadecimal number"
    call help
  end
  diskLen = length(disk)
  select
  when (diskLen = 1) then /* insert 3 zeros */
    label = labelPrefix||'000'||disk
  when (diskLen = 2) then /* insert 2 zeros */
    label = labelPrefix||'00'||disk
  when (diskLen = 3) then /* insert a zero */
    label = labelPrefix||'0'||disk
  otherwise /* it must be length 4 or query would have failed */
    label = labelPrefix||disk
  end /* select */
return label /* from getLabel */
```

## The LABEL540 XEDIT macro

Following is the code for the XEDIT macro that changes all passwords in the z/VM 5.4 USER DIRECT file:

```
/*+-------------------------------------------------------------------+*/
/*| EXEC: LABEL540  wrapper around CPFMTXA to LABEL and ALLOC DASD   |*/
/*|  retVal: 0 - success                                              |*/
/*|          1 - help was asked for or given                          |*/
/*|          2 - user is not sure                                     |*/
/*|          3 - DASD (minidisk) range is not valid                   |*/
/*|          4 - at least one DASD (minidisk) is reserved to MAINT   |*/
/*+-------------------------------------------------------------------+*/
/* For details on how this EXEC is used, see the book:
"z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES 10 SP2"
on the Web at: http://www.redbooks.ibm.com/abstracts/SG247493.html */
/*-------------------------------------------------------------------
THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT
LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT,
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR
ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS
GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES
--------------------------------------------------------------------*/

firstChar = 'M' /* change this for an LPAR ID other than 'M' */
parse upper arg res spl pag w01 w02 .

parse upper arg fn ft fm '(' options ')' res spl pag w01 w02 .
if (w02 = '') then
do
  say "5 arguments required - exiting"
  exit
end
'command set stay on'
'command set num on'
'command set nulls on'
'command set serial off'
'command set cmdline bottom'
'command set curline on 3'
'command set serial off'
'command set scale off'
'command set case m i'
'command set pre off'
'command set v 1 80'
'command top'
'command c/540RES/'firstChar'V'res'/*'
'command c/540W01/'firstChar'V'w01'/*'
'command c/540W02/'firstChar'V'w02'/*'
'command c/540SPL/'firstChar'V'spl'/*'
'command c/540PAG/'firstChar'V'pag'/*'
```

## PROFILE EXEC for Linux user IDs

This section lists the code for the PROFILE EXEC that is shared among Linux user IDs from the LNXMAINT 192 disk.

```
/* PROFILE EXEC for Linux virtual servers */
'CP SET RUN ON'
'CP SET PF11 RETRIEVE FORWARD'
'CP SET PF12 RETRIEVE'
'ACC 592 C'
'SWAPGEN 300 524288' /* create a 256M VDISK disk swap space */
'SWAPGEN 301 1048576' /* create a 512M VDISK disk swap space */
'PIPE CP QUERY' userid() '| var user'
parse value user with id . dsc .
if (dsc = 'DSC') then /* user is disconnected */
  'CP IPL 100'
else  /* user is interactive -> prompt */
 do
  say 'Do you want to IPL Linux from minidisk 100? y/n'
  parse upper pull answer .
  if (answer = 'Y') then 'CP IPL 100'
 end /* else */
```

# C

# Linux source code

This section lists the Linux source code associated with this book. Following is the **`clone`** script.

## Obtaining and using the Web material

The pdf of this book is available on the Internet at:

```
http://www.redbooks.ibm.com/abstracts/sg247492.html
```

The files associated with this book are in a zipped tar file at:

```
ftp://www.redbooks.ibm.com/redbooks/SG247492
```

Download the tar file to your NFS server and use it as is described in 7.2.1, “Copying files to the controller” on page 109. After untarring the file, you will have a directory named `virt-cookbook-RH5.2`. Under that directory are the following files and directory:

| | |
|---|---|
| `README.txt` | The main README file |
| `clone-1.0-5.s390x.rpm` | The script to clone the golden image to a target user ID |
| `vm/` | A directory containing files used on z/VM |

## The clone script

Following is the clone script that installs into `/usr/sbin/` when the RPM `clone-1.0-5.s390x.rpm` is installed:

```
#!/bin/sh
#
# clone.sh is a script that clones Linux images.  It makes use of vmcp to
# relay messages to the z/VM system and configuration files to modify
# the new image once it has been cloned.
#
# The script reads in /etc/sysconfig/clone for user setting customizations.
#
# For details on how this script works see the book:
```

```
#   "z/VM and Linux on IBM System z: The Virtualization Cookbook for RHEL4"
#   on the Web at: http://www.redbooks.ibm.com/abstracts/sg247272.html
#
# -----------------------------------------------------------------------------
# THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS
# OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY
# WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY
# OR FITNESS FOR A PARTICULAR PURPOSE.
# NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY
# DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY
# OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
# NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR
# DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED
# HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES
# -----------------------------------------------------------------------------

# These MUST be lower case!
MASTER_LINK=fffe
CLONE_LINK=ffff

#+---------------------------------------------------------------------------+
function help
# give help
#+---------------------------------------------------------------------------+
 {
  echo "Usage: clone [-v] masterGuestID cloneGuestID [rootMinidisk [minidisk1
minidisk2..]]"
  echo "    Switches"
  echo "       -v Verbose output"
  echo "    Required"
  echo "       masterGuestID the z/VM user id you want to clone from"
  echo "       cloneGuestID  the z/VM user id you want to clone to"
  echo "    Optional"
  echo "       rootMinidisk the minidisk address that contains the root filesystem"
  echo "       minidisk1..n additional minidisks that should be copied"
  exit
 }

#+---------------------------------------------------------------------------+
function cp_cmd
# echo a CP command and invoke it via cp_cmd
#   Arg1-n: the z/VM command to issue
#   Return: the z/VM command's return code
#+---------------------------------------------------------------------------+
 {
  [ -n "$VERBOSE" ] && echo "Invoking CP command: $@"
  out=$(vmcp $@ 2>&1)
  rc=$?

  # Pull the z/VM error code from the output
  if [ $rc -ne 0 ] ; then
   rc=$(echo $out | grep Error | sed s/.*#//g)
   [ -z "$rc" ] && rc=1
  fi
  return $rc
 }

#+---------------------------------------------------------------------------+
function copy_key
```

```
# If the host has a id_dsa.pub file then append that to the clone's
# authorized_keys file.
#+--------------------------------------------------------------------------+
 {
  if [ -e /root/.ssh/id_dsa.pub ] ; then
    [ ! -d /mnt/clone/root/.ssh/ ] && mkdir -p /mnt/clone/root/.ssh/
    echo "# LNXINST" >> /mnt/clone/root/.ssh/authorized_keys
    cat /root/.ssh/id_dsa.pub >> /mnt/clone/root/.ssh/authorized_keys
    chmod 600 /mnt/clone/root/.ssh/authorized_keys
  fi
 }

#+--------------------------------------------------------------------------+
function abort
# Exit the script and clean up
#+--------------------------------------------------------------------------+
 {
  umount_cloned_image

  set_offline $CLONE_LINK
  set_offline $MASTER_LINK

  unlink_one $CLONE_LINK
  unlink_one $MASTER_LINK

  exit $1
 }

#+--------------------------------------------------------------------------+
function get_target_info
# Get the TCP/IP and DNS info for the Linux ID to clone to.  This function
# will check both the shared.conf file and the specific target id's conf
# file.  If values are still missing then the user will be prompted to
# supply them.
#+--------------------------------------------------------------------------+
 {
  unset HOSTNAME
  [ -f /etc/clone/shared.conf ] && . /etc/clone/shared.conf
  [ -f /etc/clone/${cloned_linux_id}.conf ] && . /etc/clone/${cloned_linux_id}.conf

  shift # drop the MasterGuestID
  shift # drop the CloneGuestID

  # If there are still command line arguments then the user must have specified DASD
  # on the command line.  Unset whatever we have in DASD (from the config files) and
  # set DASD equal to the rest of the arguments.
  [ $# -gt 0 ] && DASD="$@" && unset DASD_ROOT

  # Loop through all of the values that we require and double check that they have
  # values.  If they don't then we will prompt the user to fill them in.
  for v in HOSTNAME IPADDR DNS GATEWAY NETMASK MTU SUBCHANNELS SEARCHDNS NETTYPE DASD
  do
   if [ -z "$(eval echo \$$v)" ]; then
    [ "$PROMPT" != "y" ] && echo "Error: missing required value for $v" && exit 1
    [ -z "$first" ] && echo "Please enter $cloned_linux_id's value for: " && first=1
    echo -n "$v: "
    read in
    eval $(echo $v=\"$in\")
    export $v
    echo "$v=$in" >> /etc/clone/${cloned_linux_id}.conf
   fi
  done

  # Expand DASD ranges if they have been defined
  if [ -n "$DASD" ] ; then
   split=$(echo $DASD | tr ',' ' ')
   DASD=""
   for s in $split
   do
    out=$(echo $s | grep \-)
    rc=$?
    [ $rc -eq 0 ] && DASD=${DASD}$(seq -s" " $(echo $s | tr '-' ' ' | tr '\n' ' '))
    [ $rc -ne 0 ] && DASD=${DASD}$(echo -n "$s ")
   done
   [ -n "$DASD_ROOT" ] && DASD=$(echo $DASD | sed "s/$DASD_ROOT//")
   DASD="$DASD_ROOT $DASD"
   # Assuming that if no DASD_ROOT is specified then the first DASD device will be
   # take as root
   if [ -z "$DASD_ROOT" ] ; then
     DASD_ROOT=$(echo $DASD | awk -F" " '{print $1}')
   fi
   export DASD
  fi

  # Grab just the hostname with out any DNS suffixes from the FQDN
  target_host=$(echo $target_fqhost | awk -F. '{print $1}')
 }


#+--------------------------------------------------------------------------+
function dd_copy
# Use the dd command to copy one disk to another
#   Arg 1: Source minidisk - assumed to be online
#   Arg 2: Target minidisk - must be brought online and dasdfmt'd
#+--------------------------------------------------------------------------+
 {
  ret_val=0

  source_mdisk=$1
  target_mdisk=$2

  # Bring the source and target devices online
  set_online $source_mdisk
  set_online $target_mdisk

  target_dev_node=`cat /proc/dasd/devices | grep "$target_mdisk(ECKD)" | awk '{ print $7
}'`
  source_dev_node=`cat /proc/dasd/devices | grep "$source_mdisk(ECKD)" | awk '{ print $7
}'`

  wait_for_device /dev/$target_dev_node

  [ -n "$VERBOSE" ] && echo "Invoking Linux command: dasdfmt -p -b 4096 -y -f
/dev/$target_dev_node"
  [ -n "$VERBOSE" ] && progress="-p"
  dasdfmt $progress -b 4096 -y -f /dev/$target_dev_node
  [ $? -ne 0 ] && echo "Error: dasdfmt failed" && ret_val=1

  wait_for_device /dev/$source_dev_node
  if [ $ret_val -eq 0 ] ; then
    nblks=`cat /proc/dasd/devices | grep $target_dev_node | awk '{ print $13 }'`
    [ -n "$VERBOSE" ] && \
    echo "Invoking Linux command: dd bs=4096 count=$nblks if=/dev/$source_dev_node 
of=/dev/$target_dev_node"
    dd bs=4096 count=$nblks if=/dev/$source_dev_node of=/dev/$target_dev_node >/dev/null
    [ $? -ne 0 ] && echo "Error: dd failed" && ret_val=1
  fi

  # Put the source and target devices offline
  set_offline $target_mdisk
  set_offline $source_mdisk

  return $ret_val
 }

#+--------------------------------------------------------------------------+
function link_one
# This will link one minidisk from another user id as the target minidisk
# address on the current z/VM user id with a link mode indicated by the
# 4th argument.
#
#   Arg1: Source z/VM ID
#   Arg2: Source minidisk virtual address
#   Arg3: Target minidisk virtual address
#   Arg4: Link mode (rr/w)
#+--------------------------------------------------------------------------+
 {
  source_id=$1
  source_mdisk=$2
  target_mdisk=$3
  link_mode=$4

  cp_cmd QUERY VIRTUAL $target_mdisk
  if [ $? != 40 ]; then
    cp_cmd DETACH $target_mdisk
  fi

  cp_cmd LINK $source_id $source_mdisk $target_mdisk $link_mode $LINK_PASSWD
  if [ $? != 0 ]; then
    echo "cp_cmd link $source_id $source_mdisk $target_mdisk $link_mode failed - 
exiting"
    abort 1
  fi
  }

#+--------------------------------------------------------------------------+
function unlink_one
# This will unlink a minidisk from the current z/VM user id.
#   Arg1: The target minidisk to unlink
#+--------------------------------------------------------------------------+
 {
  cp_cmd DETACH $1
  return $?
 }

#+--------------------------------------------------------------------------+
function copy_one
# Try to use z/VM FLASHCOPY to copy one disk to another. If that fails,
#    call dd_copy() to fall back to the Linux DD command
#   Arg 1: Source minidisk
#   Arg 2: Target minidisk
#+--------------------------------------------------------------------------+
 {
  source_mdisk=$1
  target_mdisk=$2

  if [ "$COPY_METHOD" == "AUTO" -o "$COPY_METHOD" == "auto" ] ; then
    cp_cmd FLASHCOPY $source_mdisk 0 END $target_mdisk 0 END
    rc=$?
    if [ $rc -ne 0 ]; then # FLASHCOPY failed
      [ -n "$VERBOSE" ] && echo "FLASHCOPY $source_mdisk $target_mdisk failed with $rc -
using Linux dd"
    else
     return 0
    fi
  fi

  dd_copy $source_mdisk $target_mdisk
  [ $? -ne 0 ] && return 1
 }

#+--------------------------------------------------------------------------+
function copy_disks
# Call copy_one to copy each disk passed in as an argument.
#   Arg1-n: The minidisk address to copy
#+--------------------------------------------------------------------------+
 {
  [ -n "$VERBOSE" ] && echo "Copying minidisks..."
  while [ $# -gt 0 ]; do
    link_one $master_linux_id $1 $MASTER_LINK RR
    link_one $cloned_linux_id $1 $CLONE_LINK W
    copy_one $MASTER_LINK $CLONE_LINK
    [ $? -eq 0 ] && echo "$1 disk copied ..."
    unlink_one $MASTER_LINK
    unlink_one $CLONE_LINK
    shift
  done
 }

#+--------------------------------------------------------------------------+
function link_disks
# Call link_one to link each disk passed in as an argument.
#   Arg1-n: The minidisk address to link
#+--------------------------------------------------------------------------+
 {
  [ -n "$VERBOSE" ] && echo "Linking minidisks for LVM..."
  while [ $# -gt 0 ]; do
    link_one $cloned_linux_id $1 400$# W
    set_online 400$#
    [ $? -eq 0 ] && echo "$1 disk linked ..."
    shift
  done
 }

#+--------------------------------------------------------------------------+
function unlink_disks
# Call unlink_one to unlink each disk passed in as an argument.
#   Arg1-n: The minidisk address to unlink
#+--------------------------------------------------------------------------+
 {
  [ -n "$VERBOSE" ] && echo "Unlinking minidisks ..."
  while [ $# -gt 0 ]; do
    set_offline 400$#
    unlink_one 400$#
    [ $? -eq 0 ] && echo "$1 disk unlinked ..."
    shift
  done
 }

#+--------------------------------------------------------------------------+
function ask_are_you_sure
# Ask "Are you sure?" - if not, then exit
#+--------------------------------------------------------------------------+
 {
  echo ""
  echo "This will copy disks from $master_linux_id to $cloned_linux_id"
  echo "Host name will be:  $HOSTNAME"
  echo "IP address will be: $IPADDR"
  echo -n "Do you want to continue? (y/n): "
  read ans
  if [ $ans != "y" ]; then
    abort 1
  fi
 }

#+--------------------------------------------------------------------------+
function check_logged_off
# Verify the user ID exists and is logged off
#   Arg1: The user id to query if it is logged on or not
#+--------------------------------------------------------------------------+
 {
  cp_cmd QUERY $1
  case $? in
    0)  # user ID is logged on or disconnected
      echo "$1 user ID must be logged off"
      exit 2
      ;;
    3)  # user ID does not exist
      echo "$1 user ID does not exist"
      exit 3
      ;;
   45) # user ID is logged off - this is correct
      ;;
   *) # unexpected
      echo "$1 user ID must exist and be logged off"
      exit 4
  esac
 }

#+--------------------------------------------------------------------------+
function modify_cloned_image
# Modify the networking information in appropriate files under /etc
# Regenerate SSH keys in golden image's /etc/ssh/ directory and change root pw
#+--------------------------------------------------------------------------+
 {
  source_ipaddr=$(grep IPADDR $CLONE_MNT_PT/etc/sysconfig/network-scripts/ifcfg-eth0 \
        | awk -F= '{print $2}')
  source_hostname=$(grep HOSTNAME $CLONE_MNT_PT/etc/sysconfig/network \
        | awk -F= '{print $2}')
  source_host=$(echo $source_hostname| awk -F. '{print $1}')

  [ ! -d $CLONE_MNT_PT/etc ] && echo "Error: no $CLONE_MNT_PT/etc found" && abort 1

  [ -n "$VERBOSE" ] && echo "Modifying networking info under $CLONE_MNT_PT..."
  sed -i \
    -e "s/$source_ipaddr/$IPADDR/g" \
    -e "s/$source_hostname/$HOSTNAME/g" \
    -e "s/$source_host/$target_host/g" \
  $CLONE_MNT_PT/etc/hosts

  sed -i \
    -e "s/HOSTNAME=.*/HOSTNAME=$HOSTNAME/g"\
    -e "s/GATEWAY=.*/GATEWAY=$GATEWAY/g"\
  $CLONE_MNT_PT/etc/sysconfig/network

  sed -i \
    -e "s/IPADDR=.*/IPADDR=$IPADDR/g"\
    -e "s/MTU=.*/MTU=$MTU/g"\
    -e "s/NETMASK=.*/NETMASK=$NETMASK/g"\
    -e "s/SUBCHANNELS=.*/SUBCHANNELS=$SUBCHANNELS/g"\
    -e "s/NETTYPE=.*/NETTYPE=$NETTYPE/g"\
  $CLONE_MNT_PT/etc/sysconfig/network-scripts/ifcfg-eth0

  # Modify MACADDR/HWADDR if specified (optional)
  [ -n "$MACADDR" ] && sed -i -e "s/MACADDR=.*/MACADDR=$MACADDR/g" \
  $CLONE_MNT_PT/etc/sysconfig/network-scripts/ifcfg-eth0

  [ -n "$HWADDR" ] && sed -i -e "s/HWADDR=.*/HWADDR=$HWADDR/g" \
  $CLONE_MNT_PT/etc/sysconfig/network-scripts/ifcfg-eth0

  # Regenerate the SSH keys on the new clone's root filesystem
  [ -n "$VERBOSE" ] && echo "Regenerating SSH keys in $CLONE_MNT_PT/etc/ssh/ ..."
  rm -f $CLONE_MNT_PT/etc/ssh/ssh_host*
  ssh-keygen -t rsa -N "" -q -f $CLONE_MNT_PT/etc/ssh/ssh_host_rsa_key
  ssh-keygen -t dsa -N "" -q -f $CLONE_MNT_PT/etc/ssh/ssh_host_dsa_key
  ssh-keygen -t rsa1 -N "" -q -f $CLONE_MNT_PT/etc/ssh/ssh_host_key

  copy_key
 }


#+--------------------------------------------------------------------------+
function set_online
# This will set online the target minidisk.
#   Arg1 - Minidisk virtual address to set online
#+--------------------------------------------------------------------------+
 {
  local target_mdisk=$(echo $1 | tr 'A-Z' 'a-z')
  chccwdev -e 0.0.$target_mdisk >/dev/null
  rc=$?
  if [ $rc != 0 ]; then
    echo "Error: chccwdev -e 0.0.$target_mdisk failed with $rc - exiting"
    abort 1
  fi

  local target_dev_node=`cat /proc/dasd/devices | grep "$target_mdisk(ECKD)" | awk '{
print $7 }'`
  if [ "$target_dev_node" = "" ]; then
    echo "Error: can't find $target_mdisk(ECKD) in /proc/dasd/devices - exiting"
    set_offline $target_mdisk
    abort 1
  fi
 }

#+--------------------------------------------------------------------------+
function set_offline
# This will set offline the target minidisk.
#   Arg1 - Minidisk virtual address to set offline
#+--------------------------------------------------------------------------+
 {
   target_mdisk=$(echo $1 | tr 'A-Z' 'a-z')
   chccwdev -d 0.0.$target_mdisk > /dev/null 2>&1
   rc=$?
   #if [ $rc -ne 0 ]; then
   # echo "Error: chccwdev -d 0.0.$1 failed with $rc - ignoring"
   #fi

   return $rc
 }

#+--------------------------------------------------------------------------+
function mount_cloned_image
# This will mount the cloned root filesystem.  It will pair a minidisk
# address to a device file and then mount the first partition.
#   Arg1: The minidisk address to mount
#+--------------------------------------------------------------------------+
 {
  target_mdisk=$1

  target_dev_node=`cat /proc/dasd/devices | grep "$target_mdisk(ECKD)" | awk '{ print $7
}'`

  wait_for_device /dev/${target_dev_node}1
  [ $? -ne 0 ] && echo "Error: timed out waiting for /dev/${target_dev_node}1" && abort
1

  /bin/mount /dev/${target_dev_node}1 $CLONE_MNT_PT
  [ $? -ne 0 ] && echo "Error: unable to mount cloned image" && abort 1

  /bin/mount | grep /dev/${target_dev_node}1 >/dev/null 2>&1
  [ $? -ne 0 ] && echo "Error: unable to mount cloned image" && abort 1

 }

#+--------------------------------------------------------------------------+
function mount_cloned_image_lvm
# This will mount the cloned root filesystem.  It will pair a minidisk
# address to a device file and then mount the first partition.
#   Arg1: The minidisk address to mount
#+--------------------------------------------------------------------------+
 {
  target_mdisk=$1

  /bin/mount /dev/$VG_NAME/$LV_ROOT $CLONE_MNT_PT
  [ $? -ne 0 ] && echo "Error: unable to mount cloned image" && abort 1

  /bin/mount | grep $LV_ROOT >/dev/null 2>&1
  [ $? -ne 0 ] && echo "Error: unable to mount cloned image" && abort 1
 }

#+--------------------------------------------------------------------------+
function umount_cloned_image
#   Unmount the cloned root filesystem
#+--------------------------------------------------------------------------+
 {
  /bin/umount $CLONE_MNT_PT >/dev/null 2>&1

  return $?
 }

#+--------------------------------------------------------------------------+
function check_for_conf
# Check that the configuration file exists for the ID that we are cloning to.
#+--------------------------------------------------------------------------+
 {
  if [ ! -f /etc/clone/${cloned_linux_id}.conf -a "$PROMPT" != "y" ]; then
   echo "Error: /etc/clone/${cloned_linux_id}.conf not found. Exiting"
   exit
  fi
 }

#+--------------------------------------------------------------------------+
function check_for_vmcp
# Check that the vmcp module is loaded and the vmcp binary is installed.
#+--------------------------------------------------------------------------+
 {
   # Check that vmcp exists and is executable
   [ ! -x /sbin/vmcp ] && echo "Error: can't find /sbin/vmcp" && exit

   # Load the vmcp kernel module if not already loaded
   if ! /sbin/lsmod | grep vmcp > /dev/null 2>&1 ; then
     if ! /sbin/modprobe vmcp > /dev/null 2>&1 ; then
       echo "Error: unable to load module vmcp, check kernel version"
       exit
     fi
   fi

  wait_for_device /dev/vmcp
  [ $? -ne 0 ] && echo "Error: timed out waiting for /dev/vmcp" && exit
 }

#+--------------------------------------------------------------------------+
function wait_for_device
# Sleep until a certain file exists
#   Arg1: The path of the file to sleep on.
#+--------------------------------------------------------------------------+
 {
  device=$1

  for t in $(seq 1 20)
  do
    [ -e $device ] && return 0
    sleep 1
  done
  return 1
}

#+--------------------------------------------------------------------------+
function autolog
# Issue an XAUTOLOG command to bring up the new cloned image.
#+--------------------------------------------------------------------------+
 {
  cp_cmd XAUTOLOG $cloned_linux_id
  rc=$?
  if [ $? != 0 ]; then
    echo "xautolog $cloned_linux_id failed with $rc"
    return 0
  fi
  echo "Booting $cloned_linux_id"
 }

#+--------------------------------------------------------------------------+
# main()

# Only root can run this script
[ $(id -u) != "0" ] && echo "Error: you must be root" && exit

# Check if the user has defined any clone.sh configurations
[ -f /etc/sysconfig/clone ] && . /etc/sysconfig/clone

# Set defaults for clone.sh configurations
[ -z "$PROMPT" ] && PROMPT="y"
[ -z "$CLONE_MNT_PT" ] && CLONE_MNT_PT="/mnt/clone"

# If the clone mount point does not exist then we'll create it for you
[ ! -d $CLONE_MNT_PT ] && mkdir -p $CLONE_MNT_PT

# Check if -v was specified on the command line
if [ "$1" = "-v" ] ; then
  VERBOSE=1
  shift
fi

# If no command line options were provided show the help message
[ $# -eq 0 ] && help

# If one comand line option was provided show the help message
if [ $# -lt 2 ]; then
  echo "Error: incorrect number of arguments"
  help
fi

# Check that vmcp exists and the module is loaded
check_for_vmcp

master_linux_id=$1
cloned_linux_id=$2

# Check that the master and clone z/VM IDs are logged off.
check_logged_off $master_linux_id
check_logged_off $cloned_linux_id

# Check that the clone's configuration file exists
check_for_conf

# Collect information from the clone's configuration file
get_target_info $@
[ "$PROMPT" = "y" ] && ask_are_you_sure
```

```
echo "Cloning $master_linux_id to $cloned_linux_id ..."
[ -z "$DASD" ] && echo "Error: no DASD defined in /etc/clone/${cloned_linux_id}.conf" &&
exit
copy_disks $DASD

# Update the newly cloned image locally, so link, set online then mount the
# clone's root filesystem.  Then call modify_cloned_image to update
# configuration files with the proper settings.  Finally unmount,
# set offline and unlink the disk.
echo "Updating cloned image ..."
if [ -n "$VG_NAME" ]; then
  link_disks $DASD
  # FIXME wait for disks
  sleep 2
  /sbin/vgscan
  # FIXME wait for vgscan
  sleep 2
  /sbin/vgchange -a y $VG_NAME
  mount_cloned_image_lvm $CLONE_LINK
else
  link_one $cloned_linux_id $DASD_ROOT $CLONE_LINK W
  set_online $CLONE_LINK
  mount_cloned_image $CLONE_LINK
fi
modify_cloned_image
umount_cloned_image
if [ -n "$VG_NAME" ]; then
  /sbin/vgchange -a n $VG_NAME
  unlink_disks $DASD
else
  set_offline $CLONE_LINK
  unlink_one $CLONE_LINK
fi

# Autolog the clone unless AUTOLOG has been set to "n"
[ "$AUTOLOG" = "y" ] && autolog

echo "Successfully cloned $master_linux_id to $cloned_linux_id"
```

# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks publications

For information about ordering these publications, see “How to get Redbooks” on page 240. Note that some of the documents referenced here may be available in softcopy only

- *Linux on IBM eServer zSeries and S/390: Performance Toolkit for VM*, SG24-6059
- *Linux on IBM eServer zSeries and S/390: Application Development*, SG24-6807
- *Linux on IBM System z: Performance Measurement and Tuning*, SG24-6926
- *IBM Lotus Domino 6.5 for Linux on zSeries Implementation*, SG24-7021
- *Printing with Linux on zSeries Using CUPS and Samba*, REDP-3864
- *Running Linux Guest in less than CP Privilege Class G*, REDP-3870

## Other publications

- *Linux for zSeries and S/390 Device Drivers, Features, and Commands*, LNUX-1403

  `http://download.boulder.ibm.com/ibmdl/pub/software/dw/linux390/docu/l26cdd04.pdf`
- z/VM documentation - start at:

  `http://www.vm.ibm.com/library/`
  - *z/VM V5R3.0 CP Programming Services*, SC24-6084
  - *z/VM Guide for Automated Installation and Service: Version 5 Release 3.0*, GC24-6099
  - *z/VM CP Messages and Codes: Version 5 Release 3.0*, GC24-6119
  - *z/VM TCP/IP Messages and Codes: Version 5 Release 3.0*, GC24-6124
  - *The Program Directory for Performance Toolkit for VM*, GI10-0785
  - *z/VM CP Commands and Utilities Reference: Version 5 Release 3.0*, SC24-6081
  - *z/VM CP Planning and Administration: Version 5 Release 3.0*, SC24-6083
  - *z/VM Getting Started with Linux on System z9 and zSeries: Version 5 Release 3.0*, SC24-6096
  - *z/VM TCP/IP Planning and Customization: Version 5 Release 3.0*, SC24-6125
  - *z/VM Performance Toolkit Guide*, SC24-6156
  - *z/VM Performance Toolkit Reference*, SC24-6157
  - *z/VM V5R3 Guide for Automated Installation and Service*, GC24-6099
  - *VMSES/E Introduction and Reference*, GC24-6130

# Online resources

These Web sites are also relevant as further information sources:

- The Linux for zSeries and S/390 portal:

  http://linuxvm.org/

- The linux-390 list server:

  http://www2.marist.edu/htbin/wlvindex?linux-390

- Linux on System z and S/390 developerWorks:

  http://awlinux1.alphaworks.ibm.com/developerworks/linux390/index.shtml

- Information about Apache:

  http://www.samspublishing.com/articles/article.asp?p=30115&seqNum=4
  http://www.sitepoint.com/article/securing-apache-2-server-ssl
  http://www.securityfocus.com/infocus/1786

- z/VM publications:

  http://www.vm.ibm.com/pubs/

- z/VM performance tips:

  http://www.vm.ibm.com/perf/tips/

- The *z/VM Security and Integrity* paper discusses the isolation and integrity of virtual servers under z/VM

  http://www-1.ibm.com/servers/eserver/zseries/library/techpapers/pdf/gm130145.pdf

# How to get Redbooks

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks, at this Web site:

**ibm.com**/redbooks

# Help from IBM

IBM Support and downloads

**ibm.com**/support

IBM Global Services

**ibm.com**/services

# Index

## Symbols
$ALLOC$, z/VM user ID 55, 93, 120, 131, 142
/etc/auto.master 126
/etc/fstab 110, 184
/etc/inittab file 115
/etc/samba/smb.conf file 173
/etc/samba/smbpasswd file 173
/etc/yum.repos.d/ directory 126
/nfs/rhel5/ directory 115
/nfs/virt-cookbook-RH5/ directory 115
/proc/dasd/devices 110
/proc/sys/appldata/ directory 208
/usr/share/doc/ directory 173

## Numerics
3270 emulators 25, 41
520RES 35

## A
Aduva Onstage 3
Albert Einstein 3
Alt-Tab sequence 29
Apache 160
  Enabling for SSL 163
  SSL module 163
Apache resources 164
appldata_mem module 208
appldata_net_sum module 208
appldata_os module 208
application development resources 176
application development server 176
APPLMON option 207
ATTACH command 33
Attachmate Extra! 25
AUTOLOG1, z/VM user ID 212
automount 125

## B
Bill of materials 8

## C
CHANGEPW XEDIT 61
CHANGEPW XEDIT macro 62
Changes to current book xiv
chkconfig command 89, 113
choices made
  in the book 3
CHPW52 XEDIT 61
  source code 220
clone script 139
clone.sh script 172, 176
  description 139
cloning 129
  Six more virtual servers 141
command
  dd 139
  FLASHCOPY 139
  TERM MORE 151
  XAUTOLOG 139
Common UNIX Printing System (CUPS) 171
Computer Associates
  VM Direct 3
  VM Monitor 193
  VM Secure 61
Control Program 2
Controller xiv
  Configuration 109
  Copying files to 109
  Verify your changes 117
conventions
  command retrieval 10
  in the book 9
  naming backup files 10
  volume labeling 9
CP 2
cp -a command 109
CPACCESS command 43
CPFMTXA command 48
CPFORMAT EXEC 48
  using it 49
CPSYNTAX 53
CPSYNTAX command 43
CUPS 175, 239

## D
DASD
  3390 3
  ECKD 3
  labelling convention 9
dasdfmt command
  using in a for loop 181
DEFINE STORAGE command 98, 150
desktop 19
df command 128
DIRECTXA command 56, 62, 94, 121, 143
DirMaint 3
Disconnect Timeout 42
Disconnect_Timeout 43
DISKMAP command 54, 93, 120
distributions
  31-bit and 64-bit xiii
Dynamic Memory Upgrade (DMU) 2

## E
envelope files 74
ext3 file system

  - creating 184

## F
- fdasd command 181
- Features statement 42
- FILELIST command 95
- Firewall
  - configuring 161
- FLASHCOPY command 93
- font conventions xv
- FORMAT command 56
- formatting DASD fast 48
- FTP
  - site fix 80 subcommand 95
- FTP server 46

## G
- golden image xiv, 119

## H
- hardware resources 8
- HELP command 196
- HMC 28
  - CD-ROM/DVD 30
  - Load icon 35
  - Operating System Messages 47
  - Single Object Operations 29
  - SYSG 35
  - System Console (SYSC) 47
- Hummingbird Host Explorer 25
- hypervisor 2

## I
- IBM Personal Communications 25
- IBM RACF 61
- IBM Tivoli Provisioning Manager 3
- IBM z/VM Performance Toolkit 193
  - monitor subcommand 204
- INDICATE command 194
- INDICATE LOAD command 194
- INDICATE QUEUES EXP command 195
- infrastructure
  - block diagram 4
  - design 4
- initial RAMdisk 95
- INSTPLAN 32
- INSTVM EXEC 37
- Integrated 3270 Console 28, 31
- IPL CMS 37, 98, 150
- IPWIZARD command 38

## K
- kernel 94
- kickstart 147
  - clearpart option 149
  - Configuration file 148
  - Configuring controller 148
  - Using 150
  - zerombr option 149

## L
- LABEL520 EXEC
  - source code 223
- LAMP 176
- LDAP
  - Configuring client 169
- LDAP service
  - Migrating users to 166
  - starting 166
- ldapadd command 171
- ldapdelete command 169
- ldapsearch command 166, 168
- Lightweight Directory Access Protocol (LDAP) 164
- Linux
  - Configuration file 96
  - installing with kickstart 147
  - manual installation 4
  - monitoring 193, 207
  - monitoring from the kernel 207
- Linux distribution 164, 176
- Linux RMF PM Data Gatherer (rmfpms) 206
- LINUX01 131, 160
- LNXDFLT profile 92
- LNXMAINT 54, 91
  - copying files to 58
- LNXMAINT 192 92
- Load icon 35
- logical partition xi, 1
- logical volume 93
  - adding 180
- LPAR
  - requirements 8
- lsmod command 208
- lvcreate command 183
- lvdisplay command 183
- LVM
  - block diagram 182
  - physical extents 183

## M
- Mainstar Provisioning Expert 3
- MAINT 2CC 212
- MAINT CF1 212
- Map Network Drive 174
- Master image xiv
  - Add additional RPMs 111
  - Configuration 125
  - Installation 120
  - Turn off unneeded services 112, 127
- master image
  - installation 120
- MDISK statement 54
- Minidisk cache (MDC) 195
- Minidisk passwords 61
- mke2fs command 184
- modprobe command 208

- monitor stream support 208
- monitoring Linux 206–207
- mount command 124
- MySQL 176

## N

- Network File System (NFS) 85
- networking resources 9
- NFS
  - installation method 100
- NFS server
  - configuration on Linux PC 88

## O

- OpenLDAP 164
  - configuring 165
- openmotif 111
- OPTION APPLMON 131
- overlaps
  - searching for 55

## P

- PADL Software Pty Ltd 166
- paging space 48
- paging volumes 47
  - location 49
- parameter file 94, 96
  - copying 143
  - copying, for new user ID 138
- PARMFILE 96
- passwd command 173
- passwords
  - planning 12
- Performance Toolkit 198
- PERFSVM user ID 200
- philosophy of the book 3
- PHP 176
- physical extents 183
- planning 7
- Printing
  - configuration 175
- PROFILE EXEC
  - AUTOLOG1 59, 212
  - AUTOLOG1’s 94, 121, 132
  - common to Linux user IDs 97
  - LNXMAINT 57
  - PERFSVM 204
- PROFILE LNXDFLT 92
- PROFILE TCPIP 45
- Programming 71
- PTF 72
- PUT2PROD command 37
- PuTTY 20
  - customizing 20
  - larger screen 22
  - potential security breach warning 124
- pvcreate command 182
- pvdisplay command 182
- Python 176

## Q

- QUERY ALLOC MAP command 197
- QUERY CPLEVEL command 197
- QUERY DASD command 197
- QUERY NAMES command 196
- QUERY OSA command 145, 197
- QUERY PROCESSORS command 197
- QUERY STORAGE command 197
- QUERY VIRTUAL ALL command 197
- QUERY VIRTUAL command 98
- QUERY VIRTUAL OSA command 98
- QUERY VIRTUAL STORAGE command 98
- QUERY VSWITCH ACCESSLIST command 144
- QUERY XSTOR command 197
- Quick3270 25

## R

- RAMdisk 95
- Reader, z/VM’s 94
- RealVNC 101
- reboot command 117, 128
- Recommended Service Upgrade (RSU) 71
- Red Hat Enterprise Linux (RHEL) 94, 96
- Red Hat Enterprise Linux 5.2
  - parameter file 138
- Red Hat Network 153
  - Registering your system 154
  - Without Internet access 156
- Redbooks Web site 240
  - Contact us xv
- RELEASE command 43
- RHEL 5.2
  - Installation 99
  - warning messages 100
- RHEL5 user ID
  - configuration file 96
  - second half of an install 108
- rhel5.initrd 95
- rhel5.kernel 95
- RHEL52 EXEC 99
- RHEL52 user ID
  - parameter file 95
- rmfpms 206
- rmfpms command 207

## S

- Samba 171–172
  - configuration 173
  - integrating with LDAP 174
- saved segments
  - rebuilding 68
- SCSI/FCP disks 3
- service
  - autofs 126
  - httpd 161
  - iptables 162

ldap 168
smb 174
vncserver 114
services
turning off unneeded 112
SET SIGNAL command 59
SET SRM command 59
SET VSWITCH command 94, 121, 132
SET VSWITCH GRANT command 94, 132, 144
share /usr/ file system 3
SHUTDOWN command 47
SHUTDOWN REIPL command 53, 60
SIGNAL SHUTDOWN
setting Linux to halt 115
Single Object Operations 29
site fix 80, FTP subcommand 95
slapadd command 167
slappasswd command 165
smbclient command 174
smbpasswd command 173–174
smbpasswd file 174
software resources 8
SSH
Configuring keys 116, 127
key-based authentication 116
protocol 2 20
ssh-keygen command 116
SSL
with Apache 163
Standalone Program Loader 36
Storage Area Network (SAN) 3
SuSE Linux Enterprise Server xiii
SUSE Linux Enterprise Server (SLES) xiii
SWAPGEN 139
SWAPGEN EXEC 97
swapon command 110
SYSG 35
system administration
roles 12
system block diagram
LPAR and user ID based 4
SYSTEM CONFIG 43, 64, 212
customizing 42
SYSTEM CONFIG file
adding PAGE and PERM disks 52
Features statement 42
System Console (SYSC) 47
SYSTEM DTCPARMS 212
system shutdown 59
system startup 59
System status
DASD view 145
reviewing 145
System z
virtualization capabilities 2
System z10 2
System_Identifier_Default 42

## T
tar command 206
TCP/IP
z/VM configuration 38
TCP/IP resources 9
TCPCMSU 54
TCPIP Data 212
TCPMAINT 198 212
TCPMAINT 592 92, 212
TERM MORE command 100
TYPE command 96

## U
UM97510 77
UM97520 77
UM97530 77
Universal Naming Convention (UNC) 174
usability tests
on the book 5
USER DIRECT 212
USER DIRECT file 54, 92, 120, 142
APPLMON option 207
INCLUDE statement 92
profile 92
user diskmap 93, 120
USER DISKMAP file 55
User_Volume_Include 52, 131
User_Volume_List 52
useradd command 173
userid() function 96

## V
VDISK 42
Vdisk
allowing creation of 43
VDISK swap 102, 110
Velocity Software ESALPS 193
vgcreate command 183
virtual disk (VDISK) 42
virtual NIC 92
Virtual server
application development 176
File and print 171
LDAP 164
Web server 160
Virtual servers
defining user IDs for 141
virtual servers 2
virtual switch (VSWITCH) 1, 42
Virtualization 1
vmcp command 117
vmcp module 116
VMLINK command 50
VNC 99
VNC client 101
VNC server
configuring 113
volume group 183
VSWITCH 1, 44, 47, 143

## W

- What is virtualization 2
- which command 139
- worksheets
  - examples in the book 13

## X

- XAUTOLOG command 44, 59, 94, 121
- XEDIT
  - "" prefix command 92, 142
  - ? subcommand 10
  - add subcommand 43
  - BOTTOM subcommand 43
  - CHANGE subcommand 62
  - FILE subcommand 43
  - PREFIX OFF subcommand 55
  - search subcommand (/) 42
- XEDIT command 41
- XEDIT PROFILE 41
- xinetd 111
- xstartup 114

## Y

- yum
  - RPM GPG key 126
- yum command 126, 164
- yum -y command 172

## Z

- z/VM
  - adding paging volumes 47
  - backing up to tape 62
  - cons=sysg 36
  - conventions 9
  - CPFMTXA command 48
  - customizing shutdown 59
  - customizing startup 59
  - DIRECTXA command 56
  - Disconnect Timeout 42
  - DISKMAP command 54
  - envelope files 74
  - FORMAT command 56
  - FTP server 46
  - HELP command 196
  - HIT RATIO 195
  - INDICATE command 194
  - INDICATE QUEUES EXP command 195
  - INSTPLAN EXEC 32
  - INSTVM EXEC 37
  - Integrated 3270 Console 28
  - IPWIZARD command 38
  - latest PTFs 77
  - MIGRATE rate 195
  - monitoring 193
  - paging efficiency 49
  - PAGING rate 195
  - PUT2PROD command 37
  - QUERY ALLOC MAP command 197
  - QUERY CPLEVEL command 197
  - QUERY DASD command 197
  - QUERY NAMES command 196
  - QUERY OSA command 197
  - QUERY PROCESSORS command 197
  - QUERY STORAGE command 197
  - QUERY VIRTUAL ALL command 197
  - QUERY XSTOR command 197
  - reader 94
  - relabeling system volumes 63
  - RELEASE command 43
  - Restoring from tape 68
  - security issues 60
  - Servicing 71
  - SET SIGNAL command 59
  - SET VSWITCH command 132
  - Standalone Program Loader 36
  - STEAL value 195
  - strengths 1
  - system name 42
  - TCP/IP configuration 38
  - TCPIP at IPL time 44
  - TERM MORE command 100
  - TYPE command 96
  - userid() function 96
  - virtual switch 1
  - VSWITCH 1
  - XEDIT PROFILE 41
  - XSTORE rate 195
- z/VM 5.3
  - improvements 2
- z/VM 5.4
  - improvements 2
- z/VM CPACCESS command 43
- z/VM system
  - backup Copy 63
- z/VM user IDs
  - default password 38

IBM

Redbooks

# The Virtualization Cookbook for Red Hat Enterprise Linux 5.2

(0.5” spine)
0.475”<->0.873”
250 <-> 459 pages

# z/VM and Linux on IBM System z
# The Virtualization Cookbook for Red Hat Enterprise Linux 5.2

**Hands-on instructions for installing z/VM and Linux on the mainframe**

**Updated information for z/VM 5.4 and Red Hat Enterprise Linux 5.2**

**New, more versatile file system layout**

This IBM Redbooks publication describes how to roll your own Linux virtual servers on IBM System z hardware under z/VM. This edition applies to Version 5, Release 4, Modification 0 of z/VM (product number 5741-A05) and Version 5, Release 2 of Red Hat Enterprise Linux. With a z/VM and Linux infrastructure, you can reduce the time between deciding on the acquisition of new servers and then implementing them because new servers can be deployed in a matter of minutes. This powerful build and clone capability can enable you to launch new products and services without the exhaustive planning, purchasing, installing and configuring of new hardware and software that can be associated with conventional discrete hardware servers. Development groups that need test environments built and rebuilt rapidly to enable them to efficiently deliver their projects, while handling change management in the process, can also benefit from this unique advantage. The publication adopts a cookbook format that provides you with a concise, repeatable set of procedures for installing and configuring z/VM in a logical partition (LPAR), and then installing and customizing Linux. To implement these instructions, you need an IBM System z logical partition with associated resources, z/VM 5.4 media, and a Linux distribution. This book assumes that you have a general familiarity with System z technology and terminology. It does not assume an in-depth understanding of z/VM and Linux. It is written for those who want to get a quick start with z/VM and Linux on the mainframe.

**INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION**

**BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**
**ibm.com**/redbooks

SG24-7492-00 ISBN 0738431818