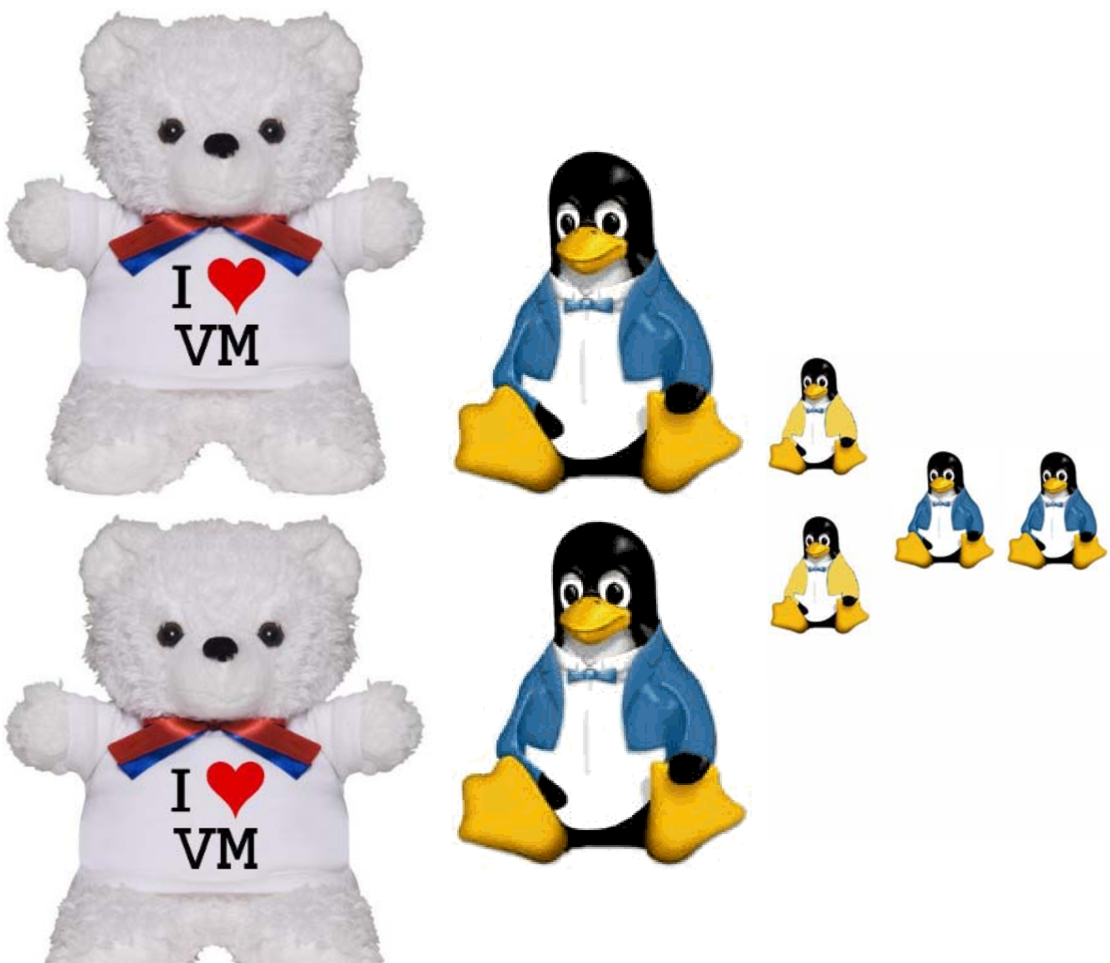
z/VM and Linux on IBM System z: The Virtualization Cookbook for z/VM 6.2 RHEL 6.2 and SLES 11 SP2

A “cookbook” for installing and customizing z/VM 6.2, RHEL 6.2 and SLES 11 SP2 on the mainframe



Michael MacIsaac
Brad Hinson
Marian Gasparovic

Contents

Preface

"*The search for truth is more precious than its possession.*"

— Albert Einstein

This book describes how to *roll your own* Linux virtual servers on IBM® System z™ hardware under z/VM®. It adopts a cookbook format that provides a concise, repeatable set of procedures for installing and configuring z/VM 6.2 into a Single System Image (SSI), then installing and customizing Linux.

You will need at least two IBM System z logical partitions (LPARs) with associated resources, z/VM 6.2 media, and either the RHEL 6.2 Linux or the SLES 11 SP2 distribution (or both).

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, and to get some virtual servers up and running in a short amount of time (days, not weeks or months).

Parts of this book

This book is comprised of the following parts:

- Part 1, "Introduction and z/VM" on page 1 introduces the entire system, describes z/VM, discusses planning, setting up ancillary machines, then installation and configuration into a two member SSI with z/VM 6.2.
- Part 2, "RHEL 6.2 Linux" on page 105 describes installing and customizing Red Hat Enterprise Linux (RHEL).
- Part 3, "SLES 11 SP2 Linux" on page 189 describes installing and customizing SuSE Linux Enterprise Server (SLES).
- Part 4, "Other topics" on page 267 - includes chapters on:
  - Live Guest Relocation (LGR) between SSI members
  - Configuring DirMaint, SMAPI and RACF
  - Monitoring z/VM and Linux
  - Miscellaneous "recipes"
  - xCAT - the eXtreme Cloud Administration Toolkit
- Part 5, "Appendices" on page 371 - includes references, cheat sheets and lists all source code in the tar file associated with this book.

Summary of changes in the July 2012 version

A relatively minor update to this book was published in July of 2012. None of the operating system version changed: z/VM 6.2, SLES 11 SP2 nor RHEL 6.2. The following additions and changes were made:

- The first words of title were changed back to *The Virtualization Cookbook*.

- Detailed steps for installing RACF into a z/VM 6.2 SSI cluster were added. This configuration describes adding the `UseRACF=yes` setting to DirMaint.
- z/VM development now recommends the use of layer 2 virtual switches (VSWITCH) exclusively. The January 2012 version added a description of using a layer 2 VSWITCH where previously only layer 3 was recommended. While this edition of the book does not remove the layer 3 examples altogether, it calls out an gives examples of using only layer 2 virtual switches.
- Steps have been added on how to attach the z/VM TCP/IP stack to the highly available virtual switch. This cannot be accomplished with the use of the **IPWIZARD** utility, but after it sets up the initial TCP/IP configuration, moving the stack over to a VSWITCH is fairly easy.
- The file `PROFILE XEDIT` is now copied to the `MAINT 19E` disk so that it need not be copied to many virtual machines' 191 disk.
- Section 6.1, "How to apply a Recommended Service Upgrade (RSU)" on page 91 has been updated for z/VM 6.2 now that the first RSU is available.
- An update to the **CPFORMAT EXEC** code has been made available. In the January 2012 version of the code, while in a non-SSI environment, `OWNER` data was still being written to CP-owned volumes. That issue has been corrected.
- The latest **SWAPGEN EXEC** has been updated in the tar file associated with this book.

Summary of changes in the January 2012 version

The changes to the January 2012 version of the book:

- The first words of title were changed to *The Cloud Computing Cookbook*.
- The z/VM sections are updated for version 6, release 2.
- The Linux sections are updated for both RHEL 6.2 and SLES 11 SP2. This is a significant change from previous versions where each book only addressed one distribution. It is assumed you will be trying both distributions, however, if you are focused on only one distribution, there are a number of "**RHEL or SLES?**" notes to address changes that would be required to use just one distribution.
- The distribution install files and other files shared by NFS are stored in `/var/nfs/` rather than `/nfs/` to be more in keeping with the Linux File Hierarchy Specification (FHS).
- Both layer 2 and layer 3 virtual switches are defined and used. In previous books, only a layer 3 virtual switch was described.
- Authorization to the virtual switches for the Linux virtual machines is granted through a **COMMAND** statement in the user directory profile. In previous books, `AUTOLOG1`'s `PROFILE EXEC` had to be modified each time a new Linux virtual server was added.
- The section on relabelling z/VM system volumes has been removed as the z/VM Single System Image (SSI) makes this task much more complex and less useful.
- There is a new chapter on Live Guest Relocation (LGR) between SSI members: Chapter 17, "z/VM Live Guest Relocation" on page 269.
- There is a new chapter on how to install and configure z/VM's DirMaint and SMAPI: Chapter 18, "Configure DirMaint, SMAPI and RACF" on page 273.
- There is a new chapter on how to install and configure the eXtreme Cloud Administration Toolkit (xCAT): Chapter 21, "xCAT" on page 357.

- There is a new section on how to install and configure **sysstat** on Linux: 19.5, "Monitor Linux with sysstat" on page 322

Conventions

The following font conventions are used in this book:

Monospace and bold	Commands entered by the user on the command line
monospace	File, directories, virtual machine and minidisk names
Monospace bold italics	Values used to test this book, such as TCP/IP addresses - they should be replaced with values correct for your enterprise.

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 $

Operating system releases used

The following releases of operating systems were used in the writing of this book:

z/VM 6.2:	GA code, December 2011
RHEL 6.2:	GA code, December 2011
SLES 11 SP2:	Release Candidate 3 code, January 2012

The team that wrote this book

This book was updated for z/VM 6.2 and RHEL 6.2 by Brad Hinson of Red Hat and Michael MacIsaac of IBM with help from Marian Gasparovic of IBM on SLES 11 SP2.

Special thanks

Special thanks goes out to the following people who performed usability tests:

Shi Lin Huang, Chen CH Ji, Song Gao, Xue Shan Zhao, Yi Chun YC Huang
IBM China

Wesley Yee
IBM Poughkeepsie

Special thanks goes out to the following person who cleaned up the REXX code:

Doug Breneman
IBM Endicott

Additional thanks goes out to the following people who contributed to this project in many different ways:

Alan Altmark, Marci Beach, Bill Bitner, Kay Blake, Chuck Brazie, Carol Everitt, Emily Kate Hugenbruch, Lori Kramer, George Madl, Mark Wagner, Mike Wilkins, Tami Zebrowski-Darrow
IBM Endicott

Thang Pham, Paul Wiggins
IBM Poughkeepsie

Dr. Manfred Gnirss, Steffen Maier, Eberhard Pasch, Hans-Joachim Picht
IBM Boeblingen

Jeremy Agee, Justin Payne
Red Hat in Raleigh NC

Mark Post
Novell

Mark Post
Novell

David Boyes
Sine Nomine

Thanks to many others in IBM Poughkeepsie and to the many who answered questions on the linux-390 and IBMVM list servers.

Comments welcome

Your comments are important to us. We want our books to be as helpful as possible. Send your comments directly to the authors:

`mikemac at us.ibm.com`

`bhinson at redhat.com`

`mariang at us.ibm.com`

Part 1

Introduction and z/VM

This part of the book starts with an introduction, discusses planning, then describes z/VM installation into a two-node SSI cluster, configuration, and servicing. It consists of the following chapters:

- Chapter 1, “Introduction to z/VM and Linux” on page 1 gives a brief introduction of the 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. Worksheets are provided for the examples, as are blank copies for your use.
- Chapter 3, “Configure a desktop machine” on page 25 describes how to set up a Windows desktop machine to access z/VM and Linux.
- Chapter 4, “Configure an NFS/FTP server” on page 33 describes how to set up a distributed server running Linux (or UNIX) to perform the initial Linux installations. Later, this machine can be retired.
- Chapter 5, “Install a z/VM SSI cluster” on page 41 shows how to install and configure z/VM 6.2 to create a two member Single System Image (SSI) cluster.
- Chapter 6, “Service z/VM” on page 89 - describes how to apply service to z/VM both in the form of Programming Temporary Fixes (PTFs) and Recommended Service Upgrades (RSUs).

1

Introduction to z/VM and Linux

"*Everything should be made as simple as possible, but not simpler.*"

— Albert Einstein

Virtualization is *still* hot in the IT industry. The IBM mainframe, z/VM and its predecessors have been doing virtualization for four different decades. Today, it can be argued that the mainframe is the most functionally rich virtualization platform. 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 virtual servers on the same System z logical partition (LPAR) under z/VM. Some customers are running many hundreds in production.

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 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, handling change management in the process can also benefit from this unique advantage.

Some of the mainframe's and z/VM's best strengths are:

- Their virtualization capabilities are more mature and robust than any other hardware and hypervisor combination.
- z/VM's virtual switch 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. There is only a relatively small number of configuration files. Properly set up, z/VM can run for months with little maintenance or administration required.

Much function has been added to z/VM since version 5.2. Following is a brief summary of the function added in the last three releases.

z/VM 6.2

z/VM 6.2, generally available in December of 2011, continues to help customers extend business value across the enterprise by integrating applications and data while providing high levels of availability, security, and operational ease. This release implements multisystem virtualization of up to four z/VM systems. This new technology will extend z/VM virtualization to a new level allowing members of the cluster to share resources and synchronize giving the appearance of being a single system image (SSI).

With the IBM z/VM Single System Image Feature (VMSSI), a running Linux virtual machine can be relocated non-disruptively from one member system to any other member, a process known as Live Guest Relocation (LGR). This provides application continuity across planned z/VM and hardware outages.

Members of a cluster are part of the same Inter-System Facility for Communications (ISFC) collection, and use ISFC channel connections to communicate. Multiple Channel to Channel devices provide a greater capability for data to flow between members. All members of a cluster share DASD for virtual machines and selected z/VM data. Sharing minidisks between members improves the integrity and performance of the system and provides a foundation for Live Guest Relocation.

Members of a z/VM SSI cluster are managed, serviced, and administered as one system. Resources including the user directory, minidisks, spool files and network devices, used by both CP and virtual machines will be shared among all members. Sharing of resources helps allow Linux guests access to the same devices and networks regardless of which member they are logged on or to where they are relocated.

Each member of a z/VM SSI cluster is able to communicate with other active members. When a z/VM system is configured as a member of a cluster, it automatically *joins* the other members during system startup. Coordination of members joining and leaving the cluster, maintaining a common view of member and resource states, and negotiating access to shared cluster resources are all accomplished in a seamless fashion. This coordination allows Linux guests to be relocated between members during planned outages. Linux guests can now be moved from one member to another during most planned outages (service upgrades) without interruption. This will allow the Linux application to have continuous run time during planned outages and therefore experience no down time.

To use the functions that define and maintain an SSI cluster, the VMSSI must be licensed and enabled. Servicing in an SSI cluster will be simplified by using a single service stream for all members. Sharing service resources will allow service to be rolled out to each member of the cluster on individual schedules, avoiding an outage for the entire cluster. This allows uninterrupted Linux guest availability as the Linux guest may be relocated to a different member prior to a planned outage.

z/VM 6.1

z/VM 6.1, available in October of 2009 is intended to be the base for all future z/VM enhancements. This release implements a new Architecture Level Set (ALS) available only on the IBM System z10 Enterprise Class server and System z10 Business Class server and future generations of System z® servers. Requiring z10 technology or later allows z/VM to take advantage of newer hardware technology for future exploitation.

Enhancements in z/VM V6.1 provide:

- Enhanced performance of virtual networking environments running heavy guest-to-guest streaming workloads
- Faster access to data when utilizing FICON Express8

- Closer integration with IBM Systems Director to eliminate the need to download agents and help simplify the installation of those agents
- Significantly better and more highly secure guest transactions when using Crypto Express3 as compared to Crypto Express2
- Guest support for IBM System Storage DS8000 Extended Address Volumes (EAVs) to help simplify storage management and relieve address constraints

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

http://www.vm.ibm.com

z/VM 5.4

z/VM 5.4, available in August of 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
- Capability to install Linux on System z from the HMC that eliminates network setup or a connection between an LPAR and the HMC
- Enhanced physical connectivity by exploiting all OSA-Express3 ports, helping service the network and reducing 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, 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 re-configuration architecture.

z/VM 5.3

z/VM 5.3 became generally available in June of 2007. Scalability was extended to allow 256GB of real memory, a total of 8TB 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 2GB for many reasons, for example in both standard I/O and Queued Direct I/O (QDIO). Now I/O can be done 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

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 z/VM's hypervisor is called Control Program (CP). When the user logs onto z/VM, the hypervisor creates a virtual machine which can run one of many different operating systems. The two operating systems that are discussed in this book are the CMS (which can be thought of as a z/VM *shell*.) and Linux. Virtual machines running Linux as guests of z/VM become *virtual servers*.

1.2 A philosophy adopted in this book

An important philosophy adopted in this book is to keep all solutions simple. Two common expressions used are “the KISS method” (Keep It Simple, Stupid) and the quote from Albert Einstein at the start of this chapter: *Everything should be made as simple as possible, but not simpler*. This book will use the latter, in an aim to use the same clear and insightful presentation.

A lot of books and papers are talking about virtualization today, but not telling you how to do it. The remainder of this book gives you the *HOWTO* that backup these marketing words.

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 a commercial Systems Management product: is not described so the reader can learn the basics.
- Directory Maintenance product versus the `USER DIRECT` file: For the most part, editing the `USER DIRECT` file and using the **`DIRECTXA`** command is recommended over a directory maintenance product such as IBM’s *DirMaint™* or CA’s *VM:Direct*. However, Chapter 18, “Configure DirMaint, SMAPI and RACF” on page 273 has now been added to show how a directory maintenance product can be used.
- Shared read-only Linux `/usr/` file system versus read-write: Some cloning solutions use an environment which shares the `/usr/` file system. This choice often makes the solution more complex, especially when adding software to the virtual servers. A read-write `/usr/` file system on the virtual servers is chosen to keep things as simple as possible.
- Conventional 3390 ECKD™ DASD versus FBA disks accessed with SCSI over FCP: 3390 DASD is described, however, section 20.4, “Add SCSI/FCP disks” on page 334 has been added.
- Cloning script or EXEC versus manual installation: Two methods of cloning are described: manually and with a Linux bash script. The manual method is described so will better learn the concepts. The Linux script is provided so you can save time.

1.4 Single System Image design

With the introduction of z/VM 6.2 in December of 2011, the architecture of Linux solutions on System z has changed dramatically. It is true that Cross Systems Extensions (CSE) allowed for a type of clustering environment for Linux on System z before z/VM 6.2, however, CSE was not widely used nor was the architecture completely enabled for clusters. z/VM 6.2, with its Single System Image (SSI) and Live Guest Relocation (LGR) functions has changed all of this. No longer is it true that a z/VM system is the most important “object” in the hierarchy. With z/VM 6.1 and earlier, the system identifier of each z/VM system was the most important. With z/VM 6.2 and later, the SSI name is the highest level identifier.

A block diagram of a four member SSI, with default volume labels, is shown in Figure 1-1. As is the recommend scenario, a four member cluster, with two members on two different CECs is shown. In such a cluster, there will be four z/VM systems and four system identifiers. However, there will be only one SSI name. In this book, a two member SSI installed onto one CEC is described.

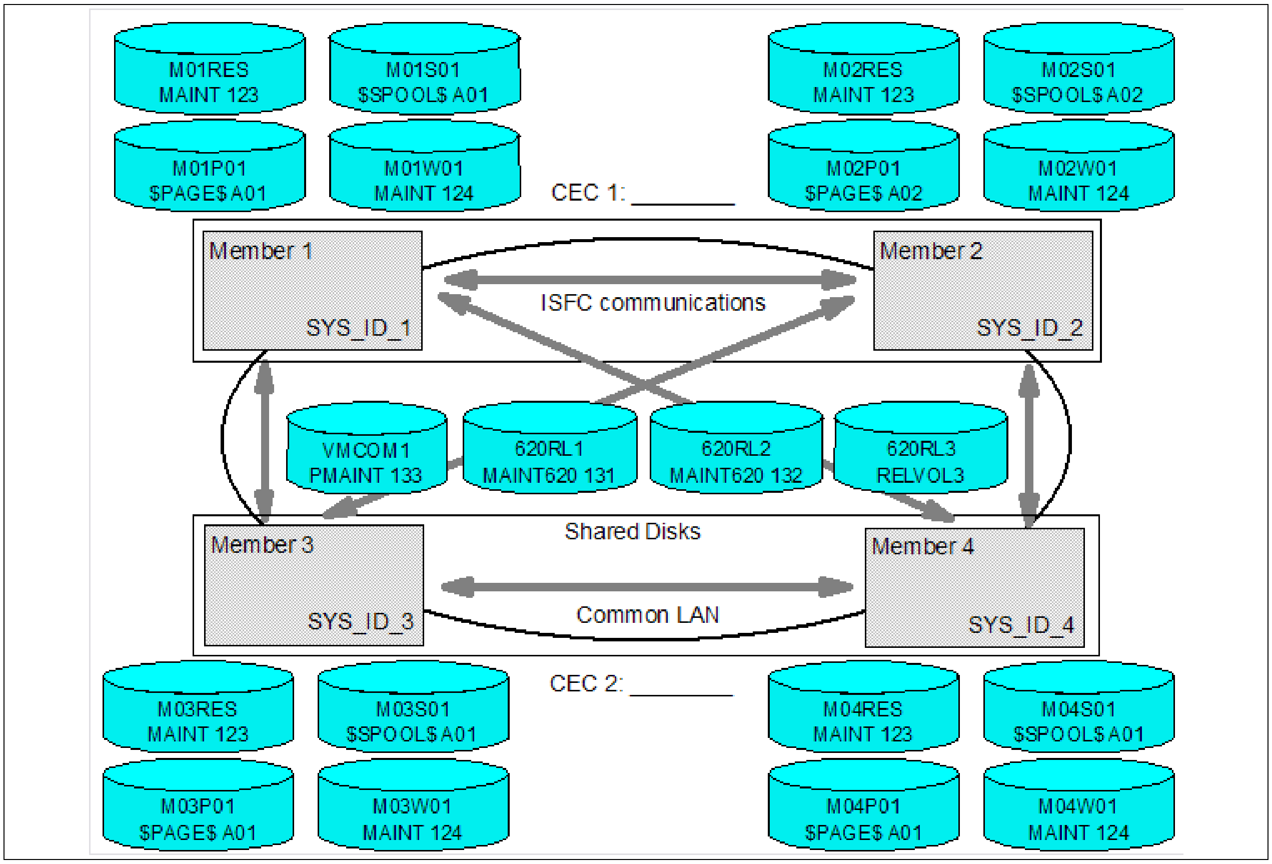


Figure 1-1 z/VM 6.2 Single System Image block diagram

1.5 Infrastructure design

To install and configure z/VM, install, configure and clone Linux, or *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 1-2 on page 6 shows a block diagram of az114 with z/VM 6.2 on two LPARs. z/VM comes with many virtual machines predefined. The most important six IDs are shown in the z/VM LPAR above the dashed line. Below the dashed line, you see the virtual machines described in this book.

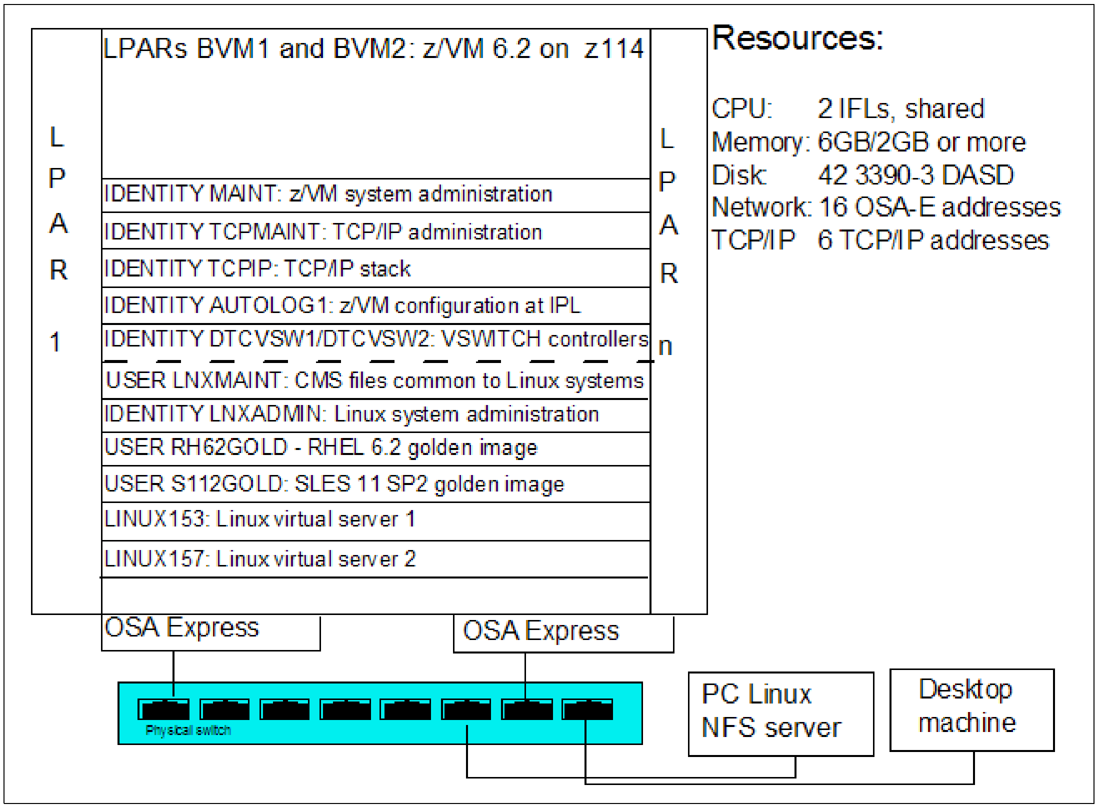


Figure 1-2 System infrastructure and z/VM virtual machines

The virtual machines that are described in this book have the following functions:

- **LNXMAINT**: A virtual machine on which to store files that will be used by both CMS and Linux
- **LNXADMIN**: The Linux system administration server that exports install trees, clones systems and performs other functions. This is an *Identity* or Multi-Configuration Virtual Machine (MCVM) that can be logged on to all SSI members at the same time
- **RH62GOLD**: A virtual machine that contains the RHEL 6.2 golden image
- **S112GOLD**: A virtual machine that contains the SLES 11 SP2 golden image
- **LINUX153 and LINUX157**: Two sample *worker* virtual machines

In addition to the two LPARs, two other machines are shown:

Linux PC NFS server	A Linux box used for the first two installations of each distribution
Desktop machine	Another Intel machine from where all of the work is done

1.6 Usability tests performed

During the yeas of writing of this book, many usability tests have been conducted. The participants had a variety of skills, but none had both solid Linux and z/VM system administration skills. By the end of two days, most participants had cloned their first Linux virtual server. You should be able to complete the steps in the book in four solid days of work, if all goes well and you work hard.

2

Planning

"*The only reason for time is so that everything doesn't happen at once.*"

— Albert Einstein

This chapter covers the planning that should be done before installing 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 resource worksheets are presented for:

- z/VM resources (other than DASD)
- DASD resources
- Linux resources
- Linux virtual machines

2.1 Planning for an SSI and LGR

There are several reasons why you might need to relocate a running virtual server: for workload rebalancing, software maintenance or hardware maintenance. Before you relocate a guest, there are architectural, disk, memory, and networking requirements you must understand. Below are some hints to help with installation of the VMSSI feature and tips to get you started relocating a Linux guest.

2.1.1 Hints and Tips

Keep in mind even if you have previous experience with installation and service of z/VM, it is important that you read the instructions for installation of z/VM 6.2 with or without the VMSSI feature. To plan and prepare for z/VM 6.2 in advance of the general availability you are encouraged to use the publications: *z/VM: Getting Started with Linux on System z* and the *z/VM: CP Planning and Administration*, chapter 25.

An SSI cluster must have direct logical links between all systems. All SSI clusters use ISFC for intra-cluster communication for LGR. ISFC uses CTC devices. For maximum throughput, when you are setting up your network, follow the Guidelines for planning your network in an SSI cluster located in the z/VM: *Getting Started with Linux on System z*, chapter 2. Faster CTC speeds increase throughput and result in shorter relocations.

Factors that can affect relocation

The following factors should be considered in planning for Linux Live Guest Relocation (LGR):

- Virtual Machine Memory: The size and use of the virtual machine's memory can affect relocation performance. Parts of the processing for relocation are proportional to the size of the virtual machine. The cost of this processing increases with larger virtual machines. Relocation performance is also impacted by the frequency and amount of memory being changed in the virtual machine.
- Matching Virtual Machine Configurations: To prepare for live guest relocation, you need to ensure that the virtual machine has a configuration that allows for it to be relocated and that a matching configuration can be set up on the destination member. For information on configuration requirements and on verifying a virtual machine's eligibility to relocate, please refer to the manual *z/VM: CP Planning and Administration*, chapter 27.
- CPU Utilization: The z/VM V6.2 Single System Image will synchronize all the members in the cluster. You must ensure that you have allocated enough system resources to account for the necessary synchronization and communication among members. After initialization, the synchronization overhead is relatively low. Communication between members does increase during negotiations for access to devices and other resources, as well as during Live Guest Relocation. For example, two independent systems that run fine at peak utilization (close to 100%), when joined in a cluster may have performance problems.

  For z/VM members that are running as a second level z/VM system, they should not be waiting for CPU more than 10% of the time. For additional details refer to the *Resource Limit Conditions* section of the manual *z/VM: CP Planning and Administration*, chapter 27.
- Paging and Other System Resources - To prepare for Live Guest Relocation, the target system must have enough system resource during and after the relocation. You will need to ensure your paging space is adequate. To be safe, there should be twice as much space available as the total virtual memory that can be defined on the system. The easiest way to check on this aspect of system resources is to issue the **`CP QUERY ALLOC PAGE`** command which will show the percent used, the slots available, and the slots in use. If you add in the size of the virtual machine(s) being relocated (a 4KB page = a 4KB slot) to the slots in use, and that brings the in use percentage over 50%, that may have an undesirable impact on system performance. Remember this query command provides only a snapshot in time.
- Real Memory - Real memory resources are important for both the source and the destination systems for relocations. You will need enough real memory (1) to hold buffers during the relocation on both systems, and (2) to accommodate the incoming guest's working set afterward on the target system. Relocation performance will also be affected by the level of overall resource constraint for both the source and destination systems.
- Linux Distributions and Live Guest Relocations - With the introduction of LGR among members of your SSI, it is increasingly important to identify the level of Linux on System z that is running within each member. The latest level of a distribution release is considered supported by the Linux Distribution Partners. The best practice for setting up VMSSI is to ensure you are running on the latest level and that your distribution is supported by your Linux Distributor. Additional information regarding distributions will be provided right here prior to the general availability of z/VM V6.2.

2.1.2 Need for ECKD DASD

If z/VM 6.2 is to be installed into an SSI, at least one ECKD volume is necessary for the Persistent Data Record (PDR).

If RACF is planned to be implemented in an SSI, the database must be configured as being shared and at least two ECKD DASD volumes are necessary. Concurrent virtual and real reserve/release must always be used for the RACF database DASD when RACF is installed in an SSI. See the *z/VM: RACF Security Server System Programmer's Guide* for more information on sharing a RACF database and *z/VM: CP Planning and Administration* for information on DASD Sharing.

2.2 Bill of materials

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

- Hardware
- Software
- Networking

2.2.1 Hardware resources

The following hardware is needed:

- One, two or four System z logical partitions (LPARs)
  - Processors or CPUs per LPAR: One IFL (or CP) minimum, two or more are recommended
  - Memory: 3 GB central/1 GB expanded minimum, 6 GB/2 GB or more recommended. This 3:1 ratio of central to expanded storage is a good starting point for relatively small systems. See the following Web site for a discussion of how to apportion memory:

    http://www.vm.ibm.com/perf/tips/storconf.html
  - DASD: 42 3390-3s or 14 3390-9s or comparable assortment
  - 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 NFS server and possibly an FTP server with at least 6 GB of disk space
- A workstation or desktop that has network access to the mainframe

2.2.2 Software resources

The following software resources are needed:

- z/VM 6.2 install media with documentation. The physical media of DVDs is described. In addition, there are now sections describing how to use electronic delivery of z/VM utilizing an FTP server, such that physical media is not needed.
- RHEL 6.2 Linux install media. If you do not have it, you can request a free 180-day evaluation copy at:

  http://www.redhat.com/z

See section 4.3, “Set up a RHEL 6.2 install tree” for details.

- SLES 11 SP2 Linux ISO image(s). For an evaluation copy, see:

  `http://www.suse.com/products/server/eval.html`

  and drill down on the **System z** link.
- An operating system for the NFS server - The same Linux distribution you will use on System z is recommended.
- The code associated with this book - on the Web at:

  `http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.tgz`
- 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
  – A VNC viewer such as RealVNC

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

2.2.3 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 card(s).

2.3 z/VM conventions

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

2.3.1 Volume labeling convention

You should have a convention for labeling DASD. Your shop 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 thoroughly, this convention guarantees that no two DASDs will have the same label. This can be an important issue especially when a z/OS® LPAR has access to the DASD.

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

With z/VM 6.2 and SSI, the rules change somewhat: there are now volumes common to the cluster and to each member LPAR. The LPAR used in this book is identified by the character **J**. 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

The letter **J** is hard-coded into the **CPFORMAT** REXX EXEC in the tar file associated with this book. This EXEC adopts the volume labelling convention. If you want a different LPAR identifier character, they can easily be changed (search for the `firstChar` variable). Following is the pertinent line of code:

```
/***************************************************************/
...
  Address COMMAND
  firstchar = 'J'
...
```

2.3.2 Backup file naming convention

It is recommend 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. Since 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.

2.3.3 The command retrieve convention

The ability to retrieve past commands is a common tool. Often it is nice to retrieve in both directions in case you “pass” the command you’re 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, though it is not defined to all virtual machines. 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* since it is right 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**.

2.4 Disk planning

There are different aspects to consider when planning how to choose and allocate disk storage. Some aspects include the following

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

DASD vs. SCSI/FCP

This book describes how to use conventional ECKD DASD and only briefly discusses FBA disks accessed over SCSI/FCP. This is not because either technology is superior, but because DASD seems to be much common than SCSI/FCP disks and it is easier to virtualize given that SCSI/FCP disks require WWPN/LUN identifiers. If you were to use SCSI/FCP disks, cloning with the **`clone.sh`** script would have to be modified to account for these identifiers, and other reasons. Sometimes a combination of these two types of disk storage is used - when that is the 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, 3390-9s or lager

Emulated 3390-3s format to about 2.3GB, while 3390-9s are three times the size or about 6.8GB. Either size will work, though 3390-3s have been recommended over 3390-9s by some performance analysts, primarily for z/VM paging space. Some large System z shops are moving to volumes larger than 3390-9s to avoid hitting the 64K limit of real device addresses (four character hexadecimal).

Disk storage per Linux image

Disk storage has the following characteristics

- This version of the book now recommends two 3390-3 DASD to create minidisks at virtual addresses `100` and `101`.
- The root file system is on `/dev/dasda1` with a recommended size of 384MB. 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 single volume group with the following characteristics:

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 4 GB out of 4.5 GB of disk space. You could choose to use other disk sizes 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 each volume or a full volume, giving each Linux about 6.8 GB or 13.6 GB of disk space, respectively.

Important: However you choose to layout the minidisks, it is important that the golden image and all target Linux virtual machines have two minidisks of the same size at virtual addresses 100 and 101. This assumption is coded into the **clone.sh** script.

2.5 Memory planning

Planning memory may be the most difficult issue with z/VM and Linux on System z, yet 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, such resource is often not be realistically available. A good rule of thumb is to allocate memory on a just enough basis for each Linux server. A good 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, here are some resources that cover this important topic:

- The Redbook *Linux on IBM System z: Performance Measurement and Tuning*, SG24-6926-01, 2008, on the Web at:

  http://www.redbooks.ibm.com/redpieces/abstracts/sg246926.html?Open

- The IBM z/VM Performance Resource pages in general, on the Web at:

  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, as this will mean more memory for the other virtual servers which remain running.

2.6 Password planning

Good passwords are critical to good security. However, requiring many different passwords generally leads to people writing them down, which clearly detracts from good security. Sometimes it is difficult to balance these two extremes.

This book considers different system administration roles:

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

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

The method of backing up z/VM data onto the Linux administration system 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 administration system, Linux virtual server virtual machines)
- The Linux virtual server end 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.

2.7 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 2.8, "Blank planning worksheets" on page 18.

2.7.1 z/VM resources used in this book

Figure 2-1 shows a block diagram with the CEC, LPARs and volume labels used in this book. The example SSI in this book is comprised of two members on a single CEC, so the bottom half of the diagram is left blank.

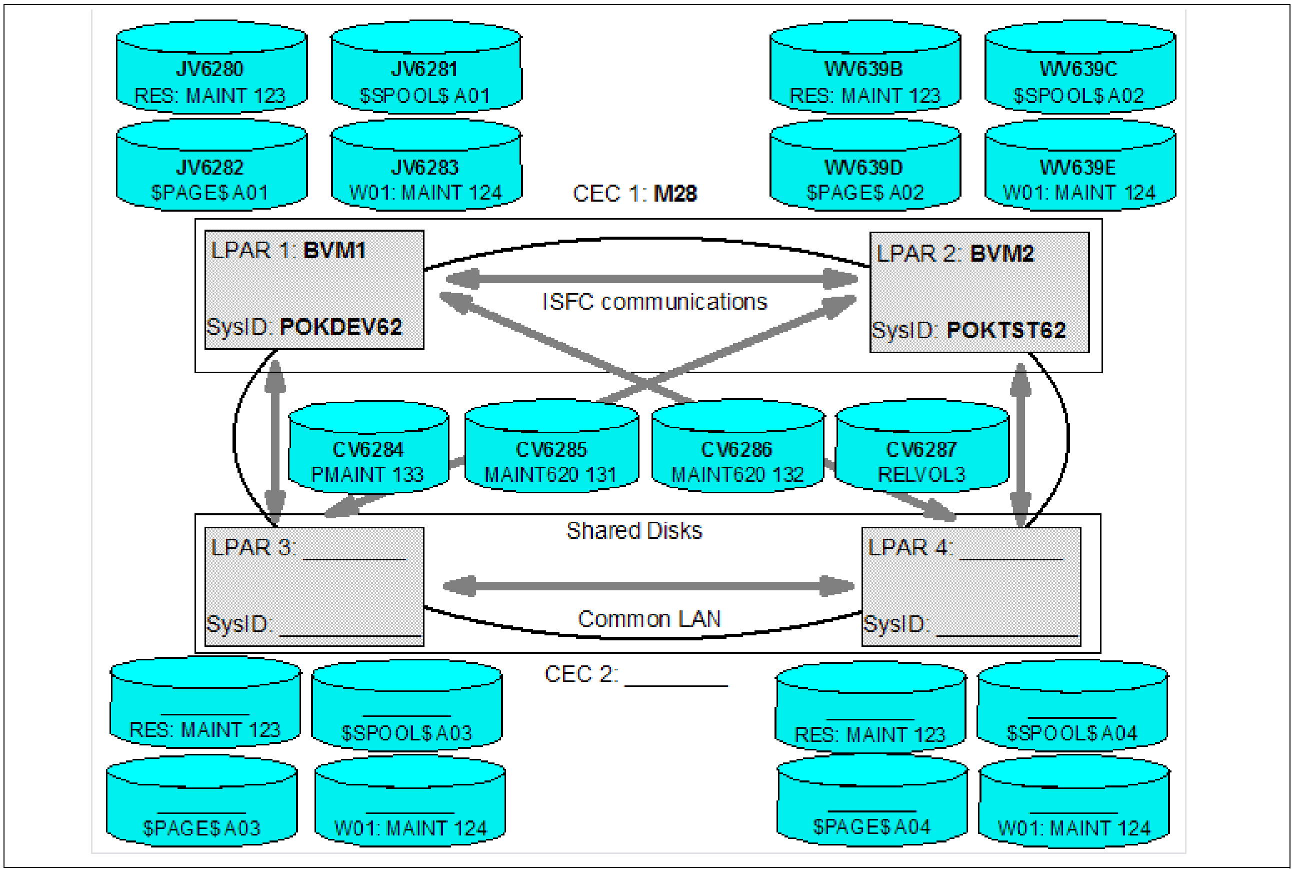


Figure 2-1 Block diagram with values used in this book

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 and networking resources worksheet

Name	Value	Comment
CEC name	MR28	Name of CEC on which LPARs are located
SSI member 1 LPAR name	BVM1	LPAR name on HMC of member 1
SSI member 1 system identifier	POKDEV62	Name to be assigned to z/VM system
SSI member 1 host name	gpok251	System hostname usually in DNS
SSI member 1 IP address	9.60.18.251	
SSI member 1 CTC addresses to SSI member 2	14A0 14A1	Usually two device addresses, up to 16
SSI member 2 LPAR name	BVM2	LPAR name on HMC of member 1
SSI member 2 system identifier	POKTST62	Name to be assigned to z/VM system
SSI member 2 host name	gpok249	System host name usually set in DNS
SSI member 2 IP address	9.60.18.249	
SSI member 2 CTC addresses to SSI member 1	14B0 14B1	Usually two device addresses, up to 16

Name	Value	Comment
TCP/IP domain name	endicott.ibm.com	System domain name usually set in DNS
TCP/IP gateway	9.60.18.129	The router to and from the local subnet
DNS server 1	9.0.2.11	Assigned by the network administrator
DNS server 2/3	9.0.3.1	Optional
OSA device name	eth0	Name of the interface to be assigned by IPWIZARD
OSA starting device number	4200	Start of OSA *triplet* for z/VM TCP/IP stack
Subnet mask	255.255.255.128	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”
MTU size	1500	Check with network administrator
Primary OSA device for layer 3 virtual switch	4203	Specify the first real device number and the next two device numbers will also be used
Secondary OSA device for layer 3 virtual switch	4300	Ideally, should be on a different CHPID/OSA card
Primary OSA device for layer 2 virtual switch	4206	
Secondary OSA device for layer 2virtual switch	4303	Ideally, should be on a different CHPID/OSA card

2.7.2 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
`61A5`	`JM61A5`	System (3390-9)	`LNXADMIN 102` on member 1
`61B2`	`JM61B2`	System (3390-9)	`LNXADMIN 103` on member 1
`6232`	`JP6232`	CP owned	Member 1 paging volume
`6233`	`JP6233`	CP owned	Member 1 paging volume
`6280`	`JV6281`	CP owned	Member 1 system residence volume
`6281`	`JV6281`	CP owned	Member 1 spool volume
`6282`	`JV6282`	CP owned	Member 1 paging volume
`6283`	`JV6283`	CP owned	Member 1 first work volume
`6284`	`CV6284`	CP owned	Common volume

Device number	Label	Type	Notes
6285	CV6285	CP owned	Common release volume 1
6286	CV6286	CP Owned	Common release volume 2
6287	CV6287	CP Owned	Common release volume 3
6288	JP6288	CP Owned	Member 1 paging volume
6289	JM6289	System (3390-3)	LNXMAINT 191, 192, LNXADMIN 101 on member 1
628A	JP628B	CP owned	Member 1 paging volume
628B	WP628A	CP owned	Member 2 paging volume
628C	JM628C	System (3390-3)	RH62GOLD 100
628D	JM628C	System (3390-3)	RH62GOLD 101
628E	JM628E	System (3390-3)	S112GOLD 100
628F	JM628F	System (3390-3)	S112GOLD 101
6290	JM6290	System (3390-3)	LNXADMIN 100 on member 1
6293	JM6293	System (3390-3)	LNXADMIN 100 on member 2
6294	JM6294	System (3390-3)	LINUX153 100
6327	JM6327	System (3390-3)	LINUX153 101
6328	JM6328	System (3390-3)	LINUX157 100
633B	WP633B	CP owned	Member 2 paging volume
633C	JP633C	CP owned	Member 2 paging volume
633E	JP633E	CP owned	Member 2 paging volume
6339	JM6339	System (3390-3)	LINUX157 101
633A	JM633A	System (3390-3)	For adding a logical volumes
639B	WV639B	CP owned	Member 2 residence volume
639C	WV639C	CP owned	Member 2 spool volume
639D	WV639D	CP owned	Member 2 paging volume
639E	WV639E	CP owned	Member 2 first work volume
63AA	JM63AA	System (3390-9)	For extending a logical volumes
D942	JMD942	System (3390-9)	Disk 1 for shared RACF database
D943	JMD943	System (3390-9)	Disk 2 for shared RACF database

2.7.3 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	9.60.18.240	
User/password	`root/lnx4vm`	
NFS-exported install directory	`/var/nfs/rhel62/` and `/var/nfs/sles11sp2/`	Directory with DVD 1 of each distribution

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 root password	lnx4vm	
TCP/IP gateway	9.60.18.129	Obtain from network administrator
Subnet mask	255.255.255.128	Obtain from network administrator
DNS server	9.0.2.11, 9.0.3.1	Obtain from network administrator
VNC installation password	12345678	Must be 8 characters

2.7.4 Host names used in this book

Table 2-6 lists the host names and associated IP addresses used in the examples in this book.

Table 2-6 Hosts used in this book

Host name	IP address	virtual machine/ LPAR	Notes
gpok144.endicott.ibm.com	9.60.18.144	`RH62GOLD`	RHEL 6.2 “golden image”
gpok145.endicott.ibm.com	9.60.18.145	`S112GOLD`	SLES 11 SP2 “golden image”
gpok151.endicott.ibm.com	9.60.18.151	`LNXADMIN`	Linux sysadmin server on member 1
gpok152.endicott.ibm.com	9.60.18.152	`LNXADMIN`	Linux sysadmin server on member 2
gpok153.endicott.ibm.com	9.60.18.153	`LINUX153`	Virtual machine to deploy systems to
gpok157.endicott.ibm.com	9.60.18.157	`LINUX157`	Virtual machine to deploy systems to
gpok249.endicott.ibm.com	9.60.18.249	`LPAR BVM1`	z/VM 6.2 SSI member 1
gpok251.endicott.ibm.com	9.60.18.251	`LPAR BVM2`	z/VM 6.2 SSI member 2

2.8 Blank planning worksheets

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

2.8.1 SSI block diagram

Use the block diagram shown in Figure 2-2 to label the CECs, LPARs, and volume labels you will use for your SSI cluster.

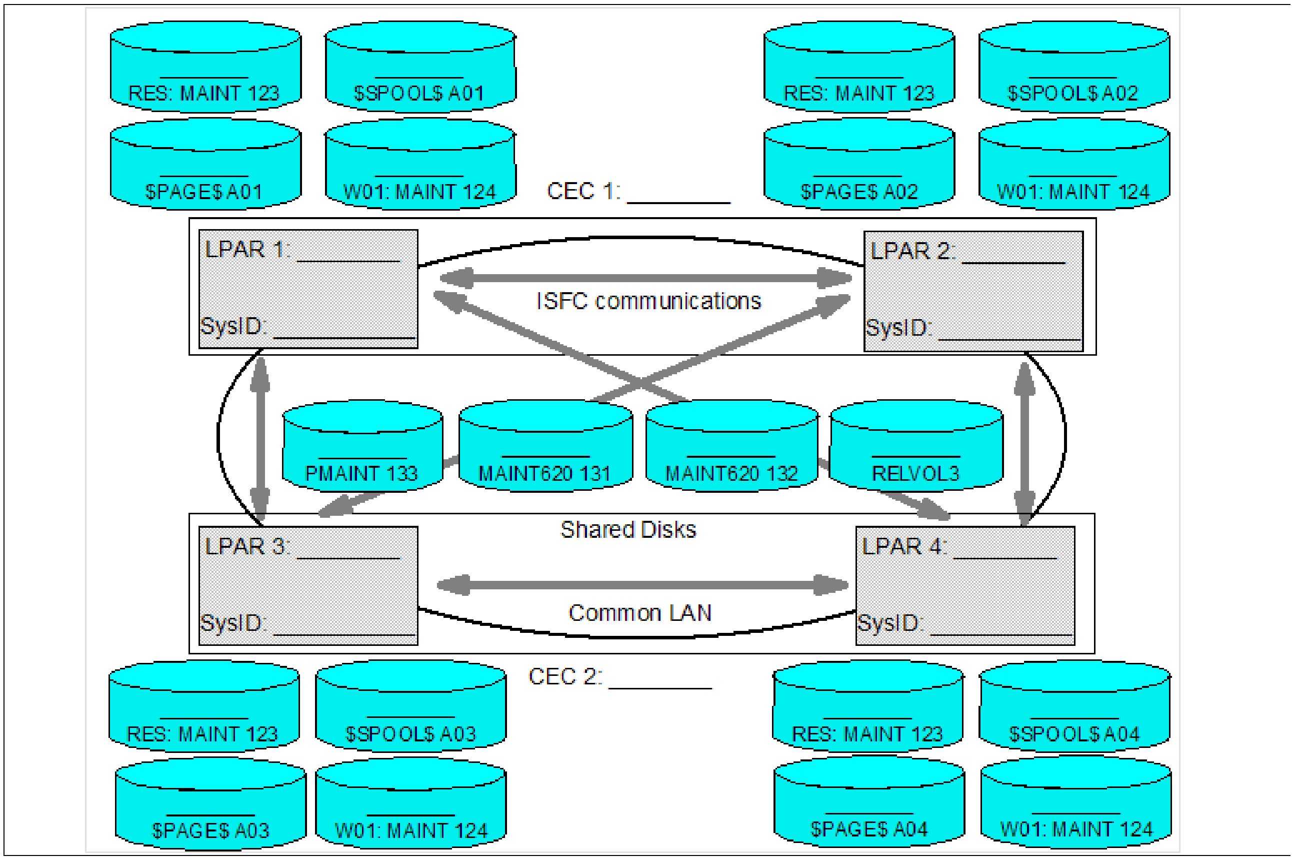


Figure 2-2 Blank SSI block diagram

2.8.2 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 and networking resources blank worksheet

Name	Value	Comment
CEC name		
SSI member 1 LPAR name		
SSI member 2 LPAR name		
SSI member 1 system identifier		
SSI member 1 host name		
SSI member 1 IP address		

Name	Value	Comment
SSI member 1 CTC addresses to SSI member 2		
SSI member 2 system identifier		
SSI member 2 host name		
SSI member 2 IP address		
SSI member 2 CTC addresses to SSI member 1		
TCP/IP domain name		
TCP/IP gateway		
DNS server 1		
DNS server 2/3		
OSA device name		
OSA starting device number		
TCP/IP address		
Subnet mask		
OSA device type		
Network type		
Port name (optional)		
Router type		
MTU size		
Primary OSA device for layer 3 virtual switch		
Secondary OSA device for layer 3 virtual switch		
Primary OSA device for layer 2 virtual switch		
Secondary OSA device for layer 2 virtual switch		

2.8.3 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

Device number	Label	Type	Notes

2.8.4 Linux resources worksheet

Use the worksheet in Table 2-9 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 2-10 to document your System z Linux resources.

Table 2-10 Linux resources blank worksheet

Name	Value	Comment
Linux install password		
Linux root password		
Linux TCP/IP gateway		
Linux TCP/IP broadcast		
Linux DNS server		
VNC Installation password		

2.8.5 Host names worksheet

Use the worksheet in Table 2-11 to document the host names and associated IP addresses and virtual machines that you will use.

Table 2-11 Host names blank worksheet

Host name	IP address	virtual machine/ LPAR	Notes

3

Configure a desktop machine

"*Technological progress is like an axe in the hands of a pathological criminal.*"

— Albert Einstein

3.1 Windows desktops

Many people use Microsoft® Windows as a desktop operating system. This chapter addresses the following tools that 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

3.1.1 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 Windows desktop, you will need a good SSH client. PuTTY is perhaps the most commonly used. You can download PuTTY from the Web at:

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

To download from this page, click on the **putty.exe** link for your architecture. Save the file in a directory path such as `C:\WINNT`. PuTTY is a stand-alone executable (no installation 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 should open. If you spend a few minutes to configure PuTTY it may pay off in time savings. The examples shown below are using PuTTY Release 0.60.

- In the *PuTTY Configuration* window, in the left Category panel, click **Session**.

- Under the *Connection Type* heading on the top right, click the **SSH** radio button as shown in Figure 3-1. This specifies to use the SSH protocol.

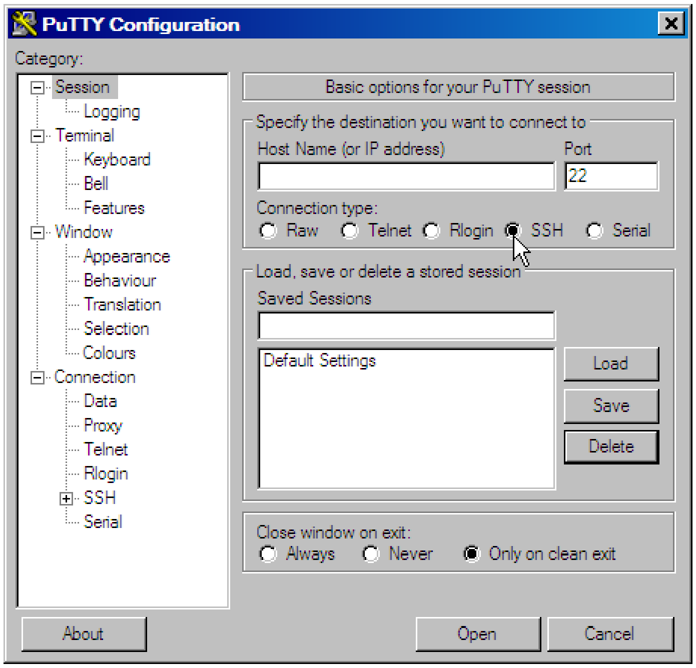


Figure 3-1 PuTTY Configuration window

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

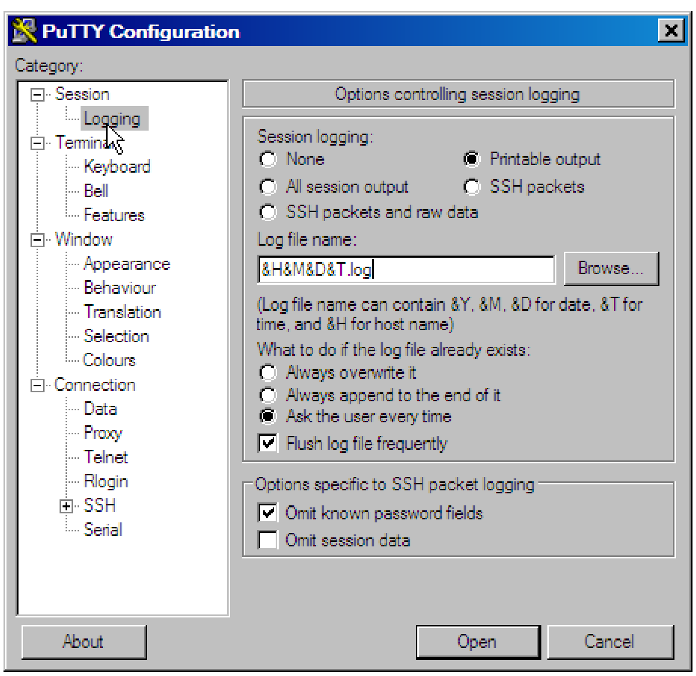


Figure 3-2 Setting logging

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

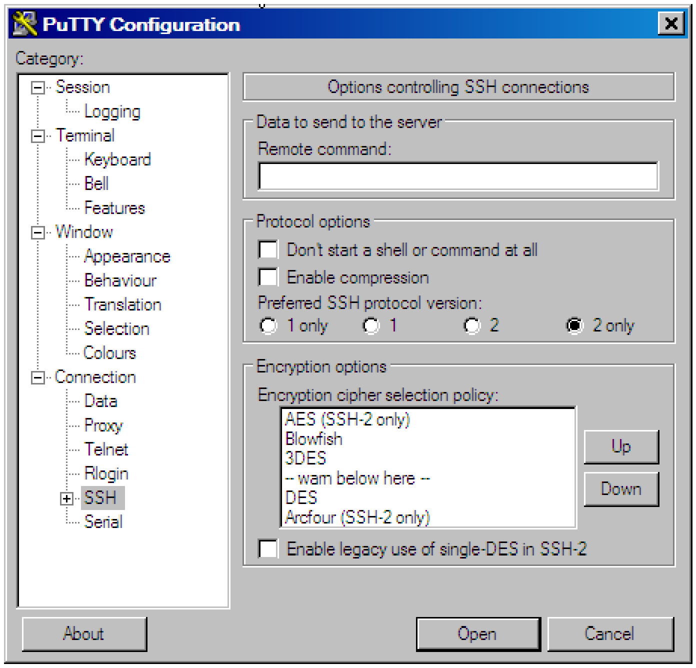


Figure 3-3 Setting SSH Protocol 2

- ► In the left Category panel, click **Terminal** as shown in Figure 3-4.
- ► Select the **Use background colour to erase screen** check box, which results in a better job of painting the screen for applications that uses block graphics.

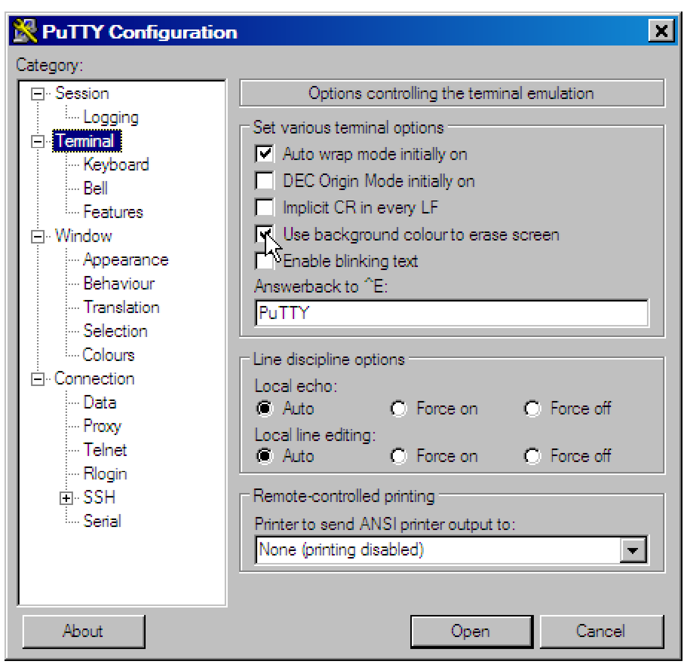


Figure 3-4 Customizing PuTTY SSH settings (Part 1 of 4)

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

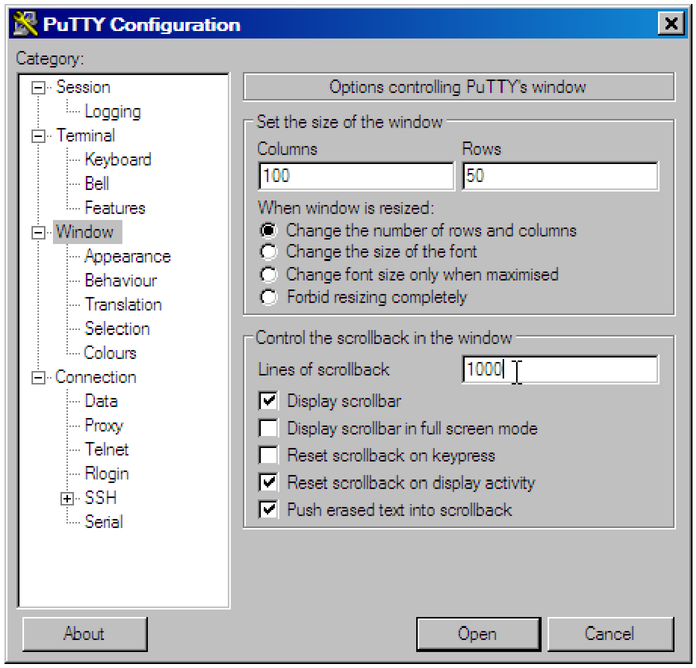


Figure 3-5 Setting Window and scroll back size

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

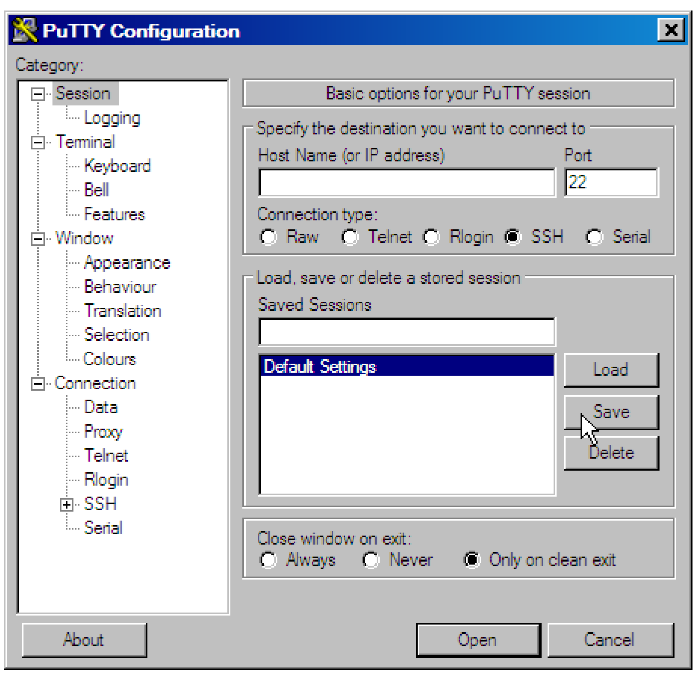


Figure 3-6 Saving new default settings

Save sessions

To save sessions perform the following steps. In this example a session for `LINUX00` is saved:

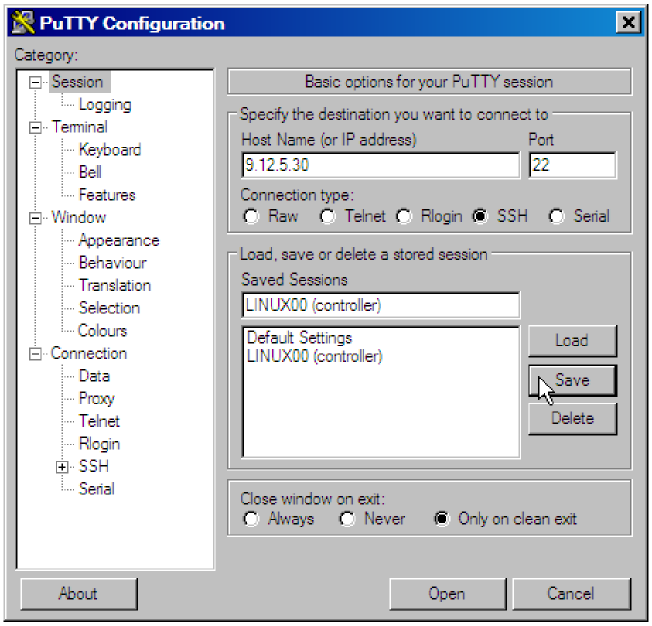


Figure 3-7 Customizing PuTTY window settings (Part 4 of 4)

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

- In the *Host Name (or IP address)* field, enter the TCP/IP address (or DNS name).
- Under *Saved Sessions* text area, choose a name that you will remember. In this example, the name `LINUX00 (controller)` is used.
- 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.

3.1.2 Set up a VNC client

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

If you have a Windows desktop, the VNC client from RealVNC is a popular choice. 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`

Click the **Download and Use** button. 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 of this book, 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* panel as shown in Figure 3-8.

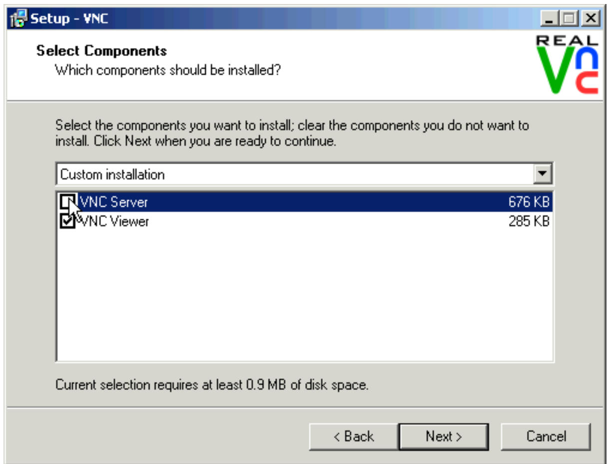


Figure 3-8 RealVNC Select Components panel

- Complete the screens and the installation process should go quickly.

> **Attention:** It seems there is no RealVNC viewer for Windows Vista nor Windows 7. The tool TightVNC might be an option for those operating systems. See:
>
> `http://www.tightvnc.com/download.php`
>
> The following text was found:
>
> "TightVNC 2.0.4 supports all client and server versions of Microsoft Windows starting at Windows 2000, up to Windows 7. "

3.1.3 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
- Others ...

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 the **Pause** key in the upper right.
- Set a larger screen. 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 screen 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 to define and save a session for each system to which you may connect, as was described for PuTTY. Then you can usually double-click the saved connection to quickly access a new 3270 session.

3.2 Linux desktops

If you are using a Linux desktop should have, or at least have access to these three tools:

- An SSH client, named **`ssh`**. It is part of the `openssh` package.
- A VNC client, named **`vncviewer`**. It is part of the `tightvnc` package.
- A 3270 emulator, named x3270. It is part of the `x3270` package.

4

Configure an NFS/FTP server

“*Anyone who has never made a mistake has never tried anything new.*”

— Albert Einstein

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, a PC Linux system is recommended. This server supplies both the RHEL 6.2 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 box (Sun™ Solaris™, Hewlett Packard HP-UX, IBM AIX® or other). You can also choose to use a Windows workstation with FTP or HTTP, if you absolutely must. Often, more problems are encountered when using a Windows workstation than a Linux or Unix workstation to serve the RHEL 6.2 install tree, so this choice is not recommended.

The steps in this chapter explain how to configure a Linux *box* as the NFS server. In addition to being an NFS server for Linux installation, this system can also be used as an FTP server for z/VM installation.

The following tasks will set up a Linux server:

- “Install Linux on a distributed server” on page 33
- “Download files associated with this book” on page 34
- “Set up a RHEL 6.2 install tree” on page 34
- “Set up a SLES 11 SP2 install tree” on page 37
- “Enable the NFS server” on page 39
- “Configure an FTP server” on page 40

4.1 Install Linux on a distributed server

If you don’t have a Linux server then you must get access to one on the network and install Linux onto it. Describing these steps is outside the scope of this book. 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) architecture. In this chapter, a PC running RHEL 6.2 is used.

4.2 Download 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 on the Web at:

http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.tgz

Perform the following steps:

- The tar file `CKB-VM62.tgz` is only about 34 KB. Download the file and untar it. The following example shows this being done from a newly created directory `/nfs/`:

```
# mkdir /var/nfs
# cd /var/nfs
... download or copy the file CKB-VM62.tgz to /nfs/ ...
# tar xzvf CKB-VM62.tgz
CKB-VM62/
CKB-VM62/rhel62/
CKB-VM62/rhel62/clone-1.0-11.s390x.rpm
CKB-VM62/sles11sp2/
CKB-VM62/sles11sp2/clone.sh
CKB-VM62/disclaimer.txt
CKB-VM62/vm/
CKB-VM62/vm/lnxmaint/
CKB-VM62/vm/lnxmaint/rhel62.exec
CKB-VM62/vm/lnxmaint/sample.parm-rh6
CKB-VM62/vm/lnxmaint/sample.conf-rh6
CKB-VM62/vm/lnxmaint/sample.parm-s11
CKB-VM62/vm/lnxmaint/profile.exec
CKB-VM62/vm/lnxmaint/sles11s2.exec
CKB-VM62/vm/lnxmaint/swapgen.exec
CKB-VM62/vm/maint/
CKB-VM62/vm/maint/callsm1.exec
CKB-VM62/vm/maint/ssicmd.exec
CKB-VM62/vm/maint/chpw620.xedit
CKB-VM62/vm/maint/ssishutd.exec
CKB-VM62/vm/maint/cpformat.exec
CKB-VM62/README.txt
```

- Review the file `README.txt` - it briefly describes each of the directories and files

You now have downloaded and uncompressed the files associated with this book.

4.3 Set up a RHEL 6.2 install tree

RHEL or SLES?: If you are working only with SLES 11 SP2, you can skip this section.

You must have a valid Red Hat entitlement for Linux on IBM System z to access the Red Hat Enterprise Linux 6 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 **SIGN UP** 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 with instructions on 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.

4.3.1 Copy from physical DVD

RHEL 6.2 is distributed on physical CDs or files that are ISO images of CDs. RHEL 6.2 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.

In the event that you have a physical DVD, but not an ISO image, it is recommended that you create an ISO image. You could skip creating the ISO image and copy the data directly from the DVD to the install tree, but creating the ISO image is recommended so you have a reference file.

The device file named `/dev/cdrom` is often associated with the CD/DVD drive, however, 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 RHEL 6.2 on System z physical DVD disc:

- Plug the DVD in CD/DVD drive. The disc should be automatically mounted over the directory `/dev/cdrom/` (if not try `/dev/dvdrom/` or `/dev/dvdrw/`).
- Make a directory to store the ISO image in and change into it:

```
# mkdir -p /var/nfs/rhel62
# cd /var/nfs/rhel62
```

- Use the **dd** command to copy the contents of the disc to an ISO image:

```
# dd if=/dev/cdrom of=RHEL6.2-20111117.0-Server-s390x-DVD1.iso
# umount /mnt/cdrom
```

You should now have an ISO image of the DVD.

4.3.2 Verify the RHEL 6.2 ISO image

An important early step is to verify the integrity of ISO image. This is done by comparing a checksum value which was calculated when the DVD was created against a checksum value calculated against your ISO image. If the two checksum values differ then there was an error somewhere in the copying process.

The **sha1sum** command allows you to compare checksum files. The checksum value for RHEL 6.2 for the s390x architecture is as follows:

```
# cat SHA1SUM
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA256

325bad9176fa812b25b6c51ee9894b06b9275c5b  RHEL6.2-20111117.0-Server-s390x-DVD1.iso
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.4 (GNU/Linux)
```

```
iQIVAwUBTsVEIRmeL5H9Qx1RAQheXg/7BYhHud6EPorwsKBOml53PRp4Gfw2Rbnp
M1wTpGLu/gILb3BxBAXbBqWfN+9Z7JRAHGlkZDgcGOO6NEWj+fJOoebp6Jj5wCYa
qajqJucNf8dkMkcOOZGVJVczYU/OaRufeqHL/OA+JuUGLwOFO4dbQUvfd9aFfT7f
IhLsy7o3RAG6ae+4URiCbICpwT+uMLUpjjNsPzDS7tGVr8P/TFfAT6X/tsnVuRbA
ExNYaqSM9hrIegnd3VkJxxIrR9ukOZ6azOyTxZEXs/OOZWcisNaV8lM5Hg2UM4mo
4388PACcjhgfObrq9VOzddmiIFch3KBYL3PBtgjEmu3qGUhiHtb2GSU46/1n5QAZ
6xTA18DP3WgZ2ONKk+PS+iMVZaqOl6J9g85oldeoHEVUirQCO3S3OAPTPyA7iT7u
BW9HXczwlCLmU2I9Q1+yj+/6wOclFYnhO8VnlfqrxO+Z4mHUcPQ87sVcZ2zgd34l
5Ul+RkNqYI3JOgZFdHI/zyIViuEsEtlZcBh4GcRiQkaN8dRUqAnhsuhsanOSvMp1
xksktBj4zfX5lPtuAAmI4EivyKOpeVC1pu7jjMdBOsVnOmDp3YXmsvwtA4O1j4jq
BsOSOdc2xS5CYiKAHxYkeaC2DOuNc5Ad+oQblB7Jx1CZ7nGCUq5iWtCqK+J+CchS
Ql57+OGSBcQ=
=OF8S
-----END PGP SIGNATURE-----
```


Run the **sha1sum** command against the MD5SUM file:

```
# sha1sum -c SHA1SUM
RHEL6.2-20111117.0-Server-s390x-DVD1.iso: OK
```

If the ISO image does not report OK, it must be downloaded or copied again until it does.

4.3.3 Mount the RHEL 6.2 ISO image

Now that you have an ISO image, it should be mounted to get access to its contents. Perform the following steps:

- Make a directory over which the ISO image will be mounted. In this example, it

```
# cd /var/rhel62
# mkdir dvd1
```

- Mount the ISO image using a loopback device:

```
# mount -o loop RHEL6.2-20111117.0-Server-s390x-DVD1.iso dvd1
```

- List the contents of the mounted ISO image:

```
# ls dvd1
.discinfo                RELEASE-NOTES-hi-IN.html  RELEASE-NOTES-te-IN.html
.treeinfo                RELEASE-NOTES-it-IT.html  RELEASE-NOTES-zh-CN.html
...
```

You now have the contents of the RHEL 6.2 DVD accessible.

> **Important:** With RHEL 5, building a new repository for **yum** was necessary. With RHEL 6.2, this step should not be necessary as the repository on the ISO image is correct. However, this short section from the previous book is left here for reference, should you need to build a repository.
>
> For the **yum** command to work, a common metadata repository must be built with 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/Server/
> # mv repodata/ repodata.orig
> # createrepo -g repodata.orig/comps-rhel5-server-core.xml .
> ...
> ```
>
> The newly created repodata/ directory contains the correct common metadata.

4.4 Set up a SLES 11 SP2 install tree

RHEL or SLES: If you are working only with RHEL 6.2, you can skip this section.

You may have a licensed version of SLES 11 SP2 on physical DVDs or you may choose to try an evaluation copy. There is an evaluation copy on the Web starting at the following URL:

http://www.suse.com/products/server/eval.html

Follow the link named **System z** and create an account to download the ISO images.

4.4.1 SLES 11 SP2 DVD ISO image file

Following are the DVD ISO images for SLES 11 SP2 that were available at the time this book was written (Release Candidate 3):

Table 4-1 SLES 11 SP2 DVDs

DVD number	File name	File size in bytes
1	SLES-11-SP2-DVD-s390x-RC3-DVD1.iso	3197044736
2	SLES-11-SP2-DVD-s390x-RC3-DVD2.iso	4935784448

The second DVD is not described in this book because it is not needed. If you are starting with a soft copy of the DVD 1 ISO image, **copy it** to the `/var/nfs/sles11sp2/` directory now.

4.4.2 Start from physical DVD

If you are starting with physical DVDs, you must first convert them to ISO images. This can be accomplished using the Linux **dd** command which basically does a byte-for-byte copy of the DVD contents.

Perform these steps only if you are starting with a SLES 11 SP2 distribution for System z on physical DVD discs:

- Plug the first DVD in CD/DVD drive. The disc should be automatically mounted over the directory `/dev/cdrom/` (if not try `/dev/dvdrom/` or `/dev/dvdrw/`).
- Make a directory to store the ISO image and change into it:

```
# mkdir -p /var/nfs/sles11sp2
# cd /var/nfs/sles11sp2
```

- Use the **dd** command with the **if** (input file) and **of** (output file) parameters to copy the contents of the disc to an ISO image:

```
# dd if=/dev/cdrom of=SLES-11-SP2-DVD-s390x-RC3-DVD1.iso
```

- Unmount the DVD disc:

```
# umount /mnt/cdrom
```

- Repeat the process for DVDs 2 and 3 using the target name `-DVD2.iso` and `-DVD3.iso`.

You should now have an ISO images of the physical DVDs.

4.4.3 Verifying the ISO images

You should first verify the integrity of the ISO images. This is done using a file of checksum values and ISO file names.The checksums were calculated from the contents of the DVD. After downloading or **dd**ing the ISO images, the checksums are calculated again and compared against the original values using the **md5sum** command and the checksum files.

Following is an example of using the **md5sum** command against an `MD5SUM` file. When you obtain the DVDs, be they physical discs or ISO images, you should also obtain an `MD5SUM` file.

SLES 11 SP2 MD5SUM values

```
# cat MD5SUM
a63698e2fb8959ff0ae7aef4b2a3003c  SLES-11-SP2-DVD-s390x-RC3-DVD1.iso
0d7d7b9ecad0b2191f86107e08b09015  SLES-11-SP2-DVD-s390x-RC3-DVD2.iso
89796b724fb277a4666fdb391ab80cdb  SLES-11-SP2-DVD-s390x-RC3-DVD3.iso
```

Use the **md5sum -c** command to verify the integrity of the ISO images. All should report OK. Following is an example:

```
# md5sum -c MD5SUM
SLES-11-SP2-DVD-s390x-RC3-DVD1.iso: OK
SLES-11-SP2-DVD-s390x-RC3-DVD2.iso: OK
md5sum: SLES-11-SP2-DVD-s390x-RC3-DVD3.iso: No such file or directory
SLES-11-SP2-DVD-s390x-RC3-DVD3.iso: FAILED open or read
```

Any ISO images that do not report `OK` must be downloaded or copied again (DVD 3 was not downloaded for this book).

4.4.4 Configure the SLES 11 SP2 install server

With SLES 9 and earlier, creating an install tree became complicated with Service Packs. SLES 10 brought the ability to *automount* the DVDs. There are different possible methods of making the SLES 11 SP2 distribution available:

- By copying the contents of all DVDs into one directory
- By loop mounting each ISO image to different directory
- By using the ISO images directly

The last option is the easiest. The installation process will access the ISO images directly and mount them automatically.

The initial RAMdisk (`initrd`) and the kernel (`vmrdr.ikr`), will have to be copied to z/VM for the first Linux install. These files are on the first DVD in the `/boot/s390x/` directory. To allow access to these files, the first DVD must be loopback mounted using the **-o loop** flag to the **mount** command. First create a directory, `dvd1/`, as a mount point:

```
# cd /var/nfs/sles11sp2
# mkdir dvd1
# mount -o loop SLES-11-SP1-DVD-s390x-GM-DVD1.iso dvd1
```

The next step is to enable the NFS server.

4.5 Enable 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 by setting the /etc/exports configuration file. Export the directory /var/nfs/rhel62/ (if you are using RHEL), /var/nfs/sles11sp2/ (if you are using SLES) and /var/nfs/CKB-VM62/ for access to the files associated with this book:

```
# cd /etc
# cp exports exports.orig
# vi exports
/var/nfs/rhel62          *(ro,sync)
/var/nfs/sles11sp2       *(ro,sync)
/var/nfs/CKB-VM62        *(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 (with the "*") for security reasons. Type **man exports** for more details.

- Set the NFS server to start with the **chkconfig** command and start it 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.

- 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:/var/nfs/rhel62/ /mnt/tmp
-or-
# mount localhost:/var/nfs/sles11sp2/ /mnt/tmp
# ls -F /mnt/tmp
...
```

  This shows that the install files is accessible through NFS.

- Unmount the rhel62/ directory, then test the CKB-VM62/ directory:

```
# umount /mnt/tmp
# mount localhost:/var/nfs/CKB-VM62 /mnt/tmp
# ls -F /mnt/tmp
disclaimer.txt  README.txt  rhel62/  sles11sp2/  vm/
# umount /mnt/tmp
```

You should now be able to use this server as the source of a RHEL 6.2 mainframe Linux installation. Later you will be able to copy the install tree to a System z Linux virtual server.

4.6 Configure an FTP server

If you will also be using this system as an FTP server to install z/VM, see section 5.2, “Configure an FTP server for z/VM installation” on page 44.

5

Install a z/VM SSI cluster

"*Example isn't another way to teach. It is the only way to teach.*"

— Albert Einstein

Important: These are not official instructions on how to install z/VM 6.2. The program directory, installation manual and other documents are on the Web starting at:

http://www.vm.ibm.com/progdir/

From that page, there is a link to the *z/VM Installation Guide, version 6 release 2* at:

http://www.vm.ibm.com/progdir/hcsk2c10.pdf

The set of official z/VM manuals is on the Web at:

http://publib.boulder.ibm.com/cgi-bin/bookmgr/Shelves/hcsh2ac0

This chapter describes installing z/VM 6.2 from an FTP server onto a two node SSI cluster residing on DASD. It also addresses installing from DVDs. If you are installing onto SCSI disks or using significantly different parameters, you should use the official z/VM documentation.

z/VM 6.2 can be installed from a DVD disc or from an FTP server. Installing from FTP server is described in this section while installing from DVD is also mentioned.

This chapter consists of the following sections that should be completed:

- “Add page and perm volumes” on page 73
- “Create a virtual machine for common files” on page 78
- “Configure AUTOLOG1’s PROFILE EXEC” on page 84

In addition, there are optional sections:

- “z/VM security issues” on page 86
- “Back up and restore your z/VM system” on page 88

5.1 Obtain z/VM through electronic download

z/VM can be ordered and delivered electronically through IBM *ShopzSeries*. A detailed discussion is outside the scope of this book, however short steps are documented. Note that the steps and links may change over time, but the basic process should remain the same.

You may download the z/VM product install files to a staging machine, such as a Windows desktop, as was done in this example, and later upload them to an FTP server. However, you may also download them directly to the machine that will be the FTP server, such as a Linux PC if it has access to the Internet.

To order z/VM, perform the following steps:

- Go to the z/VM service page:

  http://www.vm.ibm.com/service/
- Click on the link **IBM Shopz** in the section *IBM Support Portals*.
- Sign in by clicking on the link **Sign in for registered users** in the upper right.
- Click on the link **create new software orders** for service or products.
- On *Step 1*, click on the radio button **z/VM Products** and choose **VM SDO version 6** in the drop-down menu to the right. Click **Continue**.
- On *Step 2*, select a hardware system on which you plan to run z/VM from the list of Hardware systems for your customer number, and click **Continue**.
- On *Step 3*, for the Filter, select **VM - VM Base Product,** select your language and for the *Filter*, select **Show all products**. then click **Show catalog**. A sub-menu appears.
  – Select **z/VM V6 3390 System DDR** and click **Continue**.
- On *Step 4*, verify the order and click **Continue**.
- On *Step 5*, verify the entitlements and click **Continue**.
- On *Step 6*, for the *Preferred media*, select **Internet** and click **Continue**.
- On *Step 7*, review and click **Submit**.
- It may take some time for the order to be prepared. In this example, the e-mail stating that the order was ready for download was received after about four hours. When you receive the e-mail, it will contain the URL for downloading your order. Use a browser to go to that URL.
- There will be links to investigate as shown in Figure 5-1. It has the following five sections:
  – *Order Packing List* - the list of available products and manuals
  – *Installation Instructions* - clicking **View now** will take you to a Web page:

    http://www.vm.ibm.com/install/prodinst.html
  – *Product Publications* - will allow you to access different z/VM publications related to installation

– *Additional Publications* - will allow you to download a z/VM SDO document (4 pages)

– *VM product material* - *This is the most important section* as it is where you go to download z/VM product installation files. In the example used in this book, the link **Download to your workstation using IBM Download Director** was clicked as shown in the figure.

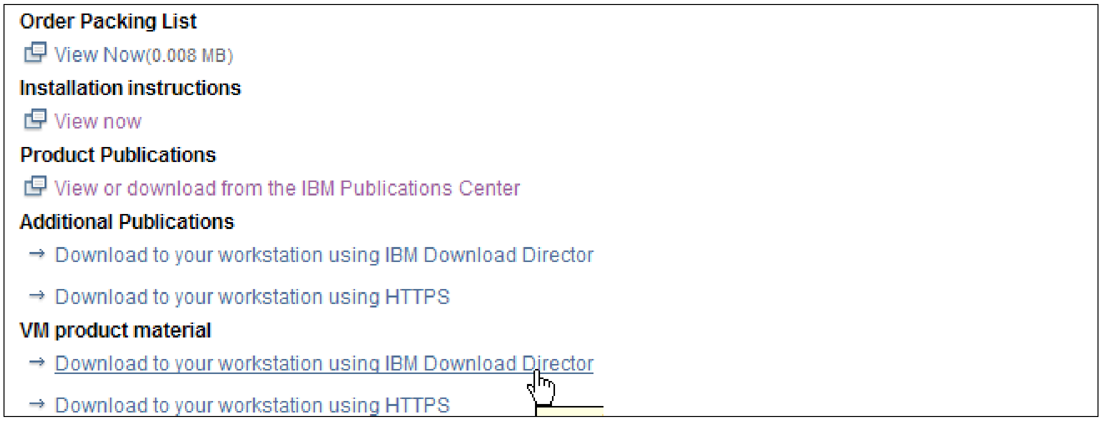


Figure 5-1 Web page for downloading z/VM electronically

- Clicking this link brought up the screen shown in Figure 5-2 on page 43. The second and third check boxes were selected as the z/VM 6.2is being installed onto 3390 DASD. The 1.3 GB of data was downloaded relatively quickly due to multiple connections being opened through the use of *IBM Download Director*.

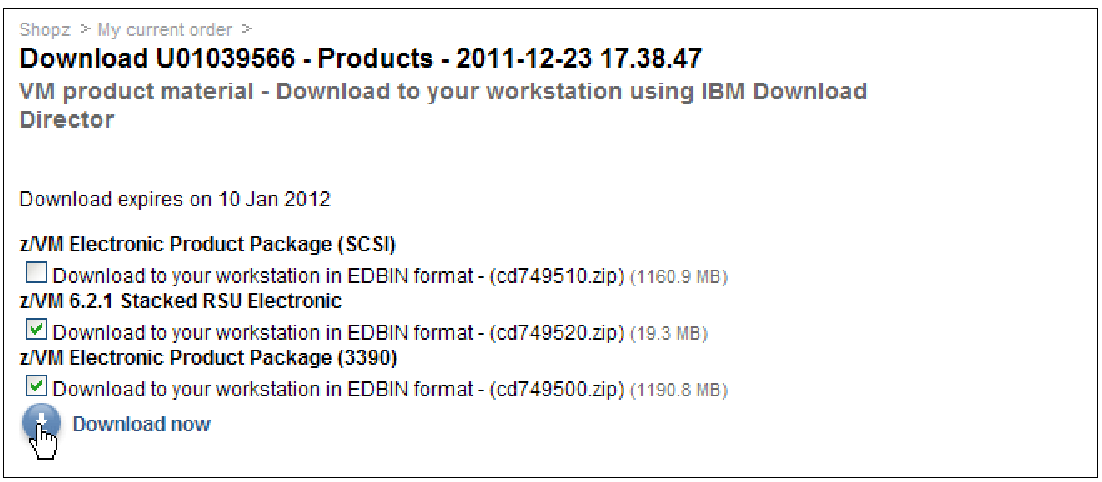


Figure 5-2 Choosing two files to be downloaded

- The z/VM install code should now be staged or ready for the FTP server to be set up. In this example where the files are staged on a Windows workstation, the two files are shown from a DOS prompt:

```
C:\zvm62> dir
...
12/24/2011  08:45 AM     1,190,826,525 cd749500.zip
12/24/2011  12:22 PM        19,268,830 cd749520.zip
```

You now have the two z/VM install zip files downloaded. You can now set up an FTP server.

5.2 Configure an FTP server for z/VM installation

This section assumes that you have access to the z/VM 6.2 install code in electronic format. Ordering it through ShopzSeries is briefly described in section 5.1, "Obtain z/VM through electronic download" on page 42. If you have completed that section, you may have the two z/VM product install files staged on a intermediate workstation, or you may be ready to download them from the Internet.

5.2.1 Prepare the z/VM product install files

The two zip files correspond to the larger first z/VM product DVD, and to the smaller second DVD - the RSU. The contents of these files must be copied to the directory of the FTP server. To accomplish this, perform the following steps:

- Start an SSH session to the distributed NFS server set up in Chapter 4, "Configure an NFS/FTP server" on page 33.
- Create a target directory. In this example the directory `/var/ftp/zVM62/` is used:

```
# mkdir -p /var/ftp/zVM62
```

- Set the group ownership of this directory, recursively, to **ftp**. This will allow the FTP daemon, which runs as the user `ftp`, to change directory into it:

```
# chgrp -R ftp /var/ftp/zvm62
```

- Either upload the two z/VM installation zip files from the intermediate workstation, or download them directly from the Internet. The following example shows copying them from an intermediate workstation Windows DOS session to the FTP server at the IP address 9.60.18.240 in the directory, `/var/ftp/zvm62/` using the add-on **pscp** command (Putty scp):

```
C:\zvm62>pscp *.zip root@9.60.18.240:/var/ftp/zvm62
...
```

- List the newly copied files:

```
# cd /var/ftp/zvm62
# ls -l
total 1182920
-rw-r--r-- 1 root root 1190826525 Nov 14 15:39 CD749500.zip
-rw-r--r-- 1 root root   19268830 Nov 14 13:14 CD749520.zip
```

- Unzip the files from DVD1, the larger file, using the **unzip** command. This will create the directory `cpdvd/`:

```
# unzip CD749500.zip
Archive: CD749500.zip
   creating: cpdvd/
  inflating: cpdvd/620GANUC
  inflating: cpdvd/620GARAM
  inflating: cpdvd/620prod.dvdimag
...
```

- Unzip the files from the RSU DVD2, the smaller file. When prompted to replace files, respond with **A** for all:

```
# unzip CD749520.zip
Archive:  CD749520.zip
replace cpdvd/620RSU.DVDIMAGE? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
  inflating: cpdvd/620RSU.DVDIMAGE
  inflating: cpdvd/62CKDRSU.SRL
  inflating: cpdvd/62FBARSU.SRL
```

```
  inflating: cpdvd/CKD50000
  inflating: cpdvd/CKD50001
  inflating: cpdvd/CKD50002
  inflating: cpdvd/CKD50003
  inflating: cpdvd/CKD50004
  inflating: cpdvd/CKD50005
  inflating: cpdvd/CKD50006
  inflating: cpdvd/CKD50007
  inflating: cpdvd/FBA50000
  inflating: cpdvd/FBA50001
  inflating: cpdvd/FBA50002
  inflating: cpdvd/FBA50003
  inflating: cpdvd/FBA50004
  inflating: cpdvd/FBA50005
  inflating: cpdvd/FBA50006
  inflating: cpdvd/FBA50007
  inflating: cpdvd/RSULEVEL.6201
```

You should now have all the z/VM product install files in place under the directory `/var/ftp/zvm62/cpdvd/`.

5.2.2 Install and configuring the FTP server

An FTP server must be installed and configured. The **vsftpd** FTP server is recommended. This section shows how to configure it as an anonymous FTP server. To accomplish these tasks, perform the following steps:

- Use the **rpm -qa** command to see if the RPM is installed:

```
# rpm -qa | grep ftpd
```

- No output shows that it is not installed. Use the **yum -y** command to install the package:

```
# yum -y install vsftpd
Loaded plugins: rhnplugin
This system is not registered with RHN.
...
Installed:
  vsftpd.s390x 0:2.2.2-6.el6
```

- Make a backup of the **vsftpd** configuration file, `/etc/vsftpd/vsftpd.conf`:

```
# cd /etc/vsftpd
# cp vsftpd.conf vsftpd.conf.orig
```

- Modify the configuration file to set the directory that anonymous user will be logged in to `/var/ftp/zvm62/` using the `anon_root` variable. Also disable local (non-anonymous) logins by commenting out the `local_enable=YES` and `write_enable=YES` lines.

```
# Example config file /etc/vsftpd/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=YES
# set the home directory of anonymous FTP to /var/ftp/
```

```
anon_root=/var/ftp/
#
# Uncomment this to allow local users to log in.
# local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
# write_enable=YES
...
```

- Set the vsftpd service to start at boot time with the **chkconfig** command and for this session with the **service** command:

```
# chkconfig vsftpd on
# service vsftpd start
Starting vsftpd for vsftpd: [  OK  ]
```

An anonymous FTP server should now be running with the z/VM 6.2 directory in `/cpdvd` (relative to the anonymous FTP root directory).

5.2.3 Test the anonymous FTP server

Test the setup by FTPing in as anonymous from another system. You should see the `cpdvd/` directory:

```
# ftp gpok240
Connected to gpok240.endicott.ibm.com.
220 (vsFTPd 2.2.2)
Name (gpok240:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
229 Entering Extended Passive Mode (|||6252|).
150 Here comes the directory listing.
-rw-r--r--    1 0        0        45088210 Nov 11 19:06 CD813270.ZIP
dr-xr-xr-x    2 0        0           24576 Nov 11 19:23 cpdvd
226 Directory send OK.
ftp> quit
```

This shows that the anonymous FTP server is working. You should now be able to install z/VM 6.2 from this FTP server.

5.3 Install z/VM from DVD or FTP server

The sections that follow assume a first level installation of z/VM from DVD or FTP server onto 3390 DASD. If you have not already done so, complete the worksheet in 2.8.2, “z/VM resources worksheet” on page 19.

If you are not familiar with the HMC and z/ VM, you may want to use the official z/VM manual: *z/VM 6.2 Installation Guide*, GC24-6246.

http://www.vm.ibm.com/progdir/hcsk2c10.pdf

If you are installing z/VM at the *second level* (z/VM under z/VM) or onto FCP/SCSI disk, you will want to use this z/VM manual as the sections that follow do not address these options.

5.3.1 Start the z/VM install

The figures in this book still show the HMC *classic* view. The new hierarchical view, as shown in Figure 5-3, is being used more frequently.

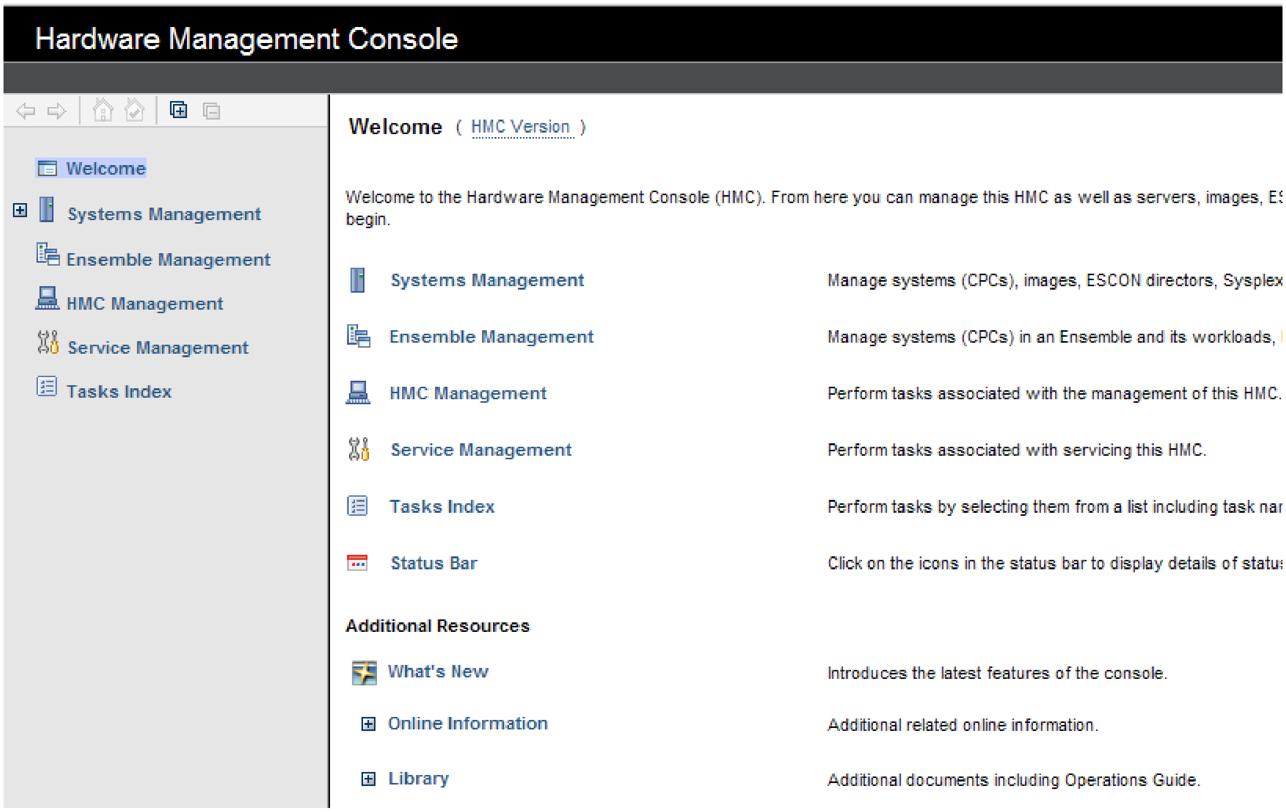


Figure 5-3 HMC Hierarchical view

To begin the z/VM 6.2 installation, perform the following steps:

- Logon to the Hardware Management Console. You should see the *HMC Workplace* window.
- Select the LPAR that will be the first member of the z/VM 6.2 SSI cluster. You may have to click **CPC images** icon. *Be sure* you have the correct LPAR selected. If you are not completely sure, check with someone who is.
- Click the *racetrack* buttons (two buttons that are circular arrows on the bottom right corner) to traverse to the *Recovery* menu. Usually, this requires one click of the counter-clockwise button.
- On the *Recovery* menu, double-click the **Integrated 3270 Console** as shown at the inFigure 5-4 on page 48. A window entitled *Integrated 3270 Console* will open.

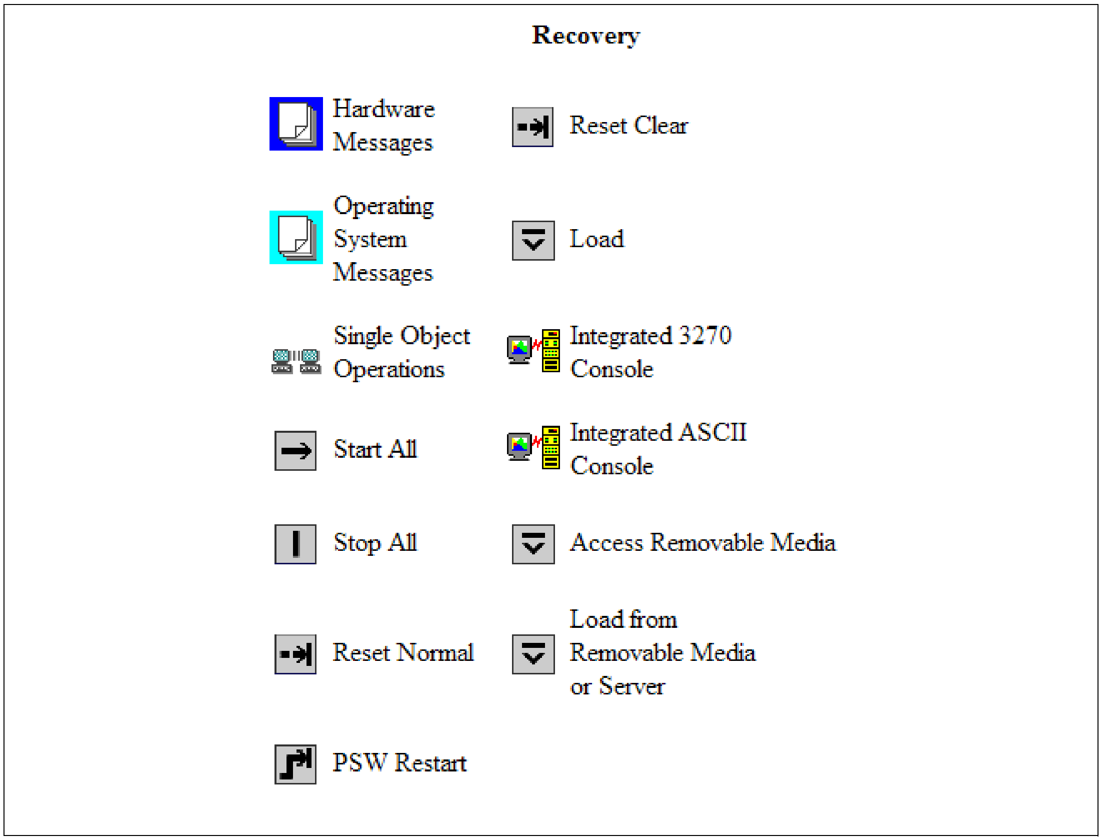


Figure 5-4 HMC Recovery menu

The LPAR that will become the first member of the SSI should still be selected. On the right you should still see the *Recovery* menu. Double-click the **Load from Removable Media or Server** icon

- On the *Load from Removable Media or Server* window as shown in Figure 5-5 on page 49, click the radio button **FTP Source**. Enter the *Host computer* the FTP server is running on, the *User ID* and *Password* and in the *File location* field, the directory where the `620vm.ins` file is. In this example, it is **zvm62/cpdvd** (as the home directory of the FTP user anonymous was set to `/var/ftp/`) Click **OK**.

If installing from DVDs: If you are installing from DVDs, the first DVD must be in the HMC drive and the *FTP Source* section will not be used.

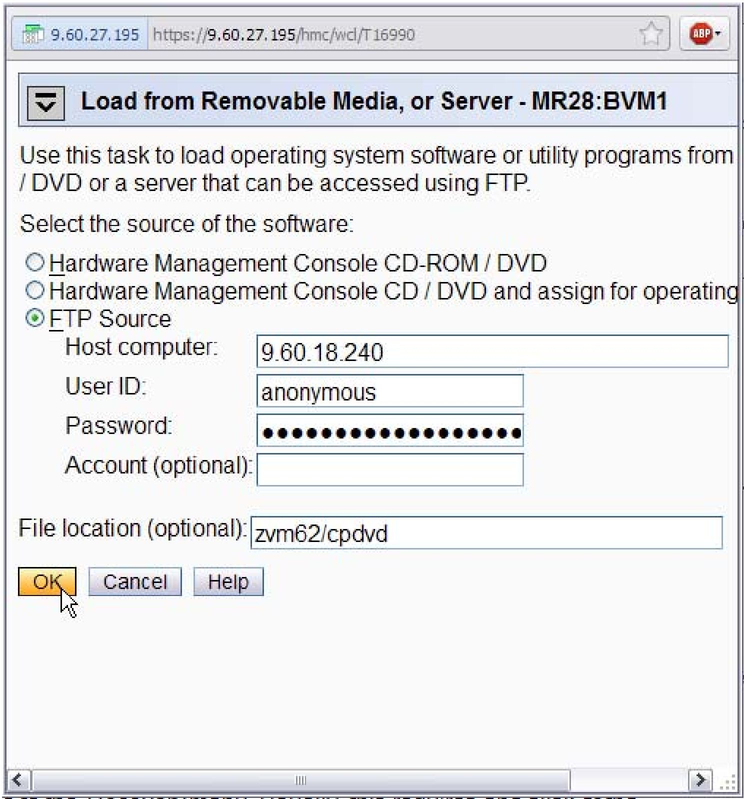


Figure 5-5 Load from Removable Media or Server panel

- Load the RAMDISK:

  a. From the *Load from Removable Media or Server* panel, the file **620vm.ins** should be selected as shown in Figure 5-6. Click **OK**.

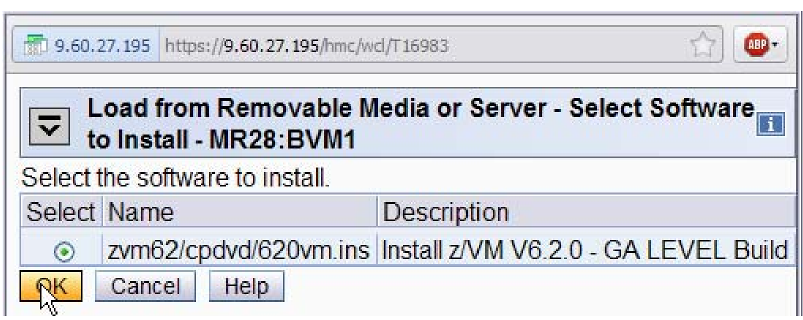


Figure 5-6 Selecting z/VM 6.2 RAMdisk system

  b. From the *Confirm the action* window, click **Yes**.

  c. You should see the *Disruptive Task Confirmation: Load from CD-ROM, DVD or Server Progress* window as shown in Figure 5-7 on page 50. You *may* be prompted for the password depending on how your HMC is configured.

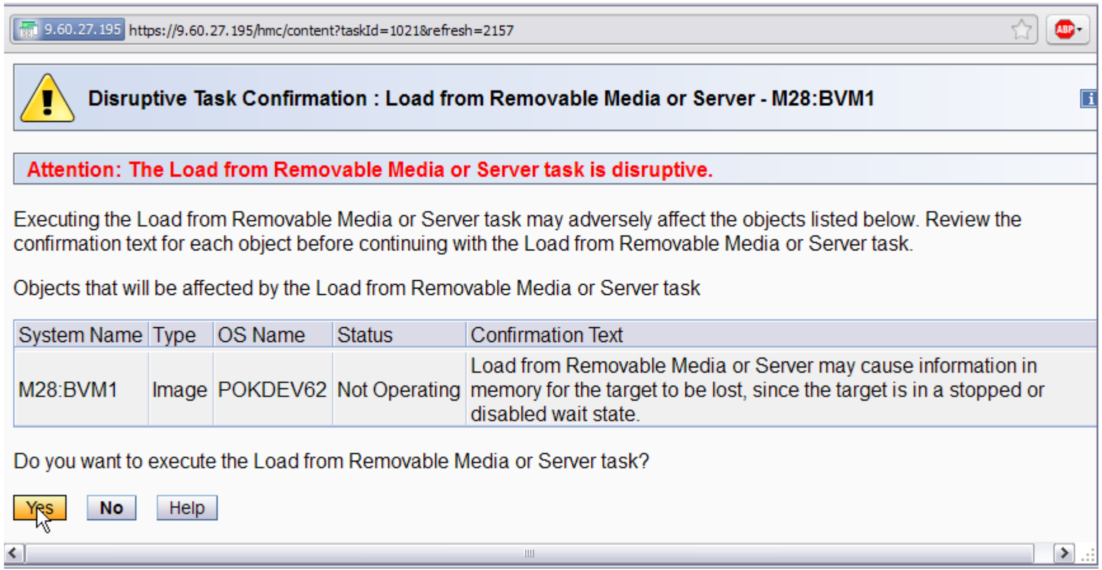


Figure 5-7 Informational window for disruptive task

d. You should see the *Load from Removable media or Server Progress* window. When you see the message `Completed successfully`. Click **OK** to close. This should normally take two minutes or less.

You now have an in-memory z/VM 6.2 system running.

5.3.2 Copy a vanilla z/VM system to DASD

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

- Move to the *Integrated 3270 Console* window. The RAMdisk should IPL and you should see z/VM boot as shown in Figure 5-8 on page 51. If the Integrated 3270 Console window is still blank, be patient - it may take a minute or two to initialize.

  Note: The “**Esc**” key in the upper left clears the Integrated 3270 console on the HMC.

```
MrMutt: Integrated 3270 Console for M28:BVM1
File  Keys  Font  Help
 09:20:35 z/VM  V6 R2.0  SERVICE LEVEL 0000 (64-BIT)
 09:20:36 SYSTEM NUCLEUS CREATED ON 2010-05-27 AT 12:20:54, LOADED FROM $RAMD$
09:20:36
09:20:36 ****************************************************************
09:20:36 * LICENSED MATERIALS - PROPERTY OF IBM*                        *
09:20:36 *                                                              *
09:20:36 * 5741-A07 (C) COPYRIGHT IBM CORP. 1983, 2010. ALL RIGHTS      *
09:20:36 * RESERVED. US GOVERNMENT USERS RESTRICTED RIGHTS - USE,       *
09:20:36 * DUPLICATION OR DISCLOSURE RESTRICTED BY GSA ADP SCHEDULE     *
09:20:36 * CONTRACT WITH IBM CORP.                                      *
09:20:36 *                                                              *
09:20:36 * * TRADEMARK OF INTERNATIONAL BUSINESS MACHINES.              *
09:20:36 ****************************************************************
09:20:36
 09:20:36 HCPZCO6718I Using parm disk 1 on volume $RAMD$ (device FFFF).
 09:20:36 HCPZCO6718I Parm disk resides on blocks 18000 through 52992.
09:20:36 The directory on volume $RAMD$ at address FFFF has been brought online.
 09:20:36 HCPWRS2512I Spooling initialization is complete.
09:20:36 No dump unit - Dump function is SET OFF
09:20:36 HCPMLM3016I HMC/SE ensemble management is not available for this system
.
09:20:36 HCPAAU2700I System gateway IBMVMRAM identified.
09:20:37 HCPLNM6640E MAINT FFFF not linked.  Minidisk has been defined with the
V mode suffix and other links to this minidisk exist.
09:20:37 z/VM Version 6 Release 2.0, Service Level 0000 (64-bit),
09:20:37 built on IBM Virtualization Technology
09:20:37 There is no logmsg data
09:20:37 FILES:   NO RDR,   NO PRT,   NO PUN
09:20:37 LOGON AT 09:20:37 EDT MONDAY 06/27/11
09:20:37 SYSG       LOGON  AS  MAINT    USERS = 1
09:20:37 HCPIOP952I 10G system storage
09:20:37 FILES: 0000001 RDR, 0000001 PRT,      NO PUN
09:20:37 HCPCRC8082I Accounting records are accumulating for userid OPERACCT.
09:20:37 HCPCRC8082I EREP records are accumulating for userid OPEREREP.
DMSIND2015W Unable to access the Y-disk. Filemode Y (19E) not accessed
DMSWSP327I The installation saved segment could not be loaded
z/VM V6.2.0    2011-04-11 14:15
DMSDCS1083E Saved segment CMSPIPES does not exist
DMSDCS1083E Saved segment CMSPIPES does not exist
DMSDCS1083E Saved segment CMSVMLIB does not exist
Ready; T=0.01/0.02 09:20:37

                                                  RUNNING   IBMVMRAM
                                                           42/1
```

Figure 5-8   First z/VM 6.2 install screen

Run DVDPRIME

- Run the **`dvdprime`** command as shown in Figure 5-9. The format is **`dvdprime`** ***`dasdtype`*** **`(`*****`source`***. In this example the ***`dasdtype`*** is **`3390`** and the ***`source`*** is **`server`** - for FTP server.

```
==> dvdprime 3390 (server
```

- You should see the following output:

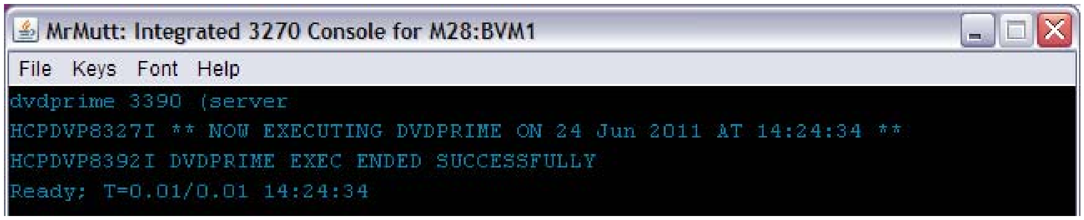


Figure 5-9 Running the DVDPRIME command

Run INSTPLAN

- Run the **`instplan dvd`** command to set up the configuration for the install process. You should see the *z/VM INSTALLATION PLANNING* panel.

```
==> instplan dvd
```

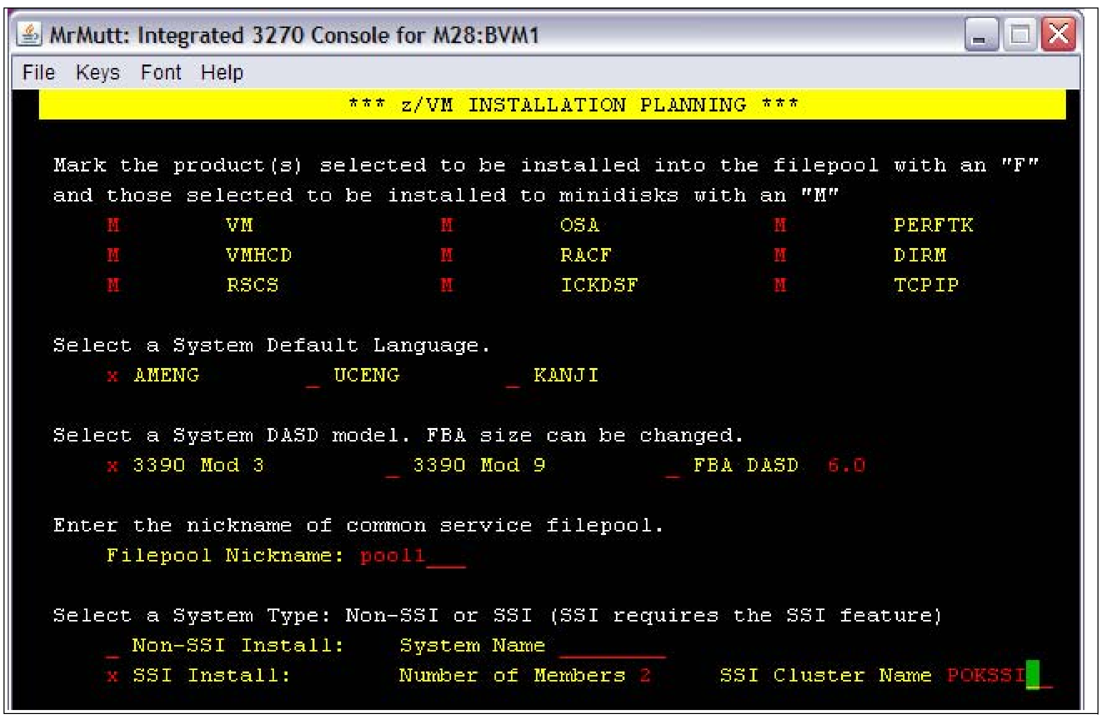


Figure 5-10 Installation planning panel

- You may need to clear the screen with the **Esc** key. You should then see the display as shown in Figure 5-10. It is recommended that you leave the M's in the top section alone.
- Type the letter **`x`** next to `AMENG` (or select your language) and `3390 Mod 3` (or the type of DASD you will use) as shown above. A Filepool name of ***`pool1`*** is used in this example. Type the letter **`x`** next to SSI Install, set the number of members (**2** in this example) and choose a name for the cluster (***`POKSSI`*** in this example).
- Press **F5**. You should see the following panel.

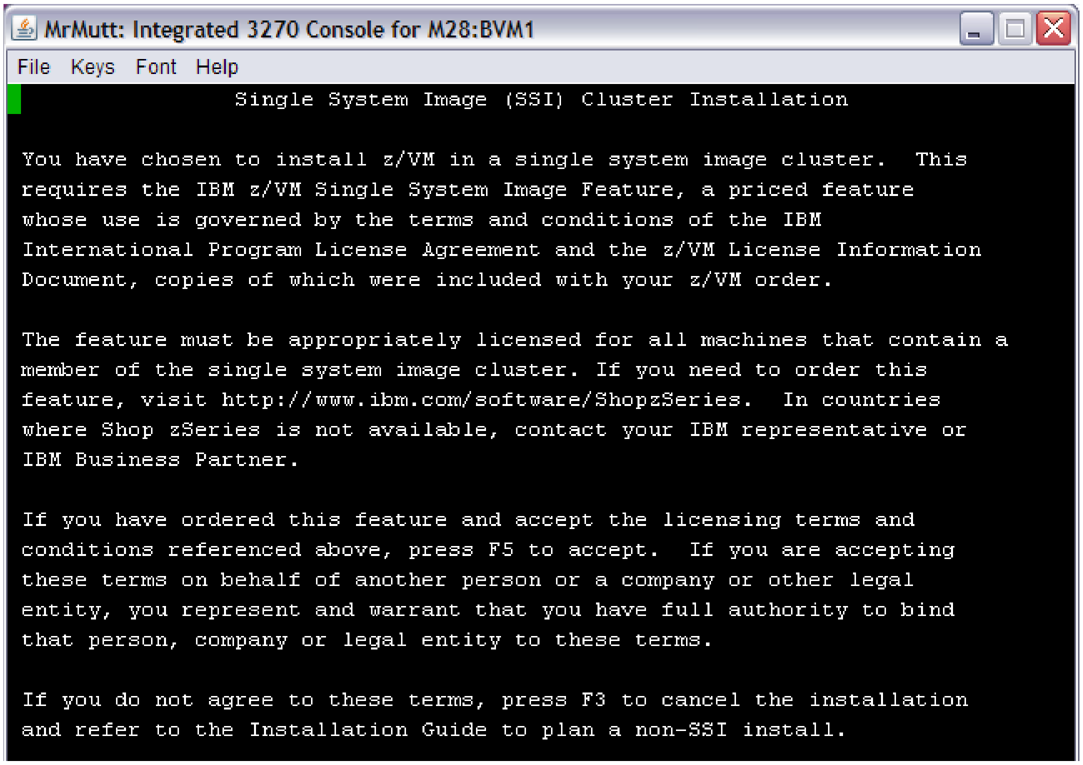


Figure 5-11 SSI Cluster Installation panel

- Press **F5** to accept the terms.
- You should see the *z/VM INSTALLATION PLANNING PANEL 2* as shown in Figure 5-12. Answer **no** to the question about having your system managed. Press **F5** to continue.

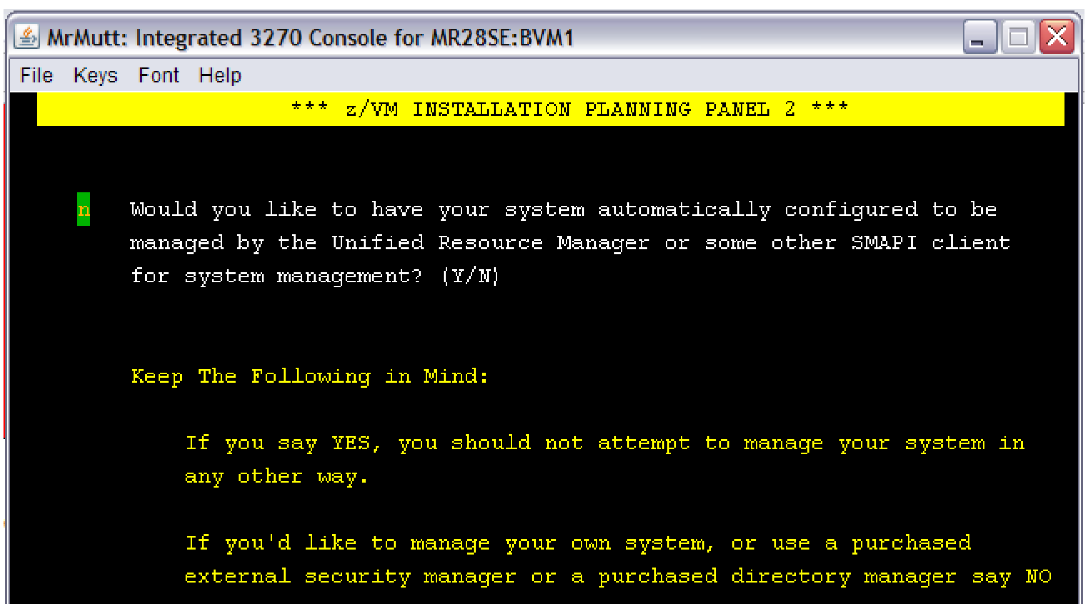


Figure 5-12 z.VM Installation Planning Panel 2

- You should see the *z/VM INSTALLATION PLANNING PANEL 3* as shown in Figure 5-13 on page 54. Enter the SSI member names and their corresponding LPAR names as seen on the HMC. Press **F5** to continue.

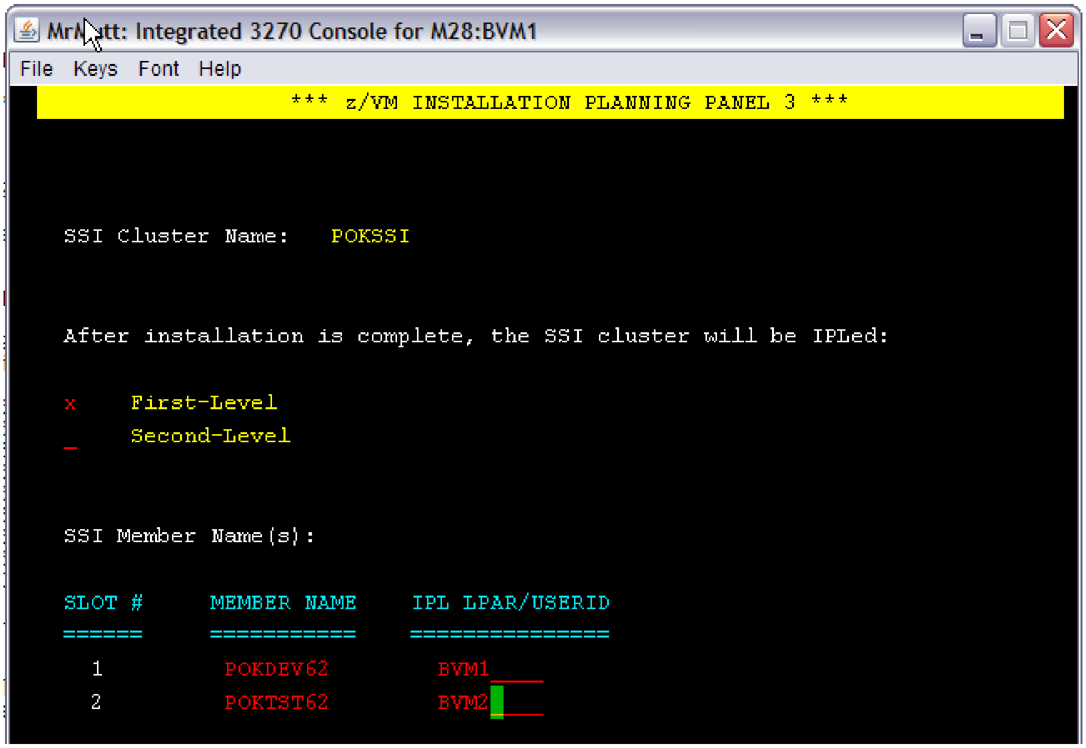


Figure 5-13 z/VM Installation Planning Panel 3

- You will be shown a summary of your choices. If the values are correct, type **Y** to the question `DO YOU WANT TO CONTINUE?`
- You should now see the *z/VM INSTALLATION VOLUME DEFINITION* panel as shown in Figure 5-14:

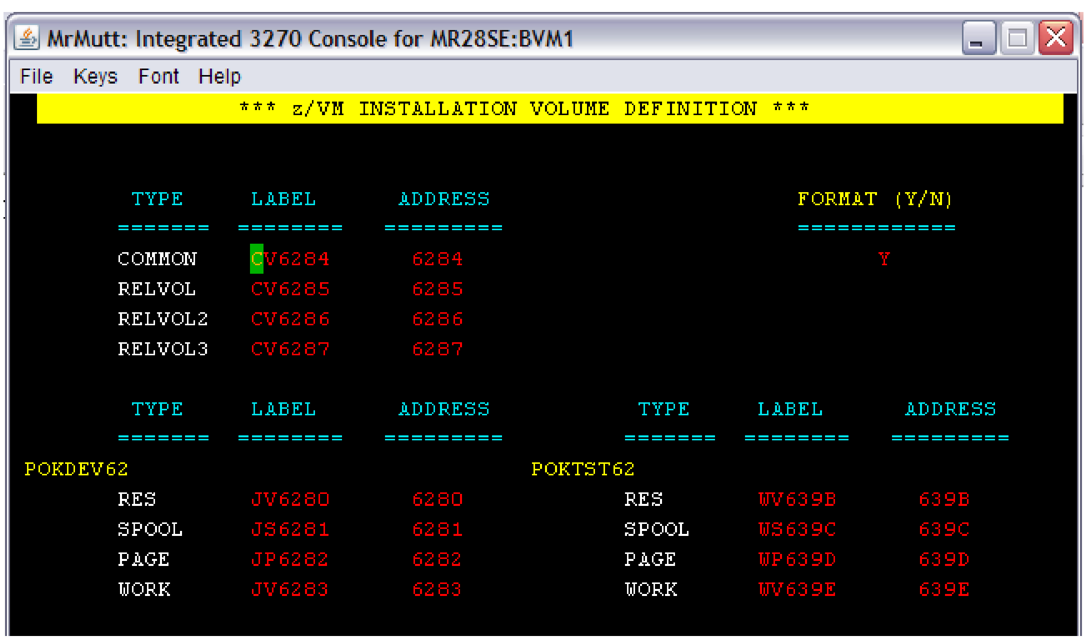


Figure 5-14 z/VM Installation Volume Definition panel

- Type in the volume labels and addresses from your worksheet. In this example, a prefix character of **J** is used for member 1, **W** for member 2 and **C** for common volumes. Press **F5** to continue.
- You should see the *z/VM Installation First-level Configuration* panel as shown in Figure 5-15 on page 55. The common volume addresses will almost always be identical (unless the same DASD was assigned different real device addresses on other LPARs).

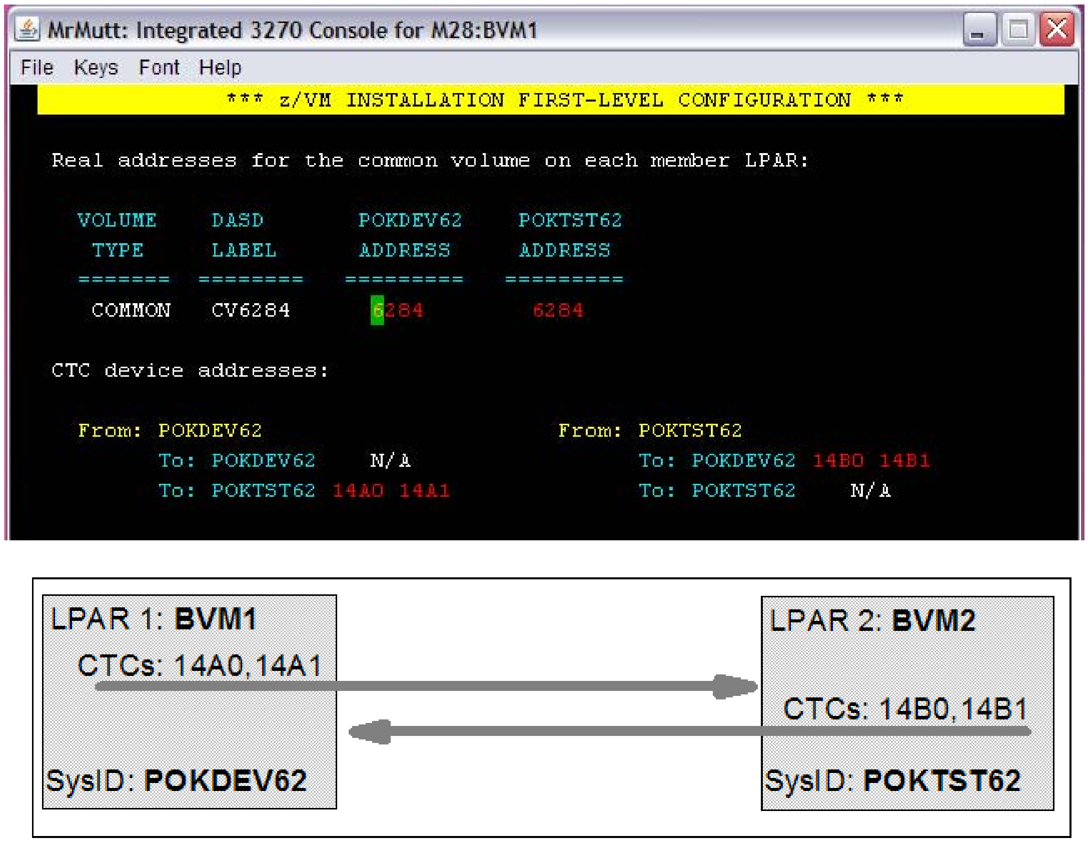


Figure 5-15 z/VM Installation First-Level Configuration

- Press **F5**. You should see a summary of your values, then the message:

```
...
HCPINP8392I INSTPLAN EXEC ENDED SUCCESSFULLY.
```

- Attach all DASD that will be part of the SSI cluster to `MAINT` with the **`ATTACH`** command. In this example, it is

```
==> att 6280-6287 639b-639e *
15:10:13 6280-6287 ATTACHED TO MAINT
15:10:13 639B-639E ATTACHED TO MAINT
```

Important: The devices ***`6280-6287 639B-639E`*** are in bold italics to signify that you should replace the example value with the correct value for your site. This convention is used throughout the book.

- Run the **`INSTALL`** command as shown in Figure 5-16. The DASD will be formatted and the z/VM system disks will be copied. The start of this process is shown in Figure 5-16. This step usually takes one to two hours.

```
File   Keys   Font   Help
install
HCPIIS8490I NOW FORMATTING VOLUME 6284 (1 OF 12)
09:40:53 HCPCRC8083I EREP record threshold has been exceeded for userid OPERERE
. Currently 00000002 records are enqueued.
09:40:53 DASD  6284 DETACHED MAINT    6284 BY MAINT
09:40:53 DASD 6284 ATTACHED TO MAINT 6284 BY MAINT WITH DEVCTL
HCPIIS8490I NOW FORMATTING VOLUME 6285 (2 OF 12)
09:42:10 DASD  6285 DETACHED MAINT    6285 BY MAINT
09:42:10 DASD 6285 ATTACHED TO MAINT 6285 BY MAINT WITH DEVCTL
HCPIIS8490I NOW FORMATTING VOLUME 6286 (3 OF 12)
```

Figure 5-16 Attaching DASD and running the INSTALL command

- If you are installing from DVD, you will be asked to place the system RSU in the drive. Insert the **z/VM Stacked Recommended Service Upgrade** DVD into the HMC DVD drive. Type **GO**. You should see a messages of the form `DVDLOAD: LOADING FILE CKD5000x IMAGE *`.

  If you are installing from an FTP server, you will not see this prompt.

- Finally, you should see the message `HCPMLP8392I INSTALL EXEC ENDED SUCCESSFULLY`.

  It is imperative the **INSTALL** EXEC succeeds. If it does not, you must go back and fix it.

- Run the **INSTSCID REMOVE** command to update the `SYSTEM CONFIG` file.

```
==> instscid remove
```

```
instscid remove
**********************************************************
*     PROCESSING REMOVE FOR ALL MEMBERS
**********************************************************
MSGPFX8404I SYSTEM CONFIG has been updated to allow all members
            to be IPL'ed ONLY from the LPAR/USERid
            defined for each member at install time.

MSGPFX8392I INSTSCID EXEC ENDED SUCCESSFULLY
```

Figure 5-17 INSTSCID REMOVE command

- Run the **SHUTDOWN** command. You should see the system going down ending in a disabled wait with a state code of 961:

```
==> shutdown
...
HCPGIR450W CP entered; disabled wait PSW 00020000 00000000 00000000 00000961
```

- IPL CMS from the 190 disk:

```
==> ipl 190
```

- Press **Enter** at the `VM READ` prompt.

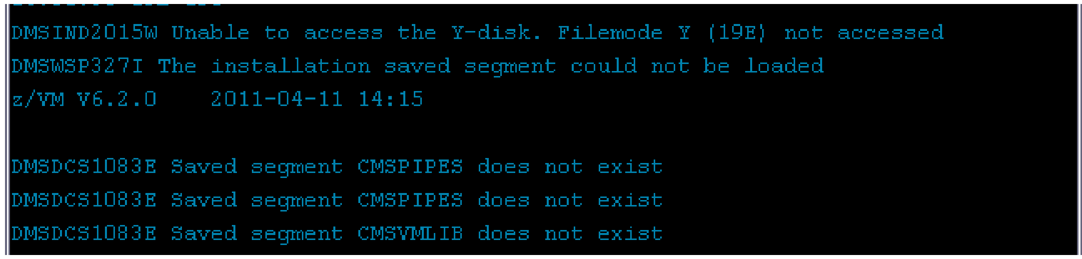


Figure 5-18 IPLing 190

- Shutdown the system again:

```
==> shutdown system ibmvmram
16:03:37 SYSTEM SHUTDOWN STARTED
```

  The in-memory copy of z/VM will now be halted on SSI member 1. The LPAR should turn red on the HMC.

5.3.3 IPL the first SSI member

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

- In the *HMC Workplace* window, the LPAR of the first SSI member should still be selected. From the *Recovery* menu. Double-click the **Load** icon in the menu at the right side.
- The Load window opens as shown in Figure 5-19 on page 58. Follow these steps:
  a. Set the *Load Address* to the new system residence volume, which is ***6280*** in this example.
  b. Set the *Load Parameter* to **SYSG**. This specifies to use the Integrated 3270 console.
  c. Click **OK** to IPL.

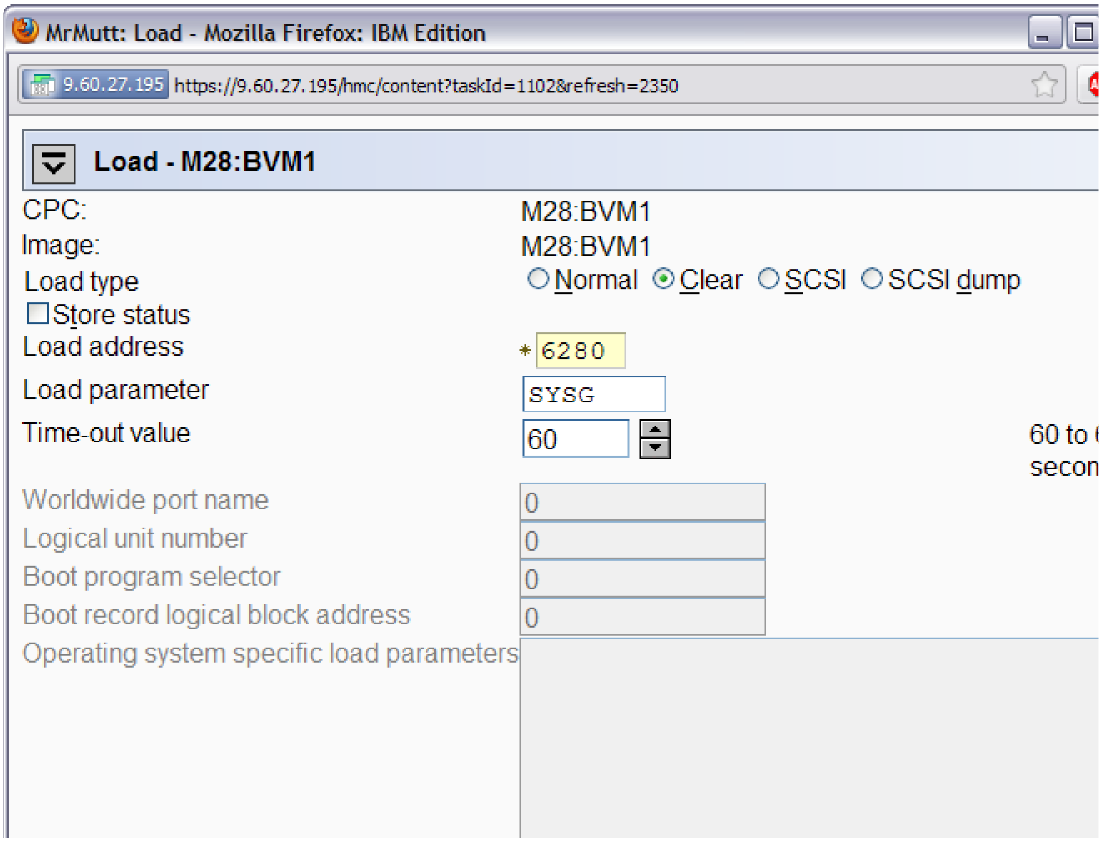


Figure 5-19 Load window

- When you see the *Load Task Confirmation* window, click **Yes**.
- After a minute or less you should see a status of *Success* in the *Load Progress* window. Click **OK**.
- Move back to the Integrated 3270 console window. You should see the *Standalone Program Loader* panel as shown in Figure 5-20 on page 59. Press the **F10** key to continue the IPL of your z/VM system. It may take a while for the system to start IPLing.

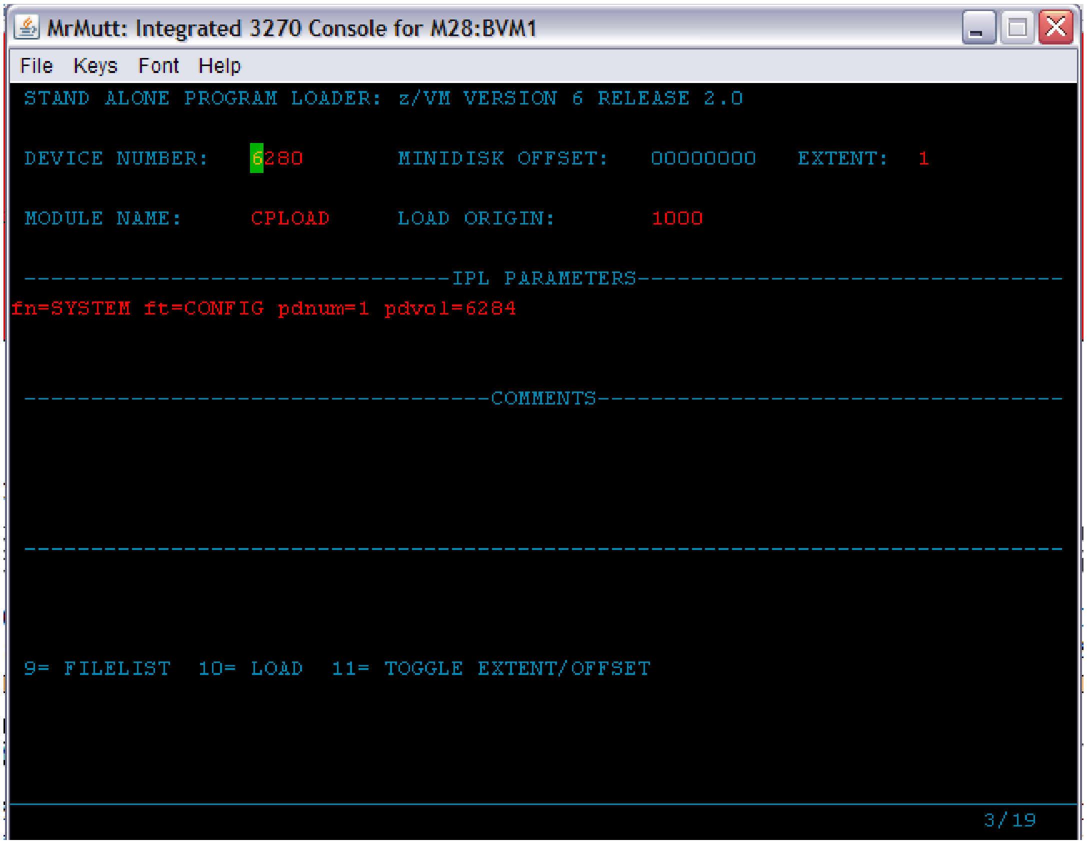


Figure 5-20 Stand Alone Program Loader

- At the `Start (Warm|Force|COLD|CLEAN)` prompt, enter the following:

  `==>` **`cold drain`**

- At the Change TOD clock prompt enter **no**:

  `==>` **`no`**

- To the message, `To continue COLD start and delete files, enter GO`, type **go**:

  `==>` **`go`**

- The first SSI member should IPL cleanly. The last message should be `HCPCRC8082I EREP records are accumulating for userID EREP`. Disconnect from the `OPERATOR` virtual machine using the **`DISCONNECT`** command:

  `==>` **`disc`**

5.3.4 IPL remaining SSI members

In this example of a two node SSI cluster, there is only one more member. If you are creating a four member SSI cluster, you will have three more members.

- IPL each of the additional members from the HMC by starting an *Integrated 3270 Console* and using the **Load** operation. Figure 5-21 on page 60 shows the second member IPLing. Note the messages prefixed with `HCPPLM` showing the status of the SSI cluster.

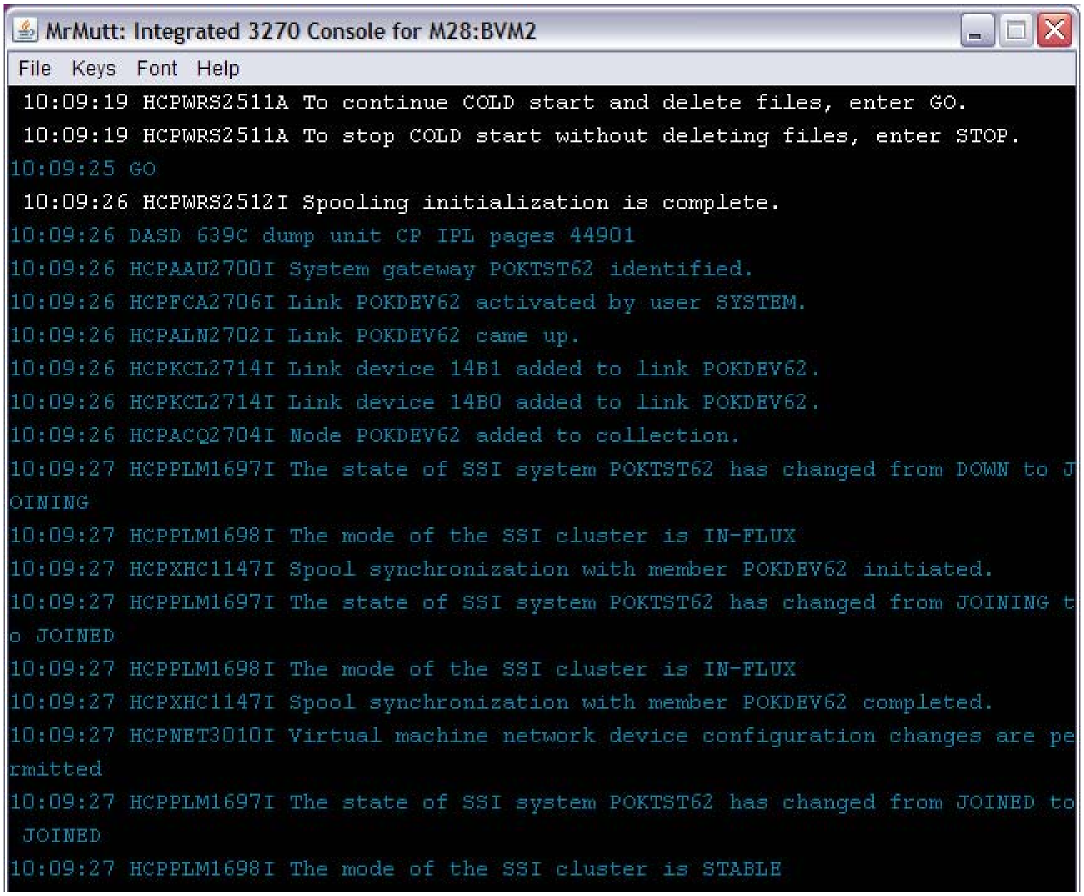


Figure 5-21 Console of second SSI member joining the cluster

Important: You may see the following message:

```
HCPPLM1669I Waiting for ISFC connectivity in order to join the cluster.
```

This is not good. The member will likely wait forever to join. Check with the system administrator and verify the CTCs are set up correctly and that you used the correct values. Verify that you typed the CTCs correctly. Figure 5-15 on page 55 shows a block diagram of the CTCs used in this example.

- Use the **DISCONNECT** command to disconnect from OPERATOR on each of the members

```
==> disc
```

5.3.5 Verify the installation

Perform one more check to be sure the RSU was installed:

- Log on to MAINT.
- Issue the QUERY CPLEVEL command to see the RSU level. Be sure the service level is **1101** (not 0000):

```
==> q cplevel
z/VM Version 6 Release 2.0, service level 1101 (64-bit)
```

```
Generated at 11/02/11 18:54:31 EST
IPL at 12/13/11 08:42:35 EST
```

If the service level is `0000`, you must fix it. Reinstalling z/VM may be the most expedient way of doing so.

Congratulations! You should now have a z/VM 6.2 SSI cluster.

5.4 Configure TCP/IP

It is recommended that you initially configure TCP/IP using the **`IPWIZARD`** command on each of the SSI members. This wizard is generally used just once. After **`IPWIZARD`** creates the initial configuration files, they are typically maintained manually. A temporary OSA triplet is used to initially get z/VM on the network. Later, the TCP/IP stack will be attached to the highly available VSWITCH.

To configure TCP/IP, perform the following steps:

- From the HMC z/VM logon panel, **logon to `MAINT`**. The default password for all z/VM virtual machines is the same as the user/identity ID. So enter a password of **`maint`** which will not be echoed on the screen.

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

  You should see output similar to that shown in Figure 5-22.

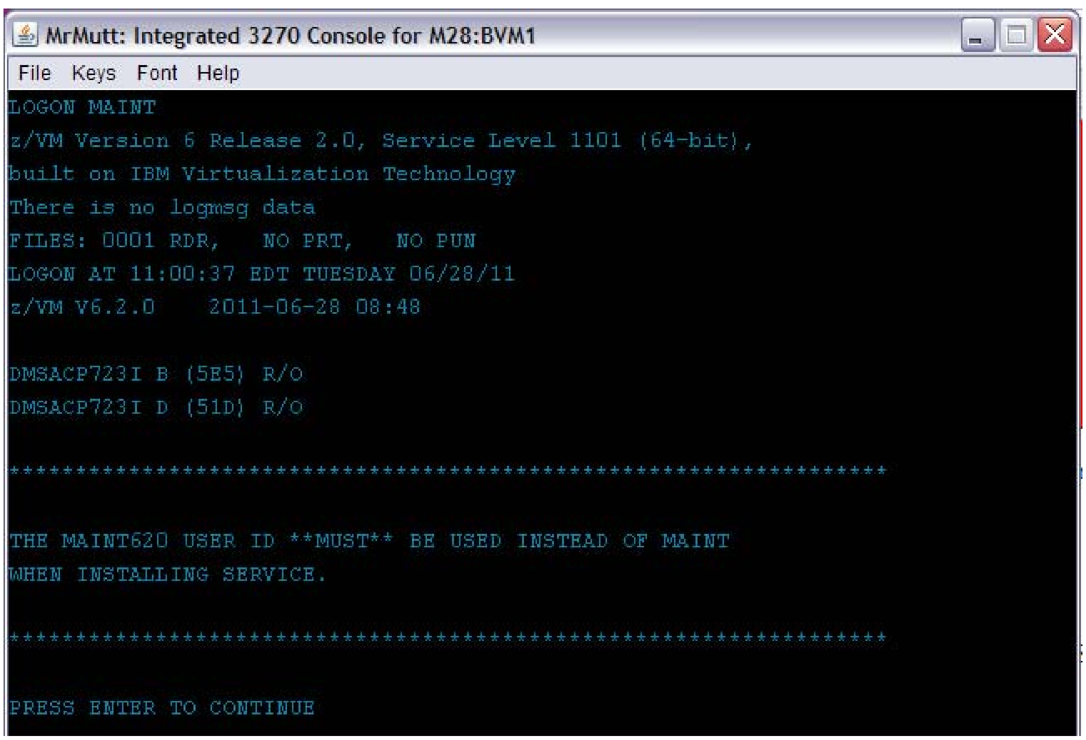


Figure 5-22 Logging on to MAINT and pressing Enter twice

- When IPLing CMS before z/VM 6.2, the **Enter** key had to be pressed when the status area in the lower right reads “`VM READ`”. Doing so allows the `PROFILE EXEC` to run. With z/VM 6.2, **Enter** must be pressed a second time on certain virtual machines such as `MAINT`.

5.4.1 Use the IPWIZARD tool

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

- Access the MAINT 193 disk:

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

- Invoke **IPWIZARD**.

```
==> ipwizard
```

You should see the screen similar to what is shown in Figure 5-23 on page 62.

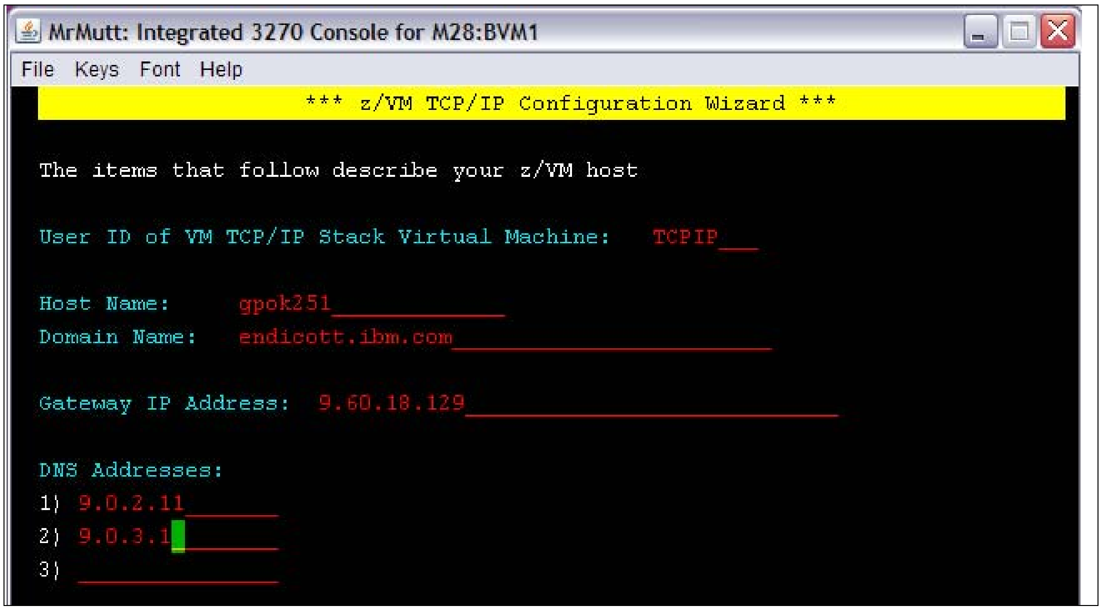


Figure 5-23 IPWIZARD screen 1

- The *z/VM TCP/IP Configuration Wizard* opens as shown in the preceding example. The first field, User ID, should always be **TCPIP**. Obtain the remaining values from the 2.8.2, "z/VM resources worksheet" on page 19 and press **F8**.

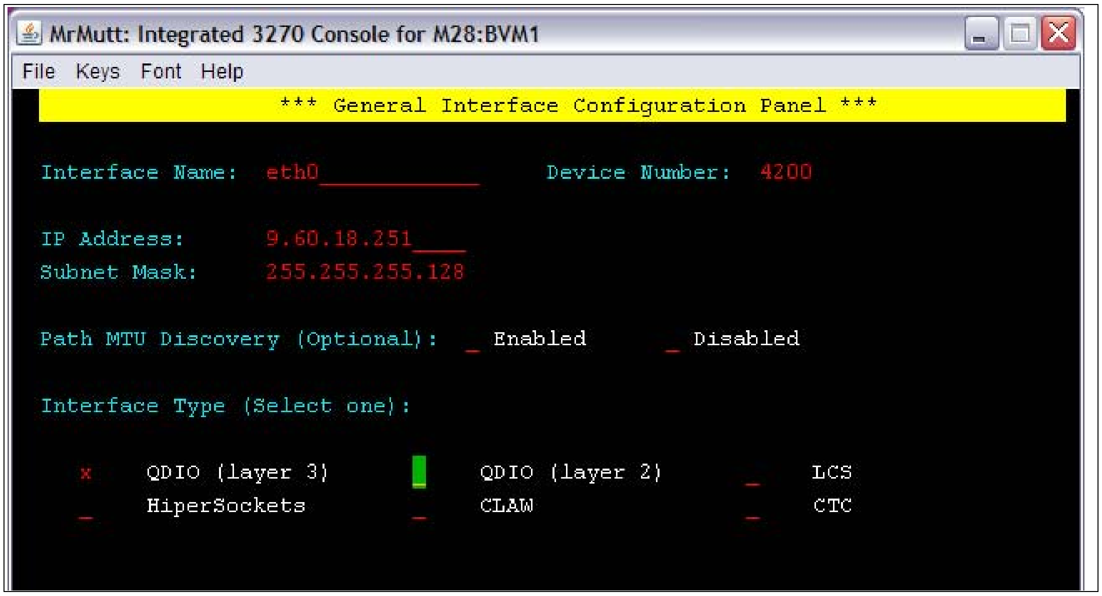


Figure 5-24 IPWIZARD screen 2

- 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* which 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 completed, press **F8**.

  Note: to utilize QDIO (layer 2), certain prerequisites must be met. Consult with the system administrator.

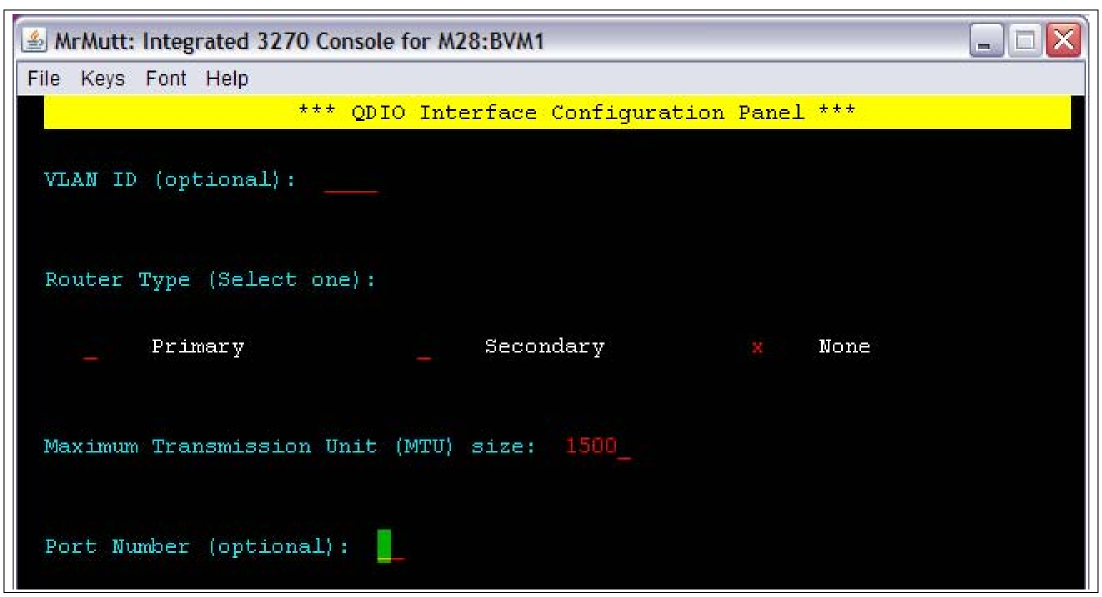


Figure 5-25 IPWIZARD screen 3

- In general, a value for the *Port Name* is no longer necessary. Press **F5** to complete the wizard.

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

- Enter **1** to restart the TCP/IP stack (you may see other warnings). Watch for the message `HCPINP8392I IPWIZARD EXEC ENDED SUCCESSFULLY`:

```
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 = 10 FORCED BY MAINT
...
Successfully PINGed Interface (9.60.18.251)
Successfully PINGed Gateway (9.60.18.129)
Successfully PINGed DNS (9.0.2.11)
Successfully PINGed DNS (9.0.3.1)
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
```

- At this point your z/VM TCP/IP stack should be up. You should now 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.
- **Log off** of `MAINT` so the `PMAINT 2CC` disk is freed up.
- **Repeat the previous steps** for all other SSI members. When you run **IPWIZARD** on the other members, you should find that the network information is remembered.

All members of the SSI cluster should now be on the network.

HMC Integrated 3270 Console or 3270 emulator? At this point, your SSI members should be accessible over the network. You can continue working at the HMC, or you can access your new system using a 3270 emulator. See 3.1.3, "3270 emulators" on page 30 for some brief words on that subject.

To switch to a 3270 emulator, first **LOGOFF** of `MAINT` from the Integrated 3270 Console, but you could also **DISCONNECT**. 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 a PC's monitor off. In general, you should **LOGOFF** of system administration virtual machines such as `MAINT`. However, you should always **DISCONNECT** from z/VM service machines such as `TCPIP` and virtual machines running Linux. Logging off of them will terminate the service or crash Linux.

5.5 Configure the XEDIT PROFILE

The **XEDIT** command looks for the file `XEDIT PROFILE` configuration file when it is invoked. Not all CMS virtual machines have a copy of this file, so XEDIT sessions look and behave differently. 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, there is a cheat sheet in Appendix A.4.1, "XEDIT cheat sheet" on page 375. The z/VM 6.2 PDF library is on the Web at:

http://www-03.ibm.com/systems/z/os/zos/bkserv/zvmpdf/#zvm62

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

To configure the XEDIT profile on the SSI cluster, perform the following steps:

- Logon to MAINT on the first SSI member if you are not already.
- 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. It is recommended 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
  ```

  Before:

  ```
  SET PF12 FILE
  ```

  After:

  ```
  SET PF12 ?
  ```

- Save your changes with the **FILE** subcommand.
- Make the modified file available to other virtual machines by copying it to the MAINT 19E disk with file mode suffix 2:
  - Release the current 19E disk:

    ```
    ==> rel 19e
    ```

  - Link to the MAINT 19E disk read-write:

    ```
    ==> link * 19e 19e mr
    DASD 019E LINKED R/W; R/O BY    10 USERS
    ```

  - Access the disk as file mode F:

    ```
    ==> acc 19e f
    ```

  - Copy it to the MAINT 19E disk (F) with file mode suffix 2 (because the MAINT 19E disk is commonly accessed with a file mode suffix of 2, files will not be seen by other virtual machines unless they have this file mode suffix):

    ```
    ==> copy profile xedit a = = f2
    ```

  - Save the CMS named saved segment:

    ```
    ==> acc 193 g
    ==> sampnss cms
    HCPNSD440I The Named Saved System (NSS) CMS was successfully defined in fileid 0
    002.
    ==> ipl 190 parm savesys cms
    HCPNSS440I Named Saved System (NSS) CMS was successfully saved in fileid 0002.
    ```

- **Repeat the previous steps** on all other members in the SSI cluster.

The same XEDIT PROFILE should now be accessible to all virtual machines in the SSI cluster.

5.6 Customize the SYSTEM CONFIG file

The first configuration file read when z/VM IPLs is the SYSTEM CONFIG file. There is only one SYSTEM CONFIG file per SSI cluster.

The following changes are recommended:

- Increase retrieve key capacity
- Allow virtual disks (VDISKs) to be created
- Turn off the Disconnect Timeout feature (this will prevent idle disconnected users from being forced off the system)

- Define virtual switches to be used for Linux networking

To make these changes, perform the following steps:

- Access the PMAINT CF0 disk read-write. Use the **LINK** command with multi-read (**MR**) parameter:

```
==> link pmaint cf0 cf0 mr
```

- Use the **ACCESS** command to get access it as your F disk:

```
==> acc cf0 f
```

- Make a backup copy of the vanilla SYSTEM CONFIG file using the **COPYFILE** command with the **OLDDATE** parameter so the timestamp 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
```

- Edit the original file:

```
==> x system config f
====> /features
```

- 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 virtual disks to be created by users by changing Userlim to **infinite** and by adding the **Syslim infinite** clause:

```
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
```

- For each member, set real device equivalency IDs for the OSA addresses to be used, and set the MAC address prefix. Real device mapping provides a means of identifying a device by an equivalency ID (EQID). This mapping ensures virtual machines relocated by LGR continue to use the same or equivalent devices following a relocation. Use the **BOTTOM** subcommand to go to the bottom of the file.

  The VMLAN MACPREFIX statement 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. In this example, ***02000B*** and ***02000C*** are used as lower values have already been assigned. the VMLAN TRANSIENT 0 statement prevents dynamic definition of Guest LANs by class G users.

Important: Regarding the setting of the of the `VMLAN MACPREFIX` value, the *CP Planning and Administration* manual states the following:

> "In an SSI cluster, system-defined locally administered MAC addresses are created using the prefix value specified on the `MACPREFIX` operand. The `MACPREFIX` value must be different for each member of the cluster. The default value is 02xxxx, where xxxx is the member's slot number on the SSI statement. If the MACPREFIX value is explicitly defined, the VMLAN statement must be qualified for the member to which it applies. Therefore if a VMLAN statement with the `MACPREFIX` operand is retained from the non-SSI system or created in this step, it must be qualified for member VMSYS01."

```
====> bot
====> a 14
/* Add EQID statements for OSA addresses and unique MAC IDs */
POKDEV62: begin
  rdev 4200-420f eqid osaset1 type osa
  rdev 4300-430f eqid osaset1 type osa
  vmlan macprefix 02000b
  vmlan limit transient 0
POKDEV62: end

POKTST62: begin
  rdev 4200-420f eqid osaset1 type osa
  rdev 4300-430f eqid osaset1 type osa
  vmlan macprefix 02000c
  vmlan limit transient 0
POKTST62: end
```

Important: The following step recommends creating both layer 3 (`IP`) and layer 2 (`ETHERNET`) VSWITCHes. z/VM development now recommends layer 2, especially as it is required for IPv6.

You may wish to define and use layer 2 VSWITCHes exclusively. If so, it is recommended that you still define a second VSWITCH, however, with no OSA devices. Such VSWITCHes can be used for private interconnecting networks. Following is an example:

```
/* Define VSWITCHes */
define vswitch vsw1 rdev 4203 4300 ethernet
define vswitch vsw2 ethernet
```

If you make this change, other examples in the book, which are noted, will have to change.

- Define a layer 2 and a layer 3 virtual switch with the `DEFINE VSWITCH` statements. Modify the two starting addresses of the OSA triplets to those that you specified in 2.8.2, "z/VM resources worksheet" on page 19:

```
====> bot
====> a 4

/* Define VSWITCHes: VSW1 - layer 3, VSW2 - layer 2 */
define vswitch vsw1 rdev 4203 4300 ip
define vswitch vsw2 rdev 4206 4303 ethernet
```

- Save your changes with the **`FILE`** subcommand:

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

- Test your changes with the **`CPSYNTAX`** command which is on the `MAINT 193` disk. It must be run once for each member of the SSI cluster using the **`LPAR`** option to the **`CPSYNTAX`** command:

```
==> acc 193 g
==> cpsyntax system config f (lpar bvm1
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
==> cpsyntax system config   f (lpar bvm2
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
```

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

- Release and detach the PMAINT CF0 disk with the **RELEASE** command:

```
==> rel f (det
DASD 0CF0 DETACHED
```

The SYSTEM CONFIG file should now be initially configured.

5.7 Configure additional network resources

The following changes are recommended to the system

- "Turn on the z/VM FTP server" on page 68
- "Shut down and re-IPL the SSI cluster" on page 70
- "Test changes" on page 72
- "Attach the z/VM TCP/IP stack to the VSWITCH" on page 69

The main TCP/IP configuration file is PROFILE TCPIP file is on the TCPMAINT 198 disk, which is accessed as the D disk.

5.7.1 Turn on the z/VM FTP server

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

- Make a backup copy of the TCP/IP configuration file:

```
==> copy profile tcpip d = tcpiorig = (oldd
```

- Edit the TCP/IP configuration file:

```
==> x profile tcpip d
```

- Add an AUTOLOG statement near the top of the file with FTPSERVE as the only entry. 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:

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
...
```

- Save your changes with the **FILE** subcommand:

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

- **Repeat the previous steps** on all other members of the SSI cluster.

You may choose to test a shutdown and re-IPL of the SSI cluster now (which is described after the next section), or you may choose to first attach the z/VM TCP/IP stack to the highly available VSWITCH, which is described in the next section.

5.7.2 Attach the z/VM TCP/IP stack to the VSWITCH

The z/VM TCP/IP stack was attached to a real OSA triplet using **IPWIZARD** to quickly get the new z/VM system on the network. Then a highly available VSWITCH was created with failover OSA addressees. It is recommended that the z/VM stack now be attached to the VSWITCH.

To do this, perform the following steps:

- Log on to TCPMAINT, if you are not logged on already.
- Edit the SYSTEM DTCPARMS file on the TCPMAINT 198 (D) disk. Comment out the last line which causes the real OSA triplet to be attached to the TCPIP virtual machine:

```
==> x system dtcparms d
.***********************************************************************
.* SYSTEM DTCPARMS created by DTCIPWIZ EXEC on 19 Apr 2012
.* Configuration program run by MAINT at 09:56:59
.***********************************************************************
:nick.TCPIP    :type.server
               :class.stack
.*             :attach.4003-4005
```

  This change will prevent three OSA devices from being attached to TCPIP as it is started.

- Make a backup copy of the working PROFILE TCPIP file created by the **IPWIZARD**:

```
==> copy profile tcpip d = tcpiwrks =
```

- Edit the PROFILE TCPIP file on the TCPMAINT 198 (D) disk. Change the real OSA starting address (4003 in this example) to the virtual starting address (0600) everywhere in the file:

```
==> x profile tcpip d
====> c/4003/0600/* *
DMSXCG517I 4 occurrence(s) changed on 3 line(s)
```

  This will instruct TCPIP to use the virtual NIC starting at virtual device number 600.

- Search for the DEVICE statement. In the **LINK** statement below it, change IP to **ETHERNET**:

```
====> /device
```

```
DEVICE DEV@0600  OSD 0600   NONROUTER
LINK ETH0 QDIOETHERNET DEV@0600   MTU 1500 ETHERNET
; (End DEVICE and LINK statements)
; ----------------------------------------------------------------------
; ----------------------------------------------------------------------
HOME
...
```

This will allow the TCPIP stack to attach to a layer 2 (ETHERNET) VSWITCH.

- Log off of TCPMAINT.
- Log on to MAINT.
- Make a copy of the original USER DIRECT file on the MAINT 2CC (C) disk:

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

- Edit the USER DIRECT file and locate the TCPIP definition. Add the following three lines that will connect the virtual machine to the VSWITCH VSW1 at logon time:

```
==> x user direct c
====> /identity tcpip
...
IDENTITY TCPIP    TCPIP     128M  256M ABG
INCLUDE TCPCMSU
BUILD ON * USING SUBCONFIG TCPIP-1
* BUILD ON @@member2name USING SUBCONFIG TCPIP-2
* BUILD ON @@member3name USING SUBCONFIG TCPIP-3
* BUILD ON @@member4name USING SUBCONFIG TCPIP-4
 COMMAND SET VSWITCH VSW1 GRANT &USERID
 COMMAND DEFINE NIC 600 TYPE QDIO
 COMMAND COUPLE 600 TO SYSTEM VSW1
...
====> file
```

These statements will grant TCPIP access to VSWITCH VSW1, will define a virtual NIC starting at virtual device address 600, and will couple it to the VSWITCH.

- Run the **DIRECTXA** command to put the changes online:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 6 RELEASE 2.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 104 disk pages
```

- For all other members in the SSI, repeat the changes to PROFILE TCPIP, and also run **DIRECTXA** (no additional changes are necessary to the USER DIRECT file as the change was done in the IDENTITY section, not down in the SUBCONFIGs).

The z/VM TCP/IP stack should come up on the highly available VSWTICH the next time z/VM is IPLed.

5.7.3 Shut down and re-IPL the SSI cluster

You can watch the z/VM member shut down and Re-IPL from the *Integrated 3270 Console*. If you issue this command from a 3270 emulator, you will lose your session and will not see most of the shut down process. To shut down and re-IPL the SSI cluster, perform the following steps:

- Log Off of MAINT on all 3270 emulator sessions.
- Start an *Integrated 3270 Console* session for the LPAR of the first SSI cluster member.
- Logon to MAINT.

- Using the **AT** command, issue the **SHUTDOWN** command for member 2. In this example, the system name is ***POKTST62***:

```
==> at poktst62 cmd shutdown
...
```

- If you have more than two members, repeat the previous **AT ... SHUTDOWN** step for those members.
- From the HMC, start an *Integrated 3270 Console* session for the first member.
- From member 1 of the cluster, issue the **SHUTDOWN REIPL** command:

```
==> shutdown reipl
...
```

  All members of the SSI cluster should now be down, and member 1 should be coming back up.

- Logon to MAINT.
- By default, the TCPIP service virtual machine is not logged on (that will be customized later). Start TCP/IP with the **XAUTOLOG** command:

```
==> xautolog tcpip
Command accepted
AUTO LOGON  ***       EREP     USERS = 12
HCPCLS6056I XAUTOLOG information for TCPIP: The IPL command is verified by the I
PL command processor.
```

- Try starting a 3270 emulator session to member 1. You should see a logon panel. If not, you will have to debug the problem from the *Integrated 3270 Console* session. For example, you could FORCE TCPIP and logon to TCPIP interactively and watch for error messages.
- Verify that TCPIP is attached with the **QUERY VSWITCH** with **DETAILS** command:

```
==> q vswitch vsw1 det
VSWITCH SYSTEM VSW1     Type: QDIO    Connected: 1    Maxconn: INFINITE
  PERSISTENT  RESTRICTED    ETHERNET                  Accounting: OFF
  USERBASED
  VLAN Unaware
  MAC address: 02-00-0E-00-00-01    MAC Protection: Unspecified
  State: Ready
  IPTimeout: 5         QueueStorage: 8
  Isolation Status: OFF
 Uplink Port:
  RDEV: 7100.P00 VDEV: 0600 Controller: DTCVSW2
        EQID: OSASET1
    Uplink Port Connection:
      RX Packets: 45         Discarded: 0          Errors: 0
      TX Packets: 82         Discarded: 0          Errors: 0
      RX Bytes: 3330                 TX Bytes: 12478
      Device: 0600  Unit: 000   Role: DATA       Port: 2049
 Adapter Connections:
    Adapter Owner: TCPIP    NIC: 0600.P00 Name: UNASSIGNED  Type: QDIO
      RX Packets: 5044       Discarded: 0          Errors: 0
      TX Packets: 82         Discarded: 0          Errors: 0
      RX Bytes: 220405               TX Bytes: 12478
      Device: 0600  Unit: 000   Role: DATA       Port: 0003
      Options: Ethernet Broadcast
        Unicast MAC Addresses:
          02-00-0E-00-00-05 IP: 9.60.18.144
        Multicast MAC Addresses:
          01-00-5E-00-00-01
```

You should now have member 1 back up with TCPIP attached to the highly available VSWITCH and the FTP server running.

IPL the other SSI members

- Go to the HMC and start an *Integrated 3270 Console* for the second SSI member
- IPL the LPAR with the **Load** task.
- Go to the *Integrated 3270 Console* and complete the IPL of z/VM as done previously (**F10** at the SAPL, **cold drain**, **no**, then **go**).
- Disconnect from `OPERATOR` on the *Integrated 3270 Console*.
- Logon as `MAINT`.
- Start `TCPIP` with the **`XAUTOLOG`** command:

```
==> xautolog tcpip
Command accepted
...
```

- If you have more than two members, repeat the previous step for those members.
- Verify the other node(s) in the cluster can be accessed through the highly available VSWITCH.

The SSI cluster should now be back up.

5.7.4 Test changes

To test the changes you have made, perform the following steps:

- Start a 3270 emulator session to the first SSI member
- Logon as `MAINT`.
- Issue the **`QUERY SSI`** command:

```
==> q ssi
SSI Name: POKSSI
SSI Mode: Stable
Cross-System Timeouts: Enabled
SSI Persistent Data Record (PDR) device: CV6284 on 6284
SLOT SYSTEMID STATE     PDR HEARTBEAT       RECEIVED HEARTBEAT
   1 POKDEV62 Joined    08/28/11   08:14:05 08/28/11   08:14:05
   2 POKTST62 Joined    08/28/11   08:13:58 08/28/11   08:13:58
   3 -------- Available
   4 -------- Available
```

- Use the **`QUERY RETRIEVE`** and **`QUERY VDISK`** 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
```

- Try starting an FTP session to all of the SSI members. You should get a logon prompt.

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

5.8 Add page and perm volumes

Each z/VM 6.2 SSI member is installed with one paging volume and one spool volume, being either 3390-3s or 3390-9s, depending on which type of disks the cluster was installed onto. One spool volume per member is probably adequate for Linux needs, however, more paging volumes are recommended.

Page and temporary disk volumes are not shared.

It is recommended that you add at least four 3390-3 paging volumes so you will have a total of five. Having adequate paging space will give you *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 virtual machines combined. A second rule of thumb is to never allow your z/VM system's paging space to go above 50% utilized.

5.8.1 Format volumes for page space

Before adding paging volumes to the SSI cluster members, 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** has been provided to allow you to format many volumes with a single command. The source code for this EXEC is in the section B.2.2, "The CPFORMAT EXEC" on page 380. 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 2.3.1, "Volume labeling convention" on page 10. 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.

5.8.2 Copy the CPFORMAT EXEC to the members

Perform the following steps:

- Log off of `MAINT` on the current member so you will be able to get the `MAINT 191` disk in read-write mode using FTP.
- Start an SSH session to the PC NFS server and **cd** to the `/var/nfs/CKB-VM62/vm/maint/` directory which was created when you untarred the files associated with this book:

```
# cd /var/nfs/CKB-VM62/vm/maint
```

- List the files for the `MAINT 191` disk:

```
# ls
callsm1.exec  chpw620.xedit  cpformat.exec  ssicmd.exec  ssishutd.exec
```

- Start an FTP session to the first SSI member as `MAINT`. If you get a reply from the FTP server it shows that it is configured correctly. Issue the **MPUT** subcommand to copy all files.

```
# ftp 9.60.18.251
Name (9.12.5.22:root): maint
331-Password: maint
230-MAINT logged in; working directory = MAINT 191
...
```

```
ftp> mput *
mput callsm1.exec [anpqy?]? a
Prompting off for duration of mput.
...
ftp> quit
```

You should now have the **CPFORMAT EXEC**, and all other necessary files on the MAINT 191 disk.

- **Repeat the previous steps** for all other members in the cluster.

5.8.3 Use the CPFORMAT EXEC

To use the **CPFORMAT** EXEC, perform the following steps:

- **Log into MAINT** on the first member. You should now have access to the CPFORMAT EXEC. Edit the file to set the first character that will be used in labels. Look for the variable **firstChar** which defaults to **'**J**'**. It is recommended that you choose a unique character for each member in the SSI cluster. In this example, the **firstChar** variable was set to **'**W**'** for the second member.

```
==> x cpformat exec
/************************************************************/
...
  Address COMMAND
  firstChar = 'J'
...
```

- 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 J<t><xxxx> where:
    <t> is type - P (page), M (perm), S (spool) or T (Temp disk)
    <xxxx> is the 4 digit address

Syntax is:
                <---------------<
   >>--CPFORMAT--.-vdev--------.--AS---.-PERM-.---------><
                 '-vdev1-vdev2-'       '-PAGE-'
                                       '-SPOL-'
                                       '-TEMP-'
```

The following example shows how to attach four 3390-3 volumes and use **CPFORMAT** to format them as paging space. Refer to the planning work sheets that you should have filled out in section 2.8.3, "z/VM DASD worksheet" on page 21:

- The DASD that will be used for paging volumes on member 1 in this example are at real device addresses ***6232*, *6233*, 6288** and ***628A*.** Query the DASD devices to see their status:

```
==> q 6232 6233 6288 628a
DASD 6232 FR6232  , DASD 6233 FR6233  , DASD 6288 FR6288  , DASD 628A FR628A
```

- Attach the devices to MAINT (the last parameter of * means the current virtual machine) using the **ATTACH** command:

```
==> att 6232 6233 6288 628a *
6232 6233 6288 628A ATTACHED TO MAINT
```

- Use the **CPFORMAT** command with the **AS PAGE** parameter:

```
==> cpformat 6232 6233 6288 628a as page
```

```
Format the following DASD:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    6232 MAINT    6232 3390  FR6232 6232          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    6233 MAINT    6233 3390  FR6233 6233          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    6288 MAINT    6288 3390  FR6288 6288          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    628A MAINT    628A 3390  FR628A 628A          0       3339

WARNING - this will destroy data!
ARE YOU SURE you want to format the DASD as PAGE space (y/n)?
y
...
DASD status after:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    6232 MAINT    6232 3390  JP6232 6232          0       3339
MAINT    6233 MAINT    6233 3390  JP6233 6233          0       3339
MAINT    6288 MAINT    6288 3390  JP6288 6288          0       3339
MAINT    628A MAINT    628A 3390  JP628A 628A          0       3339
```

This formatting job should run for about 10-30 minutes depending on many factors.

- **Repeat the previous steps** for all remaining SSI members. in this example, four different volumes are formatted for page space on member 2. Verify that you have the **firstChar** variable set in CPFORMAT EXEC if you want different first characters.

```
==> cpformat 628b 633b 633c 633e as page
...
DASD status after:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    628B MAINT    628B 3390  WP628B 628B          0       3339
MAINT    633B MAINT    633B 3390  WP633B 633B          0       3339
MAINT    633C MAINT    633C 3390  WP633C 633C          0       3339
MAINT    633E MAINT    633E 3390  WP633E 633E          0       3339
```

5.8.4 Format DASD for minidisks

In addition to CP disks such as page space, system disks will be needed to create minidisks for the virtual machines. In this section the DASD which will be used for virtual machine minidisks will be formatted.

- Start a 3270 session as MAINT on the first SSI cluster member.
- Query the DASD that will be used for minidisks. In this example they are ***6289***, ***628C-6290***, ***6293***, ***6294***, ***6327***, ***6328***, ***6339***, ***633A*** (3390-3s), ***61A5*** and ***61B2*** (3390-9s):

```
==> q 6289 628c-6290 6293 6294 61a5 61b2 6327 6328 6339 633a
DASD 6289 FR6289  , DASD 628C FR628C  , DASD 628D FR628D  , DASD 628E FR628E
DASD 628F FR628F  , DASD 6290 FR6290  , DASD 6293 FR6293  , DASD 6294 FR6294
DASD 61A5 FR61A5  , DASD 61B2 FR61B2  , DASD 6327 FR6327  , DASD 6328 FR6328
DASD 6339 FR6339  , DASD 633A FR633A
```

- Attach the volumes:

```
==> att 6289 628c-6290 6293 6294 61a5 61b2 6327 6328 6339 633a *
DASD 6289 ATTACHED TO MAINT 6289 WITH DEVCTL
628C-6290 ATTACHED TO MAINT
6293 6294 61A5 61B2 6327 6328 6339 633A ATTACHED TO MAINT
```

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

```
==> cpformat 6289 628c-6290 6293 6294 61a5 61b2 6327 6328 6339 633a as perm
```

```
Format the following DASD:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    6289 MAINT    6289 3390  FR6289 6289          0       3339
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    628C MAINT    628C 3390  FR628C 628C          0       3339
...
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    633A MAINT    633A 3390  FR633A 633A          0       3339

WARNING - this will destroy data!
ARE YOU SURE you want to format the DASD as PERM  space (y/n)?
...
DASD status after:
TargetID Tdev OwnerID  Odev Dtype Vol-ID Rdev   StartLoc       Size
MAINT    6289 MAINT    6289 3390  JM6289 6289          0       3339
MAINT    628C MAINT    628C 3390  JM628C 628C          0       3339
MAINT    628D MAINT    628D 3390  JM628D 628D          0       3339
MAINT    628E MAINT    628E 3390  JM628E 628E          0       3339
MAINT    628F MAINT    628F 3390  JM628F 628F          0       3339
MAINT    6290 MAINT    6290 3390  JM6290 6290          0       3339
MAINT    6293 MAINT    6293 3390  JM6293 6293          0       3339
MAINT    6294 MAINT    6294 3390  JM6294 6294          0       3339
MAINT    61A5 MAINT    61A5 3390  JM61A5 61A5          0      10017
MAINT    61B2 MAINT    61B2 3390  JM61B2 61B2          0      10017
MAINT    6327 MAINT    6327 3390  JM6327 6327          0       3339
MAINT    6328 MAINT    6328 3390  JM6328 6328          0       3339
MAINT    6339 MAINT    633A 3390  JM6339 633A          0       3339
MAINT    633A MAINT    633A 3390  JM633A 633A          0       3339
```

You should now have many volumes that can be used for minidisks. Note that the labels are prefixed with **JM** in this example.

5.8.5 Update the SYSTEM CONFIG file

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

- Logon to MAINT on member 1.
- Link as read-write and access the PMAINT CF0 disk:

```
==> link pmaint cf0 cf0 mr
==> acc cf0 f
```

- Make a copy of the working SYSTEM CONFIG file using the “WRKS” (it works!) suffix convention:

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

- 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
====> /page and
```

Before:

```
/* Page and Tdisk volumes for Member 1                                */
/**********************************************************************/

    POKDEV62:   BEGIN
                CP_Owned   Slot 255  JP6282
```

```
    POKDEV62:   END

/*********************************************************************/
/* Page and Tdisk volumes for Member 2                               */
/*********************************************************************/

    POKTST62:   BEGIN
                CP_Owned   Slot 255  WP639D
    POKTST62:   END
...
```

After:

```
    POKDEV62:   BEGIN
                CP_Owned   Slot 251  JP628A
                CP_Owned   Slot 252  JP6288
                CP_Owned   Slot 253  JP6233
                CP_Owned   Slot 254  JP6232
                CP_Owned   Slot 255  JV6282
    POKDEV62:   END

/*********************************************************************/
/* Page and Tdisk volumes for Member 2                               */
/*********************************************************************/

    POKTST62:   BEGIN
                CP_Owned   Slot 251  WP633E
                CP_Owned   Slot 252  WP633C
                CP_Owned   Slot 253  WP633B
                CP_Owned   Slot 254  WP628B
                CP_Owned   Slot 255  WV639D
    POKTST62:   END
```

- 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`** and no other LPAR will be using the same volumes with the same prefix, then add the following single line to include all PERM space as volume labels all begin with `JM6`.

```
====> /user_v
/*                         User_Volume_List                          */
/*********************************************************************/
/* These volumes contain the minidisks for your guests, as well as   */
/* the product disks for each installed release of z/VM in the SSI   */
/* cluster.  Volumes that hold "local" minidisks, i.e., minidisks    */
/* unique to a single member system, should be wrapped in BEGIN/END  */
/* statement.  If it becomes necessary to access a local minidisk    */
/* from a different member of the SSI cluster operating in REPAIR    */
/* mode, simply ATTACH the volume to SYSTEM.                         */
/*********************************************************************/

/*********************************************************************/
/* Shared User Volumes                                               */
/*********************************************************************/
               User_Volume_List  CV6285 CV6286 CV6287
               User_Volume_Include JM6*
...
====> file
```

Important: If other z/VM LPARs might be attaching volumes with the JM prefix, you should specifically list each volume to be attached to `SYSTEM` using the `User_Volume_List` statement. This will prevent the possibility of multiple z/VM systems writing to the same volume. In this example, the list would be:

```
User_Volume_List JM6289
User_Volume_List JM628C
User_Volume_List JM628D
User_Volume_List JM628E
User_Volume_List JM628F
...
```

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

```
==> acc 193 g
==> cpsyntax system config f (lpar bvm1
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
==> cpsyntax system config f (lpar bvm2
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
```

- When you have confirm there are no syntax errors, you can release and detach the `PMAINT CF0`:

```
==> rel f (det
DASD 0CF0 DETACHED
```

You should now have volumes formatted for paging and minidisks.

5.8.6 Attach minidisk volumes to system

You could do a **SHUTDOWN** of the system to test your changes, or you can proceed, If you proceed, attach the volumes for minidisks to `SYSTEM`.

```
==> det 6289 628c-6290 6293 6294 61a5 61b2 6327 6328 6339 633a
6289 628C-6290 6293 6294 61A5 61B2 6327 6328 6339 633A DETACHED
==> att 6289 628c-6290 6293 6294 61a5 61b2 6327 6328 6339 633a system
DASD 6289 ATTACHED TO SYSTEM JM6289
...
```

The newly formatted volumes should now be attached to `SYSTEM` and available for use as minidisks.

5.9 Create a virtual machine for common files

Now it is time to define your first z/VM virtual machine, `LNXMAINT`. It will be used to store files that will be shared by Linux virtual machines.

5.9.1 Define the user in the USER DIRECT file

A small 20 cylinder minidisk is allocated at virtual address `191` and a larger 500 cylinder minidisk (approximately 350 MB), to be shared by many guests, is defined at virtual address `192`. Use the next free DASD designated as PERM space on your worksheet (2.8.3, "z/VM DASD worksheet" on page 21). In this example, it is ***JM6289***. Cylinder 0 should always be reserved for the label therefore you should start minidisks at cylinder 1.

- Make a copy of the original USER DIRECT file:

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

- Edit the USER DIRECT file and add the following virtual machine definition to the bottom of the file. A comment is added signifying the split between z/VM system virtual machines and locally defined virtual machines (this can be helpful when moving to a new version of z/VM):

```
==> x user direct c
====> bottom
====> a 9
...
*-----------------------------------------------------------
* z/VM system virtual machines are above, user defined below
*-----------------------------------------------------------
USER LNXMAINT LNXMAINT 64M 128M BEG       1
 INCLUDE TCPCMSU                          2
 LINK TCPMAINT 592 592 RR                 3
 MDISK 0191 3390 0001 0020 JM6289 MR READ     WRITE    MULTIPLE   4
 MDISK 0192 3390 0021 0500 JM6289 MR ALL      WRITE    MULTIPLE   5
*                                                                 6
====> file
```

Note the following points for the numbers in black:

1 User ID LNXMAINT, same password, default size of 64MB, 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 JM6289

5 Define 192 minidisk of size 500 cylinders (approximately 350 MB) with the special read password of **ALL** which allows read access from any virtual machine without a disk password

6 An empty comment line for better readability

- Whenever an MDISK statement is added or modified in the USER DIRECT file you should always check for overlapping cylinders and gaps (gaps will only leave empty disk space, however, overlaps can occur because z/VM will allow you to *shoot yourself in the foot* by defining multiple minidisks over the same disk space). This is done with 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.
```

- 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
```

- At the top of the file, you should see two overlap records in the $$$$$$ volume. This is a dummy volume, so it is not a true overlap. Search for all other overlaps with the **ALL** subcommand:

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

```
         DATAMOV2    5F0     3380       00501       00501       00001 *OVERLAP*
--------------------  1  line(s) not displayed --------------------
         DATAMOV2    5FF     3380       00502       00502       00001 *OVERLAP*
--------------------  508  line(s) not displayed --------------------
```

You should see these two overlaps, which are expected.

- Type **ALL** with no argument again to get out of this mode

```
====> all
```

- Now search for all the gaps using the **ALL** subcommand. You should also see some gaps:

```
====> all /gap
                                            0         500         501    GAP
--------------------  8  line(s) not displayed --------------------
                                            0           0           1    GAP
--------------------  5  line(s) not displayed --------------------
                                            0           0           1    GAP
--------------------  198  line(s) not displayed --------------------
                                            0           0           1    GAP
--------------------  298  line(s) not displayed --------------------

====> all
```

Three GAPs should be listed on the right side:

- 501 cylinders on the $$$$$$ volume
- 1 cylinder on the $$LNX1 volume
- 1 cylinder on the $$LNX2 volume
- 1 cylinder on volume used for LNXMAINT 191 and 192 disks (JM6289 in this example)

You don't have to worry about the first three gaps as they are expected given the layout of the default USER DIRECT file. To avoid a 1 cylinder gap being reported on each user volume, it is recommended to use the virtual machine $ALLOC$. This user is set to NOLOG which means it can never be logged onto. Thus it is not a conventional virtual machine, rather, it is a convenient place to put dummy minidisk definitions for cylinder 0 of all PERM volumes.

- Get out of the file USER DISKMAP with the **QUIT** command or by pressing **F3**.
- Edit the USER DIRECT file again and add new minidisk definitions at virtual address A08-A15 for the first cylinder of each DASD volume you added. In this example, there are 14:

```
==> x user direct
====> /user $alloc
USER $ALLOC$  NOLOG
 MDISK A00 3390 000 001 CV6284 R
 MDISK A01 3390 000 001 CV6285 R
 MDISK A02 3390 000 001 CV6286 R
 MDISK A03 3390 000 001 CV6287 R
 MDISK A04 3390 000 001 JV6280 R
 MDISK A05 3390 000 001 JV6283 R
 MDISK A06 3390 000 001 WV639B R
 MDISK A07 3390 000 001 WV639E R
 MDISK A08 3390 000 001 JM6289 R
 MDISK A09 3390 000 001 JM628C R
 MDISK A0A 3390 000 001 JM628D R
 MDISK A0B 3390 000 001 JM628E R
 MDISK A0C 3390 000 001 JM628F R
 MDISK A0D 3390 000 001 JM6290 R
 MDISK A0E 3390 000 001 JM6293 R
 MDISK A0F 3390 000 001 JM6294 R
 MDISK A10 3390 000 001 JM61A5 R
 MDISK A11 3390 000 001 JM61B2 R
```

```
 MDISK A12 3390 000 001 JM6327 R
 MDISK A13 3390 000 001 JM6328 R
 MDISK A14 3390 000 001 JM6339 R
 MDISK A15 3390 000 001 JM633A R
```

- Save your changes with the **FILE** subcommand 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 `$ALLOC$` virtual machine, you should see the disk map of the volume you added for `LNXMAINT`:

```
==> 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.
==> x user diskmap
====> prefix off
====> all /gap
                                            0         500         501    GAP
--------------------  8  line(s) not displayed --------------------
                                            0           0           1    GAP
--------------------  5  line(s) not displayed --------------------
                                            0           0           1    GAP
--------------------  497  line(s) not displayed --------------------
```

- Quit XEDIT by pressing **F3**.

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

- 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 6 RELEASE 2.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 105 disk pages
```

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

- Log Off of `MAINT` on member 1 to free up the `MAINT 2CC` disk.
- Logon to `MAINT` on member 2. Run the **DIRECTXA** command on this member, then query the virtual machine again.

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 6 RELEASE 2.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 105 disk pages
==> q lnxmaint
HCPCQU045E LNXMAINT not logged on
```

- If you have more than two members, repeat the **DIRECTXA** step for all other members.

You have now defined your first z/VM virtual machine named `LNXMAINT` and brought it online to both SSI members.

5.9.2 Format LNXMAINT’s minidisks

Now you should be able to logon to the new virtual machine and format its two minidisks.

- Log off of `MAINT`.
- Log on to `LNXMAINT`.

```
LOGON LNXMAINT
```

```
z/VM Version 6 Release 2.0, Service Level 1101 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 17:32:38 EDT WEDNESDAY 07/27/11
z/VM V6.2.0    2011-06-28 08:48

DMSACP112S A(191) device error
```

You should see an error message ending in "`device error`". 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 virtual machine, however, it has never been formatted as a CMS file system.

- 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** 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)
```

- 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 500 cylinders formatted on D(192)
```

- You have now 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
LNX191 191  A   R/W    20 3390 4096        0          7-00      3593       3600
LXM192 192  D   R/W   500 3390 4096        0         13-00     89987      90000
MNT190 190  S   R/O   207 3390 4096      702      18924-51     18336      37260
MNT19E 19E  Y/S R/O   500 3390 4096     1093      30301-34     59699      90000
```

5.9.3 Create a PROFILE EXEC

Create a simple **PROFILE EXEC** that will be run each time this virtual machine is logged on.

- Create the new file using XEDIT and add the following lines (be sure to type the **A** file mode so you don't 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.

- You could test your changes by logging off and logging back on. However, typing the command **PROFILE** will do the same.

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

- 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. Verify these three disks are accessed with 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        2           9-01      3591       3600
LXM192 192  D   R/W   500 3390 4096        1          14-01     89986      90000
TCM592 592  E   R/O   140 3390 4096      882       14787-59     10413      25200
MNT190 190  S   R/O   207 3390 4096      702       18924-51     18336      37260
MNT19E 19E  Y/S R/O   500 3390 4096     1093       30301-34     59699      90000
```

- 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
...
```

5.9.4 Copy files associated with this book

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. Perform the following steps:

- **Log off of 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:

```
# cd /var/nfs/CKB-VM62/vm/lnxmaint
```

- List the files for the LNXMAINT 192 disk:

```
# ls
profile.exec  sample.conf-rh6  sample.parm-s11  swapgen.exec
rhel62.exec   sample.parm-rh6  sles11s2.exec
```

- 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 all in ASCII and the default behavior is to convert to ASCII to EBCDIC. Use the **mput *** subcommand to copy the files from the vm/ directory to LNXMAINT:

```
# ftp 9.60.18.249
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> mput *
...
ftp> quit
```

- Logon to LNXMAINT.
- Use the **FILELIST** command to show the files on the D disk:

```
==> 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     Time
      PROFILE  EXEC     D1 V         70         52          1 11/16/11 11:16:17
      RHEL62   EXEC     D1 V         63         44          1 11/16/11 11:16:17
      SAMPLE   CONF-RH6 D1 V         38         13          1 11/16/11 11:16:17
      SAMPLE   PARM-RH6 D1 V         80          3          1 11/16/11 11:16:17
      SAMPLE   PARM-S11 D1 V         69         11          1 11/16/11 11:16:17
      SLES11S2 EXEC     D1 V         72         10          1 11/16/11 11:16:17
      SWAPGEN  EXEC     D1 V         72        467          6 11/16/11 11:16:17
      PROFILE  XEDIT    D1 V         45         17          1 11/15/11 17:16:38
```

You should now have copied the files associated with this book to LNXMAINT 192.

5.10 Configure AUTOLOG1's PROFILE EXEC

When z/VM IPLs, normally the AUTOLOG1 virtual machine is logged on (unless the NOAUTOLOG parameter is specified at IPL). Its PROFILE EXEC is run when CMS IPLs. It is recommended that the following tasks be accomplished using this file:

- Configure Linux to shut down gracefully using the **SET SIGNAL** command
- Limit in central storage and turn it off in expanded storage
- Start virtual machines that should be started using the **XAUTOLOG** command

As AUTOLOG1 is now a multi-configuration virtual machine (IDENTITY), there is one virtual machine on each member. To configure the AUTOLOG1 PROFILE EXEC, perform the following steps:

- Log on to AUTOLOG1.
- Before pressing Enter at the VM READ prompt, type **acc (noprof** so that the **PROFILE EXEC** is not run.

```
LOGON AUTOLOG1
z/VM Version 6 Release 2.0, Service Level 1101 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 14:47:15 EDT FRIDAY 06/22/12
z/VM V6.2.0    2012-06-21 16:54
==> acc (noprof
```

- Make a backup copy of the PROFILE EXEC:

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

- Edit the PROFILE EXEC and add a line so the virtual machine PERFSVM is started at z/VM IPL time:

```
==> x profile exec
====> /customer
...
/* Customer processing can be added here                            */
/*******************************************************************/
```

```
"CP XAUTOLOG TCPIP"             /* Autolog TCPIP                    */
"CP SET MDC STOR OM 256M"       /* Limit minidisk cache in CSTOR    */
"CP SET MDC XSTORE OM OM"       /* Disable minidisk cache in XSTOR  */
"CP SET SIGNAL SHUTDOWN 600"    /* Allow guests 10 min to shut down */
...
====> file
```

- For all other members in the SSI, perform the same steps.

The `PROFILE EXEC` on `AUTOLOG1 191` disk should be configured for all members in the SSI.

5.10.1 Shut down and re-IPL the SSI cluster

It is recommended that you again shut down and re-IPL to test the changes.

- Log on to `MAINT` on the first SSI member.
- Before you shut down, note that you have only one page volume (`JV6282` in this example) using the **`QUERY ALLOC PAGE`** command. A REXX EXEC has been provided to run any CP command on all members in the SSI cluster. It is named **`SSICMD EXEC`**. Use it to issue the **`QUERY ALLOC PAGE`** command across the SSI cluster:

```
==> ssicmd q alloc page
POKDEV62:
EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
JV6282 6282          1       3338 600840      0      0   0%
                                  ------ ------        ----
SUMMARY                           600840      0          0%
USABLE                            600840      0          0%

POKTST62:
EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
WP639D 639D          1       3338 600840      0      0   0%
                                  ------ ------        ----
SUMMARY                           600840      0          0%
USABLE                            600840      0          0%
```

Important: The **`SSICMD EXEC`** should not be used with asynchronous CP commands. The output may not be properly aligned with the member name prefix. Also, the AT command does not support commands that have an **`AT`** argument such as **`QUERY`**.

- It is time to shutdown and re-IPL the cluster again. In section 5.7.3, “Shut down and re-IPL the SSI cluster” on page 70, this task was accomplished manually. Such steps can become repetitive, especially for a 4-member SSI cluster. The **`SSISHUTD EXEC`** has been written to expedite this task. It can be called with no parameter to just shut down the cluster. It can also be called with the **`REIPL`** parameter to reboot the cluster.

  Issue this command now with the **`REIPL`** parameter:

```
==> ssishutd reipl
Are you sure you want to SHUTDOWN REIPL the SSI cluster? (y/n)
y
SYSTEM SHUTDOWN STARTED
HCPSHU960I System shutdown may be delayed for up to 630 seconds
VMSERVP : DMS5BC3108I Shutdown Signal received. STOP processing started
...
```

You will lose your 3270 emulator session(s). If you watch the HMC, the SSI member LPARs should immediately turn from white to green, then return to white after a minute or so.

- After the system comes back, logon as MAINT.
- Use the **SSICMD EXEC** again to issue the **QUERY ALLOC PAGE** command across the SSI cluster. You should now see that you have five paging volumes on each of the members:

```
==> ssicmd q alloc page
POKDEV62:
                EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
JP628A 628A          0       3338 601020      0      0   0%
JP6288 6288          0       3338 601020     12     12   1%
JP6233 6233          0       3338 601020      0      0   0%
JP6232 6232          0       3338 601020      0      0   0%
JP6282 6282          1       3338 600840      0      0   0%
                                  ------ ------        ----
SUMMARY                            2934K     12          1%
USABLE                             2934K     12          1%

POKTST62:
                EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
WP633E 633E          0       3338 601020      0      0   0%
WP633C 633C          0       3338 601020      0      0   0%
WP633B 633B          0       3338 601020      0      0   0%
WP628B 628B          0       3338 601020      0      0   0%
WP639D 639D          1       3338 600840      0      0   0%
                                  ------ ------        ----
SUMMARY                            2934K      0          0%
USABLE                             2934K      0          0%
```

The output shows there are five paging volumes on each SSI member constituting 2934 K pages, or about 11.7 GB of page space (a page is 4KB).

5.11 z/VM security issues

This section briefly discusses the following security issues.

- z/VM security products
- High level z/VM security
- Linux virtual machine privilege classes
- z/VM virtual machine and minidisk passwords

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 users access attempts.This book does not address the use of security products.

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 on the Web at:

http://www.vm.ibm.com/library/zvmsecint.pdf

Linux virtual machine privilege classes

Another security issue is the privilege class that Linux virtual machines are assigned. The IBM Redpaper *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

5.11.1 Change passwords in USER DIRECT

All passwords in a vanilla z/VM system are the same as the USER/IDENTITY ID. This is a large security hole. Changing the passwords can be done manually in **XEDIT**, however, this can be both tedious and error-prone. Therefore, an XEDIT macro named `CHPW620 XEDIT` has been included with this book. The source code is in Appendix B.2.1, "The CHPW620 XEDIT macro" on page 377.

To modify all virtual machine and minidisk passwords to the same value, perform the following steps.

- Logon to `MAINT`.
- You should have copied the `CHPW620 XEDIT` macro to the `MAINT 191` disk previously. Verify this with the **LISTFILE** command:

```
==> listfile chpw620 xedit a
CHPW620  XEDIT    A1
```

- 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 good candidate for a password.

- Edit the `USER DIRECT` file with a parameter of **(profile CHPW620)** followed by the new password. Rather than invoking the default profile of `PROFILE XEDIT`, this command will invoke the **XEDIT** macro named `CHPW620 XEDIT` and pass it the new password. For example, to change all passwords to `lnx4vm`, enter the following command:

```
==> x user direct c (profile chpw620) lnx4vm

Changing all passwords to: LNX4VM

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

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

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

- Bring the changes online with the **DIRECTXA** command:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 6 RELEASE 2.0
```

```
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 103 disk pages
```

- **Repeat the DIRECTXA command** on all other members of the SSI.

Your new directory should now be online on both SSI members. Do not forget the new password!

Note that this **XEDIT** macro will only work on a vanilla USER DIRECT file because it searches for the original USER/IDENTITY IDs next to passwords. If you want to change your password again, it should be much easier as you can use the **XEDIT CHANGE** subcommand. For example to change all passwords from lnx4vm to vm4lnx, invoke the following commands:

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

Congratulations, your z/VM system is now customized and ready for Linux. It is recommended that you back up your system to tape.

5.12 Back up and restore your z/VM system

Your SSI system should now be customized with a running TCP/IP stacks, two highly available virtual switches, a startup and shutdown process and with a virtual machine for shared CMS files. You should have changed the passwords. This would be a good time to back up the system to tape. To do so, refer to *Appendix E, Back up the z/VM system to Tape* in the manual *z/VM Installation Guide, version 6 release 2*, GC24-6246.

It is good to practice to restore a system. You don't want to be doing your first restore when the pressure is on. After you complete the backup, try a restore following *Appendix H. Restore the z/VM system backup from tape* in the same manual.

If you do not have a tape device, there are also appendixes on backing up and restoring to and from DASD.

6

Service z/VM

"You cannot solve a problem with the same kind of thinking that created it."

--Albert Einstein

A new release of z/VM is made available approximately every 12 to 18 months. In addition to incorporating fixes to previously identified problems, new releases place an emphasis on new function and features that improve virtualization and the use of z/VM as a hypervisor for other System z operating systems. Customers are advised to run their production z/VM systems at the most current supported version/release available. IBM provides recommended maintenance service for all components, products, and features delivered with the z/VM base system in a single package called a Recommended Service Upgrade (RSU). An RSU contains cumulative service in a prebuilt format that is available on tape or electronically. Customers are advised to maintain RSU currency of a minimum of six months on their production z/VM systems.

This chapter describes how to apply the two main types of service:

- A Recommended Service Upgrade (RSU) which is analogous to a *Service Pack.*
- A Programming Temporary Fix (PTF) which is analogous to a *bug fix*.

The process to install these types of service is basically the same.

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

The application of corrective service to z/VM is covered in two manuals:

- *z/VM V6.1 Guide for Automated Installation and Service*, (see Part 4), on the Web at:

  http://publibz.boulder.ibm.com/epubs/pdf/hcsk2c00.pdf

- *z/VM Service Guide, version 6, release 1*, on the Web at:

  http://publib.boulder.ibm.com/epubs/pdf/hcsf1c00.pdf

These manuals are much more complete than this chapter. You might consider using these first, rather than this chapter, or you should certainly use them as references.

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

- Installs an RSU or applies CORrective service for z/VM components, features, or products.
- Displays either the RSU level of the component specified or whether a particular PTF or APAR has been applied (when used with STATUS).
- 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 deliverable, and were serviced, into production. A good Web site to start at is:

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

The body of the page should look similar to the following figure:

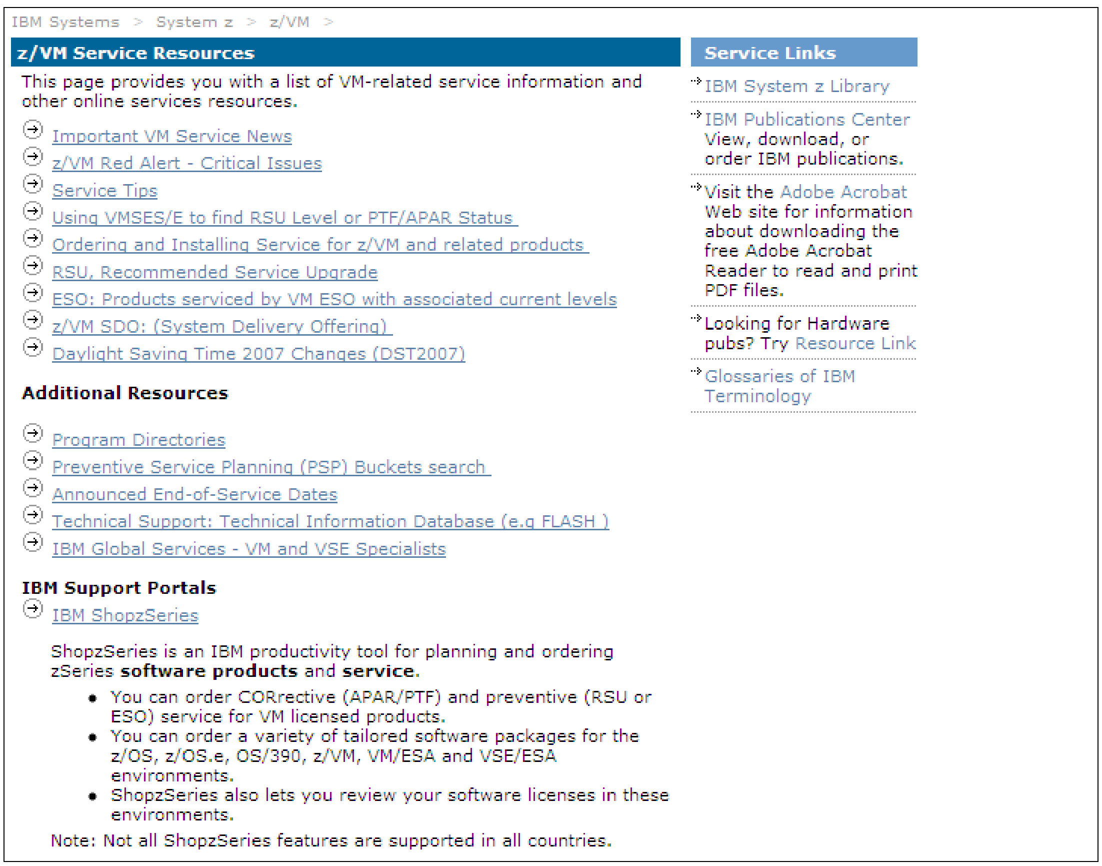


Figure 6-1 z/VM Service main Web page

You may want to consider viewing some of the links from this page.

The following sections comprise this chapter:

- "How to apply a Recommended Service Upgrade (RSU)" on page 91
- "How to apply a PTF" on page 96
- "How to determine the service level of TCPIP" on page 102

6.1 How to apply a Recommended Service Upgrade (RSU)

Applying an RSU is very similar to applying a PTF described in the previous section. z/VM service can be preventive (RSU) or corrective (COR).

The following Web site contains the latest RSU content information:

`http://www.vm.ibm.com/service/rsu/`

The following Web site contains Red Alerts, which contain information about potential high-impact items:

`http://www.vm.ibm.com/service/redalert/`

The section that follows is a summary of applying service and also describes how to obtain service over the Internet using IBM Shopz.

You must first determine if your system needs service. Use the **`QUERY CPLEVEL`** command:

```
==> q cplevel
z/VM Version 6 Release 2.0, service level 1101 (64-bit)
Generated at 01/31/12 15:19:24 EDT
IPL at 06/15/12 10:30:33 EDT
```

The *service level* (or RSU) is a four digit field comprised of two segments, each consisting of two digits. The first two digits represent the last two digits of the year and the second two digits represent the sequential RSU level within that year. Some examples are `0903RSU`, and `1002RSU`. With `0903`, the first two the digits in the level, `09` represent the last two digits of the year 2009 and the `03` represents the third RSU service level of that year. Therefore the `0903` is the third RSU issued in 2009. RSU `1002` would be the second RSU issued in 2010.

The overall steps in applying an RSU are as follow:

- “Get service from the Internet”
- “Download the service files” on page 92
- “Receive, apply, and build the service” on page 93
- “Put the service into production” on page 95

6.1.1 Get service from the Internet

An RSU is obtained by its PTF number. The PTF for the most current RSU is of the form **UM97xyz** where **xyz** is the z/VM version-release-modification level. So for z/VM 6.2 the RSU would be UM97620.

With Shopz, knowing the PTF number is not necessary. If you know you want the latest RSU, you can get it directly, based on the version of z/VM you are running.

Perform the following steps (note that these same steps are documented with some screen shots in 6.2, “How to apply a PTF” on page 96):

- Point a Web browser to the z/VM Service page:

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

- Click on **IBM Shopz** under the *IBM Support Portals* section.
- Click on the link **Sign In for registered users**, usually in the upper right. If you have an user ID and password, use that. If you do not, click on the link **New user registration** and fill out the form to create an ID and password. You must have your IBM customer number. (If you work for IBM, note the link *IBM employees must **sign in here***).

- Click on the link **create new software orders** near the top.
- The *My Orders* page should show. Under the *Package Category* section, click on the **z/VM - Service** radio button and also choose **RSU recommended service** in the drop-down menu. Click **Continue**.
- There will be five screens of forms that are hopefully self-explanatory. On screen 3 of 5, choose the radio button that is applicable to your version of z/VM. In this example **z/VM Version 6.2.0 Stacked 6202RSU (PTF UM97620)** was available.
- On screen 4 of 5 choose **Internet** as the delivery mechanism.
- On screen 5 of 5, complete the form and click **Submit**.
- In a few minutes, you should get two e-mails - one for the core RSU and one for the *PSP bucket* (additional fixes that may have come out after the RSU). Alternatively, you can click the refresh button on your browser - after some time the *Status* should change to a link named **Download** as shown in the following figure:

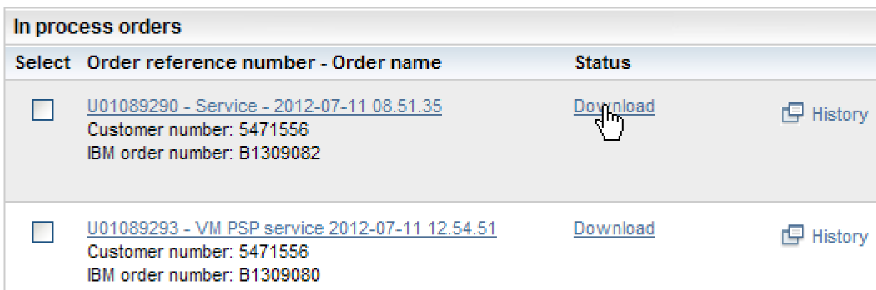


Figure 6-2 Downloading service directly from your browser

6.1.2 Download the service files

In this example, the service files are staged on a desktop machine then copied to z/VM with FTP.

- Download the files to your desktop or another staging system. This example has two files: the `SHIPTFSS` file is for the PSP bucket and the `SHIPRSU1` file is for the RSU.
- FTP the file to the `MAINT620 500` disk. Following is an example of FTPing from a DOS session:

```
C:\Downloads>ftp 9.60.18.251
User (9.60.18.249:(none)): maint620
Password:
ftp> cd maint620.500
250 Working directory is MAINT620 500
ftp> bin
200 Representation type is IMAGE.
ftp> quote site fix 1024
200 Site command was accepted.
ftp> put S9338801.shiptfss
...
ftp> put S9338766.shiprsu1
...
ftp> quit
```

- Logon to `MAINT620`.
- Access the `MAINT620 500` disk as file mode `C`. Query the disks:

```
==> acc 500 c
DMSACC724I 500 replaces C (2CC)
```

```
==> q disk
LABEL  VDEV M  STAT   CYL TYPE BLKSZ   FILES  BLKS USED-(%) BLKS LEFT  BLK TOTAL
MNT191 191  A   R/W   175 3390 4096       26         231-01     31269      31500
MNT5E6 5E6  B   R/W     9 3390 4096      131        1265-78       355       1620
MNT500 500  C   R/W   900 3390 4096        2       50705-31    111295     162000
MNT51D 51D  D   R/W    26 3390 4096      299        1731-37      2949       4680
PMT551 551  E   R/W    40 3390 4096        9          92-01      7108       7200
MNT190 190  S   R/O   207 3390 4096      694       16694-45     20566      37260
MNT19E 19E  Y/S R/O   500 3390 4096     1126       29765-33     60235      90000
```

- List the files on the `C` disk and note the two new files:

```
==> listfile * * c
S1309082 SHIPRSU1 C1
6201RSU1 SERVLINK C1
S1309082 SHIPDOC  C1
```

- Deterse the documentation file changing the file name prefix character to ‘d’:

```
==> deterse s1309082 shipdoc c d1309082 = =
```

- Deterse the RSU file changing the file type to `SERVLINK` (this step can take some time):

```
==> deterse s1309082 shiprsu1 c = servlink =
```

  Usually this step should succeed. However, very large RSUs can fill up the `MAINT 500` disk either on the **FTP** or the **DETERSE** steps. For example, you may get the error on the **DETERSE** step:

```
DMSERD107S Disk C(500) is full
 No traceback - not enough CTL storage
```

  If this occurs, an extra step of creating a larger disk may be necessary.

6.1.3 Receive, apply, and build the service

You must receive, apply, and build the service. Then it can be put into production.

In the past, this was a more lengthy and detailed 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 ALL** 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.

- Log onto a 3270 session as `MAINT620`.
- Access the `MAINT620 500` disk as `C`:

```
==> acc 500 c
DMSACC724I 500 replaces C (2CC)
```

- Apply the service with the **SERVICE ALL** command. The RSU must be applied first (**S8873950 SERVLINK** in this example). Then any PTFs that came after the RSU can be applied:

```
==> service all s1309082
```

```
...
VMFSUT2760I VMFSUFTB processing started
VMFSUT2760I VMFSUFTB processing completed successfully
VMFSRV2760I SERVICE processing completed with warnings
Ready(00004); T=*.**/*.** **:**:**
```

A return code of 0 is ideal. If the last `Ready` line has a number in parenthesis, that is the return code. 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. View details with the **`VMFVIEW SERVICE`** command:

```
==> vmfview service
===> VMFVIEW - Message Log Browse of $VMFSRV  $MSGLOG  A1 <===
You are viewing ¬ST: messages from the LAST run.
Number of messages shown = 7 <===> Number of messages not shown = 764
**********************************************************************
****            SERVICE              USERID: MAINT620             ****
**********************************************************************
****           Date: 07/11/12           Time: 10:54:38            ****
**********************************************************************
CK:VMFSUI2104I PTF UM33449 contains user information. Review the :UMEMO
CK:            section in file UM33449 $PTFPART
WN:VMFBDC2250W The following VMHCD objects have been built on BUILD0 300
WN:            (I) and should be copied to your workstation:
WN:VMFBDC2250W EEQINSTM MSIBIN
WN:VMFBDC2250W The following OSA objects have been built on BUILD0 100
WN:            (K) and should be copied to your workstation:
WN:VMFBDC2250W IOAJAVA  BIN
CK:VMFSRV1233I The following products have been serviced.
CK:VMFSRV1233I CMS CP OSA TCPIP VMHCD
```

For these example warnings, if you are running OSA or HCD then as the `VMFBDC2250W` message states you will need to copy the stated objects to your workstation at some point.

- Press **F3** to get out of **`XEDIT`**.
- ReIPL CMS and press **Enter** at the `VM READ` prompt:

```
==> ipl cms
DMSACC724I 19E replaces Y (19E)
DMSACP723I Y (19E) R/O
z/VM V6.2.0    2012-06-26 17:16
...
```

- Re-access the `MAINT 500` disk as `C`:

```
==> acc 500 c
DMSACC724I 500 replaces C (2CC)
```

- Apply the PSP bucket if there is one. (In this example, there was no PSP bucket for RSU6202, so an older PSP bucket is shown):

```
==> service all S9338801
...
VMFSUT2760I VMFSUFTB processing started
VMFSUT2760I VMFSUFTB processing completed successfully
VMFSRV2760I SERVICE processing completed with warnings
Ready(00004); T=29.96/33.46 15:55:40
```

In this example, the service was installed, but there were warnings.

- Run the **`VMFVIEW SERVICE`** command:

```
==> vmfview service
    ===> VMFVIEW - Message Log Browse of $VMFSRV  $MSGLOG  A1 <===
You are viewing ¬ST: messages from the LAST run.
```

```
Number of messages shown = 1 <===> Number of messages not shown = 510
************************************************************************
****            SERVICE              USERID: MAINT                 ****
************************************************************************
****           Date: 09/16/10           Time: 15:53:09             ****
************************************************************************
RO:VMFAPP2112W PTF UK59536 has a IFREQ requisite for PTF UM33113 in
RO:            product 6VMCMS10 (CMS component for z/VM 6.1.0)
* * * End of File * * *
```

This message is letting you know that there is a relationship between the two PTFs (UM33113 and UK59536). It is advisable to make sure you have both, or know about the requisite and decide it isn't important in your environment.

- Press **F3** to get out of **XEDIT**.
- Log off of MAINT620.

6.1.4 Put the service into production

This section describes how to use the **PUT2PROD** command to put the service into production.

> **Important:** The **PUT2PROD** command will affect your production environment. It is recommended that all users be logged off before running it. Placing service into production should be performed as part of a planned system outage because a **SHUTDOWN REIPL** is recommended after running it.

- Log on to MAINT620 on the first member.
- IPL CMS:

```
==> ipl cms
z/VM V6.2.0    2012-06-26 17:16
...
```

- Use the **PUT2PROD** command to put the service into production. Many screens will scroll by. This command can take quite a number of minutes to complete:

```
==> put2prod
...
VMFP2P1239I CP was serviced. Shutdown and re-IPL the system to employ the new
            service.
VMFP2P1239I CMS was serviced. Re-IPL CMS in all virtual machines running CMS to
            employ the new service.
VMFP2P2760I PUT2PROD processing completed successfully
```

- Review the messages with the **VMFVIEW PUT2PROD** command:

```
==> vmfview put2prod
You are viewing ¬ST: messages from the LAST run.
Number of messages shown = 4 <===> Number of messages not shown = 436
************************************************************************
****      PUT2PROD      SYSTEM: LEFT620       USERID: MAINT620      ****
************************************************************************
****            Date: 07/11/12           Time: 11:16:35             ****
************************************************************************
CK:VMFP2P1233I The following products have been put into production.
CK:            Recycle the appropriate servers.
CK:VMFP2P1233I CMS CP OSA TCPIP VMHCD
CK:VMFP2P1239I CP was serviced. Shutdown and re-IPL the system to employ
CK:            the new service.
CK:VMFP2P1239I CMS was serviced. Re-IPL CMS in all virtual machines
```

```
CK:            running CMS to employ the new service.
```

In this example the only messages are informational. If there are warning or error messages, those issues should be addressed.

- Press **F3** to get out of **XEDIT**.
- Even though the service has been “put into production”, the **QUERY CPLEVEL** command should still return the current service level; in this example 1101 (the first RSU in the year 2011). This is because the new CP load module (nucleus) has not been loaded:

```
==> q cplevel
z/VM Version 6 Release 2.0, service level 1101 (64-bit)
Generated at 06/27/12 09:00:40 EDT
IPL at 06/27/12 09:34:06 EDT
```

- To load the new CP load module, you may use the **SSISHUTD EXEC** included with the files associated with this book:
  - Log off of MAINT620.
  - Log on to MAINT.
  - Issue the **SSISHUTD REIPL** command:

```
==> ssishutd reipl
Are you sure you want to SHUTDOWN REIPL the SSI cluster? (y/n)
y
SYSTEM SHUTDOWN STARTED
HCPSHU960I System shutdown may be delayed for up to 630 seconds
VMSERVS : DMS5BC3108I Shutdown Signal received. STOP processing started
...
```

When your system comes back up, it should be at the new CP service level.

- After the system comes back up in a few minutes, start a new 3270 session and logon as MAINT on the first member.
- Run the **QUERY CPLEVEL** command again,

```
==> q cplevel
z/VM Version 6 Release 2.0, service level 1201 (64-bit)
Generated at 07/11/12 10:55:40 EDT
IPL at 07/11/12 11:31:09 EDT
```

This shows that the new CP load module is now being used, and that the service level is the first RSU in the year 2012.

6.2 How to apply 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), VM65060, was opened when a problem was found with CMM.

The APAR was assigned the following Programming Temporary Fix (PTF) numbers for each of the following z/VM releases:

z/VM 5.4	UM33537
z/VM 6.1	UM33538
z/VM 6.2	**UM33539**

So for z/VM 6.2, you want to apply PTF **UM33539**. Following is an example of how to do so.

6.2.1 Get service using Shopz

Service for z/VM is still available on the media of tape. However, getting service over the Internet is more convenient and becoming more common. To do so, perform the following steps:

- Point a browser to the following URL:

  `http://www14.software.ibm.com/webapp/set2/psearch/search?domain=sysz`

- Enter the APAR number in the *Search For:* text field. In this example the APAR is VM65060, and there was one hit as shown in Figure 6-3 on page 97.

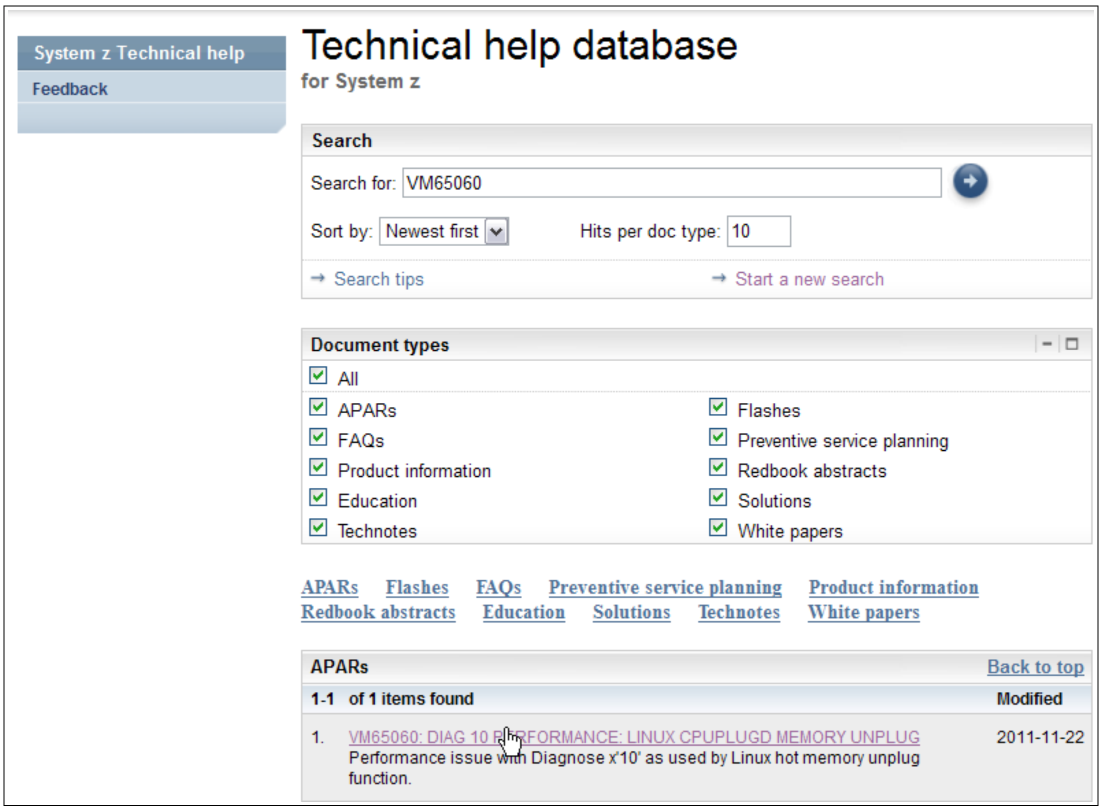


Figure 6-3 Searching for PTFs by APAR number

- Click the link of the APAR description.
- Farther down on the page, note the *Fixed component name* which is important. In this example it is **VM CP**.

At the bottom of the page the Applicable component levels section shows that PTF **UM33539** is available for z/VM 6.2. Before getting that PTF, you may want to be sure that it has not already been applied.

6.2.2 Determine if a PTF has been applied

Check to make sure the PTF has not previously been applied. In this example, the PTF to check for is UM33539.

- Log on to `MAINT620`.
- Use the **`SERVICE ALL STATUS`** command followed by the PTF number to query whether it has been applied:

```
==> service all status um33539
VMFUTL2767I Reading VMFINS DEFAULTS B for additional options
VMFSRV2195I SERVICE ALL STATUS UM33539
VMFSRV2760I SERVICE processing started
DASD 0491 LINKED R/W; R/O BY    10 USERS
DASD 0492 LINKED R/W; R/O BY    10 USERS
DASD 019D LINKED R/W; R/O BY    17 USERS
DASD 0402 LINKED R/W; R/O BY    13 USERS
DASD 193C LINKED R/W; R/O BY    16 USERS
DASD 0200 LINKED R/W; R/O BY     2 USERS
DASD 0201 LINKED R/W; R/O BY PERSMAPI at POKDEV62
DASD 01CC LINKED R/W; R/O BY PERSMAPI at POKDEV62
DASD 029D LINKED R/W; R/O BY     2 USERS
VMFSRV1227I UM33539 is not received or applied
VMFSRV2760I SERVICE processing completed successfully
```

This shows that PTF UM33539 has *not* been applied. The sections that follow describe how to obtain and apply it.

6.2.3 Download the service to z/VM

From the previous APAR web page search, the link for **UM33539** is clicked on which results in a Web page that should be similar to the following:

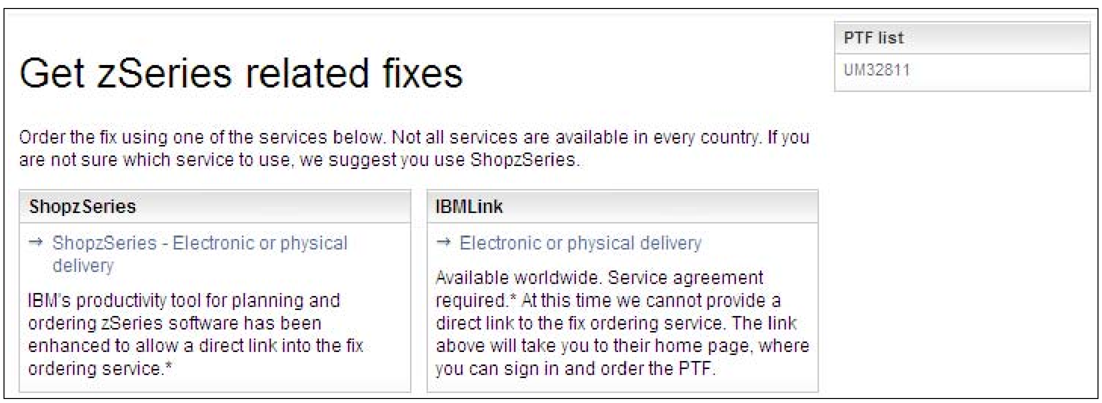


Figure 6-4 Getting fixes from Shopz on IBMLink

- In this example the link **ShopzSeries - Electronic or physical delivery** is clicked on.
- Sign into Shopz with your IBM ID and follow the five self-explanatory steps to order your PTF (if you work for IBM note the **sign in here** link) When you are finished, click on **Submit** to place your order.
- You should receive an e-mail within a few minutes. It will have your order number a link to start the download of service files. Following is an example of the important information in the e-mail.

```
From:    Oms Client01/Boulder/IBM
Subject: IBM Order <Bxxxxxxx> is ready for download.
...
To access your order directly, go to:
https://www14.software.ibm.com/webapp/ShopzSeries/ShopzSeries.jsp?action=download&orderI
d=<Uxxxxxxxd>0
```

- Point your browser to the link in the e-mail. You should see a Web page similar to the following:

Figure 6-5 Web page created for downloading a PTF

- Choose a method of downloading the **VMSES PTF Envelope** and the **VMSES Documentation Envelope** to a desktop or staging machine. In this example, **Download Director** was used.
- Copy both the SES and the documentation envelopes to z/VM in binary with fixed 1024 byte records to the `MAINT 500` disk. Usually, FTP is used. As you are downloading the files, note the file sizes. Following is an example of FTPing from a DOS session:

```
C:\downloads> ftp 9.60.18.251
User (9.60.18.249:(none)): maint620
Password:
...
ftp> cd maint620.500
250 Working directory is MAINT620 500
ftp> bin
200 Representation type is IMAGE.
ftp> quote site fix 1024
200 Site command was accepted.
ftp> mput s1041690.*
150 Storing file 'S1041690.SHIPDOCS'
250 Transfer completed successfully.
ftp: 6144 bytes sent in 0.00Seconds 6144000.00Kbytes/sec.
mput S1041690.SHIPTFSS? y
150 Storing file 'S1041690.SHIPTFSS'
250 Transfer completed successfully.
ftp: 10240 bytes sent in 0.00Seconds 10240000.00Kbytes/sec.
ftp> quit
```

- Logon to z/VM as `MAINT620`.
- Access the `MAINT620 500` disk as `C`:

```
==> acc 500 c
DMSACC724I 500 replaces C (2CC)
```

- Verify that the files are there with the **`LISTFILE`** command:

```
==> listfile * * c
S1041690 SHIPDOCS C1
S1041690 SHIPTFSS C1
6201RSU1 SERVLINK C1
```

- The envelope files arrive in a compressed format to speed downloads. In order to use them they must first be renamed to have a file type of `SERVLINK` and uncompressed with the **`DETERSE`** command. Therefore it is recommended to leave the file name of the SES envelope unchanged, but change the prefix letter of the documentation envelope to `D`. First rename them, then use the **`DETERSE`** command with the **`(REPLACE`** parameter to uncompress them in place and save disk space:

```
==> rename s1041690 shiptfss c = servlink =
==> rename s1041690 shipdocs c d1041690 servlink =
==> deterse s1041690 servlink c = = = (replace
==> deterse d1041690 servlink c = = = (replace
```

Be sure all commands complete successfully.

6.2.4 Receive, apply, and build service

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

To prepare to use the **`SERVICE`** command, you must have a minidisk with a lot of free space - that is what the `MAINT620 500` minidisk is for.

- Access the `MAINT620 500` disk as file mode C:

```
==> acc 500 c
DMSACC724I 500 replaces C (2CC)
```

- Use the **`SERVICE ALL`** command specifying the envelope files you downloaded. Many screens of output will scroll by and will automatically be cleared. Important messages will be saved to the `500` disk. This process may take many minutes. Following is an example:

```
==> service all d1041690
...
VMFSUT2760I VMFSUFTB processing completed successfully
VMFSRV2760I SERVICE processing completed successfully
==> service all s1041690
...
VMFSRV1233I The following products have been serviced.
VMFSRV1233I CP
VMFSRV2760I SERVICE processing completed successfully
```

If you see no number in parenthesis after the `Ready;` prompt, then the return code is 0. Any non-zero return code will be in parenthesis. A return code of 0 is ideal. In general a return code of 4 is acceptable - it means that only warnings were issued. A return code of 8 or greater generally means that errors were encountered.

- The output files are of the form `$VMF* $MSGLOG`. You may wish to inspect these files.

```
==> filel $vmf* $msglog
$VMFSRV  $MSGLOG  A1 V        80       1582        29  1/31/12 15:19:27
      $VMFBLD  $MSGLOG  A1 V        80        841         12  1/31/12 15:19:25
      $VMFAPP  $MSGLOG  A1 V        80        212          3  1/31/12 15:19:15
      $VMFREC  $MSGLOG  A1 V        80         69          1  1/31/12 15:19:15
      $VMFMRD  $MSGLOG  A1 V        80        270          4  1/31/12 15:19:14
      $VMFINS  $MSGLOG  A1 V        80        223          4 11/29/11  2:32:50
      $VMFP2P  $MSGLOG  A1 V        80       1741         32 11/29/11  0:55:22
```

- 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
===> VMFVIEW - Message Log Browse of $VMFSRV  $MSGLOG  A1 <===
You are viewing ¬ST: messages from the LAST run.
Number of messages shown = 2 <===> Number of messages not shown = 126
**************************************************************************
****             SERVICE              USERID: MAINT620             ****
**************************************************************************
****           Date: 01/31/12           Time: 15:19:13             ****
**************************************************************************
CK:VMFSRV1233I The following products have been serviced.
CK:VMFSRV1233I CP
```

Ideally there will be no output. If there are errors they must be addressed. If there are warnings, they may be acceptable but should be investigated.

6.2.5 Put the service into production

To put the service into production, perform the following steps:

- Logon as MAINT620.
- IPL CMS:

```
==> ipl cms
z/VM V6.2.0    2011-11-15 11:26
```

- Access the VMSES/E test build disk as file mode B:

```
==> acc 5e6 b
DMSACC724I 5E6 replaces B (5E6)
```

- Use the **PUT2PROD** command to put the service into production.

```
==> put2prod
...
VMFP2P1239I CP was serviced. Shutdown and re-IPL the system to employ the new
            service.
VMFP2P2760I PUT2PROD processing completed successfully
```

  Note the second to last message informs you that a **SHUTDOWN** and re-IPL is necessary. Again, watch for a return code of 0.

- Your PTF should now be *put into production*. You may or may not have to re-IPL the system, depending on the nature of the PTF applied. If you have to, Be sure you are in a position to re-IPL your system. You may want to shutdown and ReIPL one member at a time with Live Guest Migrations of important Linux systems in between.
- Your z/VM system should come back in a few minutes. When the system comes back up, start a 3270 session to MAINT and again query the status of the PTF:

```
==> service cp status UM33539
VMFUTL2767I Reading VMFINS DEFAULTS B for additional options
VMFSRV2195I SERVICE CP STATUS UM33539
VMFSRV2760I SERVICE processing started
VMFSRV1226I CP (6VMCPR20%CP) PTF UM33539 status:
VMFSRV1226I    RECEIVED  01/31/12 15:19:15
VMFSRV1226I    APPLIED   01/31/12 15:19:15
VMFSRV1226I    BUILT     01/31/12 15:19:27
VMFSRV1226I    PUT2PROD  01/31/12 15:24:46 POKDEV62
VMFSRV2760I SERVICE processing completed successfully
```

- **Repeat the steps in this section** for all members in the SSI cluster.

This shows that the PTF has been successfully applied.

6.2.6 Check for APARMEMO files

After you have applied PTFs, you should check for files with a file type of APARMEMO on the MAIN620T 500 disk. These files may have additional instructions on work to do after the PTFs have been applied. Perform the following steps:

- Access the MAINT 500 disk as C and list the files with file type APARMEMO:

```
==> acc 500 c
DMSACC724I 500 replaces C (2CC)
==> listfile * aparmemo c
6VMCPR20 APARMEMO C1
```

  In this example, there is one APARMEMO file.

- Look at the contents of the file:

```
==> type 6vmcpr20 aparmemo c

APAR MEMOS      01/30/12.14:16:55
=================================

        THE FOLLOWING MEMOS WERE INCLUDED WITH THE PTFS SHIPPED:

        NONE.
```

In this example the APARMEMO file was created, but no additional memorandums are present.

You will not see any new information in the APARMEMO file if you have not done **SERVICE** against the documentation SERVLINK file. This is because the <prodid> MEMO file is in the documentation SERVLINK file.

6.3 How to determine the service level of TCPIP

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/

Perform the following steps:

- Logon to **TCPMAINT** on one of the SSI members. Use the **QUERY VMLAN** command to determine the latest APAR applied:

```
==> q vmlan
q vmlan
VMLAN maintenance level:
  Latest Service: Base
VMLAN MAC address assignment:
  System MAC Protection: OFF
  MACADDR Prefix: 02000B USER Prefix: 020000
  MACIDRANGE SYSTEM: 000001-FFFFFF
             USER:   000000-000000
VMLAN Unified Resource Manager status:
  Hypervisor Access: YES       Status: DISABLED BY SMAPI
  ID: NONE
  MAC Prefix: 02D737
```

```
VMLAN default accounting status:
  SYSTEM Accounting: OFF       USER Accounting: OFF
VMLAN general activity:
  PERSISTENT Limit: INFINITE   Current: 3
  TRANSIENT  Limit: INFINITE   Current: 0
```

The **Latest Service:** line shows that no APAR has been applied.

- The maintenance level of the TCP/IP stack is important to virtual networking. To determine this, first get the active virtual switch controller:

```
==> q vswitch vsw1
VSWITCH SYSTEM VSW1     Type: QDIO    Connected: 2    Maxconn: INFINITE
  PERSISTENT  RESTRICTED    NONROUTER                 Accounting: OFF
  USERBASED
  VLAN Unaware
  MAC address: 02-00-0B-00-00-01    MAC Protection: OFF
  State: Ready
  IPTimeout: 5         QueueStorage: 8
  Isolation Status: OFF
 Uplink Port:
  RDEV: 4203.P00 VDEV: 0600 Controller: DTCVSW1
        EQID: OSASET1
  RDEV: 4300.P00 VDEV: 0603 Controller: DTCVSW2  BACKUP
        EQID: OSASET1
```

This shows the controller is named DTCVSW1.

- 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 620       TCP/IP Server Name: DTCVSW1

IBM 2818; z/VM Version 6 Release 2.0, service level 1101 (64-bit), VM TCP/IP Lev
el 620; RSU 0000 running TCPIP MODULE E2 dated 09/30/11 at 06:55
TCP/IP Module Load Address:  00C15000
```

This shows information about the current 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
DTCLVL3306I SLVL data obtained; file TCPIP SLVLDATA A created
==> x TCPIP SLVLDATA
...
SLVL  TCPIP     ZVM620
...
SLVL SLVL  TCTOOSD   ZVM620
...
```

6.4 Moving on

You should now be done installing, configuring and servicing z/VM. A great attribute of z/VM is that it normally hums along with little maintenance required. It is now time to change your focus to Linux.

Part 2

RHEL 6.2 Linux

This part of the book focuses on Red Hat Enterprise Linux (RHEL). It consists of the following chapters:

- Chapter 7, “Install RHEL 6.2 on LNXADMIN” on page 107, describes how to install and configure RHEL 6.2 onto the *Linux Administration* server, which does the cloning and other tasks.
- Chapter 8, “Install and configuring the RHEL 6.2 golden image” on page 135, describes how to install and configure the *golden image* which it is cloned from.
- Chapter 9, “Configure RHEL 6.2 for cloning” on page 151 explains how to prepare z/VM virtual machines and clone your first virtual server both manually and by using a shell script.
- Chapter 10, “Installing Linux with kickstart” on page 163, describes how to use Red Hat’s **kickstart** tool to create Linux systems. This is fundamentally different from cloning in that an automated install is implemented. You may try kickstart and you may also try cloning. Understand that they try to accomplish the same goal of being able to quickly get Linux systems up and running, and that you do not need to use both.
- Chapter 11, “Create RHEL 6.2 appliances” on page 169, shows how to configure cloned Linux images into the following *appliances*:
  - Web server virtual server
  - LDAP virtual server
  - File and print virtual server
  - Application development system
- Chapter 12, “Service Linux with the Red Hat Network” on page 185, describes how the *Red Hat Network* works. It provides centralized management and provisioning for multiple RHEL 6.2 systems.

7

Install RHEL 6.2 on LNXADMIN

"*The most incomprehensible thing about the world is that it is at all comprehensible.*"

— Albert Einstein

RHEL or SLES?: If you are working only with SLES 11 SP2, you can skip the majority of this chapter and follow the general approach in Chapter 13, "Install SLES 11 SP2 on LNXADMIN" on page 191. However you will have to define the `IDENTITY LNXADMIN` as described in section 7.1.1, "Create the identity LNXADMIN" on page 108, then, while installing SLES 11 SP2, create a large logical volume to be mounted over `/var/`.

By now, you have created a new z/VM *user ID* or Single Configuration Virtual Machine (SCVM), `LNXMAINT`. Its main purpose is to provide a common CMS disk for all Linux virtual machines.

Now it is time to create the first *Identity* or Multi-Configuration Virtual Machine (MCVM), `LNXADMIN`. An MCVM can be logged on to all members of the SSI at the same time. Therefore, it is not possible to migrate an MCVM between SSI members.

This virtual machine will serve a number of administrative purposes:

- The Linux installation server: a file system *tree* of RPMs and other files required for installation are made available with NFS.
- The clone server: for cloning from the golden image to target virtual machines (see Chapter 9, "Configure RHEL 6.2 for cloning" on page 151)
- The Red Hat Kickstart server: for hosting the files necessary for automated installations (see Chapter 10, "Installing Linux with kickstart" on page 163)
- The administration server for other systems management tools such as xCAT.

In this section, you will perform following tasks:

- "Install the Linux administration system" on page 108
- "Configure the Linux administration system" on page 124

7.1 Install the Linux administration system

In this section you will install RHEL 6.2 onto the IDENTITY LNXADMIN.

7.1.1 Create the identity LNXADMIN

In this section you will define an SSI *identity*, also known as a Multi-Configuration Virtual Machine (MCVM). To do so, perform the following steps:

- Logon to MAINT.
- Determine the number of physical processors active with the **QUERY PROCESSORS** command:

```
==> q proc
PROCESSOR 00 MASTER CP
PROCESSOR 01 ALTERNATE CP
```

  In this example, there are two processors. This number will be used in the following steps.

- Make a backup of the USER DIRECT file:

```
==> copy user direct c = direwrks = (rep
```

- 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. Next you will create a user directory profile for Linux systems.

- 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, type the **""** prefix commands to duplicate the IBMDFLT profile on lines 38-50:

```
"" 38 *
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 *
"" 50 *****************************
```

- Press **Enter** and the block will be duplicated.
- Edit the duplicated profile by deleting the three LINK MAINT 040x lines, and inserting the lines that are shown in bold text:

```
PROFILE LNXDFLT
  COMMAND SET VSWITCH VSW1 GRANT &USERID
  COMMAND DEFINE NIC 600 TYPE QDIO
  COMMAND COUPLE 600 TO SYSTEM VSW1
  COMMAND SET VSWITCH VSW2 GRANT &USERID
  COMMAND DEFINE NIC 700 TYPE QDIO
  COMMAND COUPLE 700 TO SYSTEM VSW2
  CPU 00 BASE
```

```
CPU 01
IPL CMS
MACHINE ESA 8
OPTION CHPIDV ONE
CONSOLE 0009 3215 T
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
LINK MAINT 0190 0190 RR
LINK MAINT 019D 019D RR
LINK MAINT 019E 019E RR
LINK LNXMAINT 0192 0191 RR
LINK TCPMAINT 0592 0592 RR
```

Notes:

- The six **COMMAND** lines gives the virtual machine access to virtual switches VSW1 and VSW2 at logon time when the virtual machine is created. This precludes the need to add **VSWITCH GRANT** statements each time a Linux virtual machine is created.
- The two **CPU** lines define two virtual CPUs. It is recommended to set the number of virtual CPUs less than or equal the number of physical CPUs.
- The MACHINE statement sets the virtual machine type to ESA with a maximum of 8 CPUs. Even if your hardware does not have 8 IFLs, it is alright to set the maximum to 8 to leave *headroom*.
- The **OPTION CHPIDV ONE** allows virtual machines to be relocated between SSI members.
- The last line provide read access to LNXMAINT 192 disk as the user's 191 disk.

- Go to the bottom of the file and add the definition for a new identity named LNXADMIN. This virtual machine is given class B, D and E privilege classes, aside from the typical class G, in order to run the **FLASHCOPY** command (B), the **QUERY ALLOC MAP** (D) command, and the **QUERY NSS** (E) commands:

```
IDENTITY LNXADMIN LNX4VM 256M 1G BDEG
 INCLUDE LNXDFLT
 BUILD ON POKDEV62 USING SUBCONFIG LNXADM-1
 BUILD ON POKTST62 USING SUBCONFIG LNXADM-2
 OPTION LNKNOPAS
SUBCONFIG LNXADM-1
 MDISK 0100 3390 0001  3338 JM6290 MR LNX4VM LNX4VM LNX4VM
 MDISK 0101 3390 0521  2818 JM6289 MR LNX4VM LNX4VM LNX4VM
 MDISK 0102 3390 0001 10016 JM61A5 MR LNX4VM LNX4VM LNX4VM
 MDISK 0103 3390 0001 10016 JM61B2 MR LNX4VM LNX4VM LNX4VM
SUBCONFIG LNXADM-2
 MDISK 0100 3390 0001  3338 JM6293 MR LNX4VM LNX4VM LNX4VM
```

This virtual machine will have the following minidisks and virtual disks:

Table 7-1 Minidisks to be defined

Minidisk or virtual disk	Description
LNXADM-1 100	The root file system of the Linux administration system on SSI member 1. This will serve as the administration point for all your Linux virtual servers.
LNXADMIN-1 101-103	Minidisks used to create a logical volume mounted over /var/ This file system is used to make the installation trees, files associated with this book, and possibly other data available over NFS.

Minidisk or virtual disk	Description
LNXADM-2 100	The root file system of the Linux administration system on SSI member 2. This will be used in Chapter 13, "Install SLES 11 SP2 on LNXADMIN" on page 191.
300-301	These are virtual disk swap spaces that are not defined in USER DIRECT file, but defined by calls to the **SWAPGEN EXEC** in the user's **PROFILE EXEC** so that when the user ID logs on the virtual disks are created.

- Go back to the top of the file and search for string USER $ALLOC$. Add cylinder 0 of each of the new volumes to this dummy user ID so they don't show up as gaps in the USER DISKMAP report file:

```
====> top
====> /user $alloc$
USER $ALLOC$  NOLOG
 MDISK A00 3390 000 001 CV6284 R
 MDISK A01 3390 000 001 CV6285 R
 MDISK A02 3390 000 001 CV6286 R
 MDISK A03 3390 000 001 CV6287 R
 MDISK A04 3390 000 001 JV6280 R
 MDISK A05 3390 000 001 JV6283 R
 MDISK A06 3390 000 001 WV639B R
 MDISK A07 3390 000 001 WV639E R
 MDISK A08 3390 000 001 JM6289 R
 MDISK A09 3390 000 001 JM6290 R
 MDISK A0A 3390 000 001 JM6293 R
 MDISK A0B 3390 000 001 JM61A5 R
 MDISK A0C 3390 000 001 JM61B2 R
...
====> file
```

- Run **DISKMAP** to check for overlaps and gaps.

```
==> diskmap user
==> x user diskmap
====> all /gap/|/overlap/
...
====> quit
```

- When the disk layout is correct run **DIRECTXA** to bring the changes online:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 6 RELEASE 2.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 104 disk pages

EOJ DIRECTORY UPDATED AND ON LINE
```

You have now defined the virtual machine that will be the Linux administrative system.

7.1.2 Set LNXADMIN to start a IPL time

It is recommend that the new Linux administrative system be started at SSI IPL time. To do so, add an **XAUTOLOG** statement to the **PROFILE EXEC** on AUTOLOG

- Use the **LINK** and **ACCESS** commands to link and access the AUTOLOG1 191 disk read-write. This is the disk with the common EXEC that is run at IPL time for each member:

```
==> link autolog1 191 1191 mr
==> acc 1191 f
```

- Edit the file **PROFILE EXEC**. Add a line to automatically start the LNXADMIN identity with the **XAUTOLOG** command:

```
==> x profile exec f   // add one line
...
"CP XAUTOLOG TCPIP"              /* Autolog TCPIP                    */
"CP SET MDC STOR 0M 256M"        /* Limit minidisk cache in CSTOR    */
"CP SET MDC XSTORE 0M 0M"        /* Disable minidisk cache in XSTOR  */
"CP SET SIGNAL SHUTDOWN 600"     /* Allow guests 10 min to shut down */
"CP XAUTOLOG LNXADMIN"           /* Start the Linux admin machine    */
...
====> file
```

- Release and detach the AUTOLOG1 191 disk with the **RELEASE** command:

```
==> rel f (det
DASD 1191 DETACHED
```

The LNXADMIN identity should now be automatically started on all SSI cluster members.

7.1.3 Prepare RHEL 6.2 bootstrap files

To IPL a RHEL 6.2 installation system, four bootstrap files must be prepared. Three are punched to z/VM reader and then IPLed: 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 DVD. Also, a small REXX EXEC is commonly used to clean out the reader, punch the three files and IPL the reader. A sample RHEL 6.2 parameter file, configuration file and install EXEC are supplied and should be on the LNXMAINT 192 disk (see section 5.9.4, “Copy files associated with this book” on page 83). Therefore, only the kernel and RAMdisk need to be copied.

- Start an SSH session as root on the NFS server.
- Use the **ftp** command to copy the RHEL 6.2 kernel and initial RAMdisk to LNXMAINT’s 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/rhel62/dvd1/images
# ftp 9.60.18.251
Name (9.60.18.251: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 rhel62.initrd
...
23651842 bytes sent in 00:01 (11.34 MB/s)
ftp> put kernel.img rhel62.kernel
...
8016384 bytes sent in 00:01 (6.01 MB/s)
ftp> quit
```

- From your 3270 session, Log Off of MAINT if you are still logged on.
- Logon to LNXMAINT.
- The files SAMPLE PARM-RH6, SAMPLE CONF-RH6 should exist on the LNXMAINT 192 (D) disk as they were copied in 5.9.4, "Copy files associated with this book" on page 83. Use the **FILELIST** command to verify that the files were copied:

```
==> filel sample *-rh6 d
LNXMAINT FILELIST A0  V 169  Trunc=169 Size=2 Line=1 Col=1 Alt=0
Cmd   Filename Filetype Fm Format Lrecl    Records     Blocks   Date     Time
      SAMPLE   CONF-RH6 D1 V         38         13          1 11/16/11 11:16:17
      SAMPLE   PARM-RH6 D1 V         80          3          1 11/16/11 11:16:17
```

- The RHEL62 EXEC should exist as well as the RHEL 6.2 kernel and initial RAMdisk. Verify they were in Fixed 80 byte record format. Use the **FILELIST** command to verify that the files were copied:

```
==> filel rhel62 * d
LNXMAINT FILELIST A0  V 169  Trunc=169 Size=3 Line=1 Col=1 Alt=0
Cmd   Filename Filetype Fm Format Lrecl    Records     Blocks   Date     Time
      RHEL62   KERNEL   D1 F         80     113488       1705 11/16/11 17:16:55
      RHEL62   INITRD   D1 F         80     223194       4360 11/16/11 17:16:35
      RHEL62   EXEC     D1 V         63         44          1 11/16/11 11:16:17
```

- Verify that the file **RHEL62 EXEC** has the correct information. Note the kernel and RAMdisk have hard coded file names (RHEL6), but the file name of the parameter file will be the user ID (**userid()** function) of the user running the EXEC:

```
==> type rhel62 exec d
...
Address 'COMMAND'
'CP SPOOL PUN *'
'CP CLOSE RDR'
'CP PURGE RDR ALL'
'PUNCH RHEL62 KERNEL * (NOHEADER'
'PUNCH' Userid() 'PARM-RH6 * (NOHEADER'
'PUNCH RHEL62 INITRD * (NOHEADER'
'CP CHANGE RDR ALL KEEP'
'CP IPL 00C CLEAR'
Exit
```

- There are two text files needed to install RHEL 6.2: a parameter file and a configuration file. A sample parameter file is provided, named SAMPLE PARM-RH6. It has some values, the most important value, the CMSCONFFILE variable, points to the configuration file which remains on a CMS minidisk. Copy the sample parameter file to a new file with a file name of LNXADMIN. Change the configuration file variable to point to a file with the same file name:

```
==> copy sample parm-rh6 d lnxadmin = =
==> x lnxadmin parm-rh6 d
root=/dev/ram0 ro ip=off ramdisk_size=40000
CMSDASD=191 CMSCONFFILE=LNXADMIN.CONF-RH6
vnc vncpassword=12345678
```

- Copy the sample configuration file and modify the appropriate fields. Refer to the worksheet in section 2.8.5, "Host names worksheet" on page 23. Following are the values used for the example in this book.

Important: If you decided to work with only layer 2 VSWITCHes instead of one layer two and one layer 3, make one additional change:

```
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
```

This will specify to Linux to use the first NIC at virtual device addresses 0600-0602, which will attach it to VSWITCH vsw1.

```
==> copy sample conf-rh6 d lnxadmin = =
==> x lnxadmin conf-rh6
DASD=100-103,300-301
HOSTNAME=gpok151.endicott.ibm.com
NETTYPE=qeth
IPADDR=9.60.18.151
SUBCHANNELS=0.0.0700,0.0.0701,0.0.0702
NETMASK=255.255.255.128
SEARCHDNS=endicott.ibm.com
GATEWAY=9.60.18.129
DNS=9.0.3.1
MTU=1500
PORTNAME=DONTCARE
PORTNO=0
LAYER2=1
VSWITCH=1
```

- Linux user IDs will pick up their PROFILE EXEC from LNXMAINT 192. This file runs when you press **Enter** at the VM READ prompt. It creates two virtual disks with the SWAPGEN EXEC to later be used as swap spaces. It also performs a few other functions including IPLIng Linux automatically if the virtual machine is logged on disconnected. View the contents of the PROFILE EXEC with 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
```

7.1.4 Install RHEL 6.2 Linux

Perform the following steps to begin the Linux installation:

- Logon to LNXADMIN. The **PROFILE EXEC** from the LNXMAINT 192 disk should prompt you to IPL minidisk 100. Since there is nothing installed yet, answer **no**.

```
LOGON LNXADMIN
00: NIC 0600 is created; devices 0600-0602 defined
```

```
00: z/VM Version 6 Release 2.0, Service Level 1101 (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 15:50:36 EDT THURSDAY 07/28/11
HCPSWS2858I LNXADMIN connection to SYSTEM VSW1 has been updated by LNXADMIN.
00: Command complete
z/VM V6.2.0    2011-06-28 08:48
Enter
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
```

- Set the memory size to 1 GB with the CP **DEFINE STORAGE** command:

```
==> def stor 1g
00: STORAGE = 1G
00: Storage cleared - system reset.
```

- IPL CMS, and again answer no:

```
==> ipl cms
z/VM V6.2.0    2012-06-21 16:54
Enter
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
```

- To begin the install program, run the **RHEL62 EXEC**. You should see many screens of output scrolling by:

```
==> rhel62
   00: 0000003 FILES PURGED
   00: RDR FILE 0013 SENT FROM LNXADMIN PUN WAS 0013 RECS 113K CPY  001 A NOHOLD NO
   KEEP
   00: RDR FILE 0014 SENT FROM LNXADMIN PUN WAS 0014 RECS 0003 CPY  001 A NOHOLD NO
   KEEP
   00: RDR FILE 0015 SENT FROM LNXADMIN PUN WAS 0015 RECS 224K CPY  001 A NOHOLD NO
   KEEP
   00: 0000003 FILES CHANGED
   00: 0000003 FILES CHANGED
   Initializing cgroup subsys cpuset
   Initializing cgroup subsys cpu
   Linux version 2.6.32-202.el6.s390x (mockbuild@s390-009.build.bos.redhat.com) (gc
   c version 4.4.5 20110214 (Red Hat 4.4.5-6) (GCC) ) #1 SMP Wed Sep 21 15:34:33 ED
   T 2011
   setup: Linux is running as a z/VM guest operating system in 64-bit mode
   Zone PFN ranges:
     DMA      0x00000000 -> 0x00080000
     Normal   0x00080000 -> 0x00080000
   Movable zone start PFN for each node
   early_node_mapÝ1¨ active PFN ranges
       0: 0x00000000 -> 0x00040000
   PERCPU: Embedded 12 pages/cpu @00000000029fd000 s18688 r8192 d22272 u65536
   pcpu-alloc: s18688 r8192 d22272 u65536 alloc=16*4096
   pcpu-alloc: Ý0¨ 00 Ý0¨ 01 Ý0¨ 02 Ý0¨ 03 Ý0¨ 04 Ý0¨ 05 Ý0¨ 06 Ý0¨ 07
   pcpu-alloc: Ý0¨ 08 Ý0¨ 09 Ý0¨ 10 Ý0¨ 11 Ý0¨ 12 Ý0¨ 13 Ý0¨ 14 Ý0¨ 15
```

```
pcpu-alloc: Ý0¨ 16 Ý0¨ 17 Ý0¨ 18 Ý0¨ 19 Ý0¨ 20 Ý0¨ 21 Ý0¨ 22 Ý0¨ 23
pcpu-alloc: Ý0¨ 24 Ý0¨ 25 Ý0¨ 26 Ý0¨ 27 Ý0¨ 28 Ý0¨ 29 Ý0¨ 30 Ý0¨ 31
pcpu-alloc: Ý0¨ 32 Ý0¨ 33 Ý0¨ 34 Ý0¨ 35 Ý0¨ 36 Ý0¨ 37 Ý0¨ 38 Ý0¨ 39
pcpu-alloc: Ý0¨ 40 Ý0¨ 41 Ý0¨ 42 Ý0¨ 43 Ý0¨ 44 Ý0¨ 45 Ý0¨ 46 Ý0¨ 47
pcpu-alloc: Ý0¨ 48 Ý0¨ 49 Ý0¨ 50 Ý0¨ 51 Ý0¨ 52 Ý0¨ 53 Ý0¨ 54 Ý0¨ 55
pcpu-alloc: Ý0¨ 56 Ý0¨ 57 Ý0¨ 58 Ý0¨ 59 Ý0¨ 60 Ý0¨ 61 Ý0¨ 62 Ý0¨ 63
Built 1 zonelists in Zone order, mobility grouping on.  Total pages: 258560
Kernel command line: root=/dev/ram0 ro ip=off ramdisk_size=40000
                     CMSDASD=191 CMSCONFFILE=LNXADMIN.CONF-RH6
                     vnc vncpassword=12345678
...
```

- When installing RHEL 6.2 onto a layer 2 virtual switch, there does not appear to be a way to set the parameters such that a MAC address is not prompted for. You should see a question similar to the following for setting the MAC address value. Press **Enter twice** to continue:

```
Unique MAC address (e.g. 02:00:00:00:00:00, ? for help). Default is 02:00:0B:00:
00:08:
Enter, Enter
...
```

- If the DASD you are using has never been formatted for Linux, you may get many screens 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 just have to clear the screen many times or the install process will freeze. An alternative to clearing the screen many times is to issue the following CP **TERM** command:

```
#cp term more 0 0
```

  Press **Enter** and the screen 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 then another 10 seconds to clear the screen with the command: **#cp term more 50 10**.

- You should be prompted again. Enter **c** to continue:

```
...
Initial configuration completed.

c) continue, p) parm file/configuration, n) network state, r) restart, s) shell
c
...
```

- The kernel should continue to boot until you see the following messages:

```
...
Connect now to 9.60.18.151 and log in as user 'install' to start the installatio
n.
E.g. using: ssh -x install@9.60.18.151
For VNC or text mode, disable X11 forwarding (recommended) with 'ssh -x'.
For X11, enable X11 forwarding with 'ssh -X'.

You may log in as the root user to start an interactive shell.
```

- **Important!** The message says to log in as the user **install**, however, there is an intermediate step. There is an issue where the Red Hat installer does not recognize disks that were have been formatted with **CPFMTXA**. If you have followed all the steps in this book, this will be the case (if you have previously used **dasdfmt** to format these minidisks, you can skip this step) You must first start an SSH session, login as root and use **dasdfmt** to format the disks.

To do so, perform the following steps:

– Start SSH session to the install system and log in as **root**. A password will not be required.

– Invoke the **lsdasd** command and observe the disks:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasdb     94:4    ECKD  ???    2347MB    ???
0.0.0101   active      dasdc     94:8    ECKD  ???    1981MB    ???
0.0.0102   active      dasdd     94:12   ECKD  ???    7042MB    ???
0.0.0103   active      dasde     94:16   ECKD  ???    7042MB    ???
0.0.0300   active      dasdf     94:20   FBA   ???    256MB     ???
0.0.0301   active      dasdg     94:24   FBA   ???    512MB     ???
```

Here, the mindisks `100-103` correspond to `dasdb`, `dasdc`, `dasdd` and `dasde`.

– Format the minidisks in parallel with the following **for** loop:

```
# for i in b c d e
> do
>   dasdfmt -b 4096 -y -f /dev/dasd$i &
> done
[1] 1005
[2] 1006
[3] 1007
[4] 1008
```

– You may need to press **Enter** after a few minutes to see the jobs in the background complete.

```
Rereading the partition table... ok
Finished formatting the device.
Rereading the partition table... ok
Finished formatting the device.
Rereading the partition table... ok
Finished formatting the device.
Rereading the partition table... ok

[1]   Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
[2]   Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
[3]-  Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
[4]+  Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
```

– After the **for** loop completes, exit that SSH session

```
# exit
```

- Start an SSH session to the install system and log in as **install**. A password will not be required.
- Figure 7-1 on page 117 shows the initial screen of the installer. Use the **Tab** key to move between fields. Use the arrow keys to move among choices and **Enter** to select a choice.

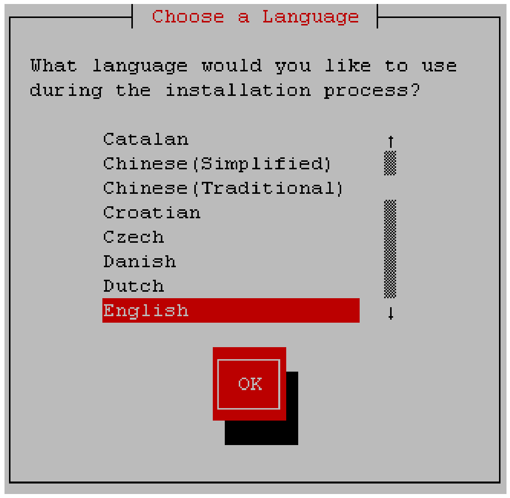


Figure 7-1 Initial screen of installer

- The `Choose a Language` screen should appear. Select your language, Tab to **OK** and press **Enter**.
- The *Installation Method* screen should appear. Choose **NFS directory** for the install method, and select **OK**.
- The *NFS Setup* screen should appear. Enter the IP address of the PC NFS server on the first line, then the path to the installation tree on the second line, and select **`OK`**. See the example in Figure 7-2 which uses the NFS server at IP address ***`9.60.18.240`***:

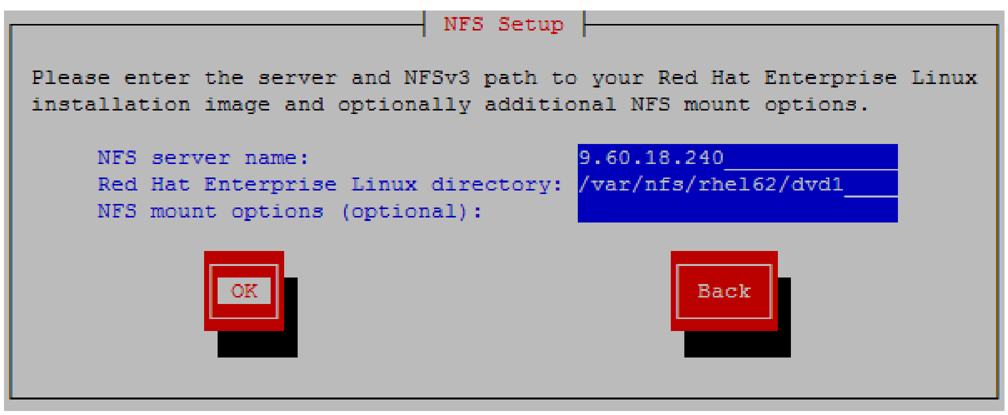


Figure 7-2 NFS setup screen

- Now the block mode 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 1.2 for zSeries

detecting hardware...
waiting for hardware to initialize...
detecting hardware...
waiting for hardware to initialize...
Running anaconda 13.21.142, the Red Hat Enterprise Linux system installer - please
wait.
14:55:55 Starting VNC...
14:55:56 The VNC server is now running.
14:55:57
```

```
You chose to execute vnc with a password.


14:55:57 Please manually connect your vnc client to gpok151.endicott.ibm.com:1
(9.60.18.151) to begin the install.
14:55:57 Starting graphical installation.
```

- Start a VNC client (e.g. 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 `LNXADMIN PARM-RH6` file (**12345678** in the sample file). In the following example, Linux is being installed with the IP address ***`9.60.18.151`***:

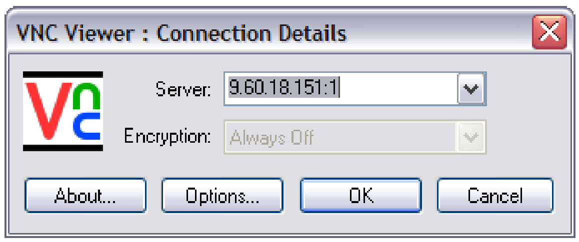


Figure 7-3 Connecting with VNC client

7.1.5 Stage 2 of the RHEL 6.2 installation

After you have connected using VNC, perform the following steps:

- A splash screen appears as shown in the top half of Figure 7-4 on page 119. Click **`Next`**.
- You will be asked what type of devices to use as shown in the bottom half of Figure 7-4 on page 119. For DASD devices, choose **Basic Storage Devices** and click **Next**.

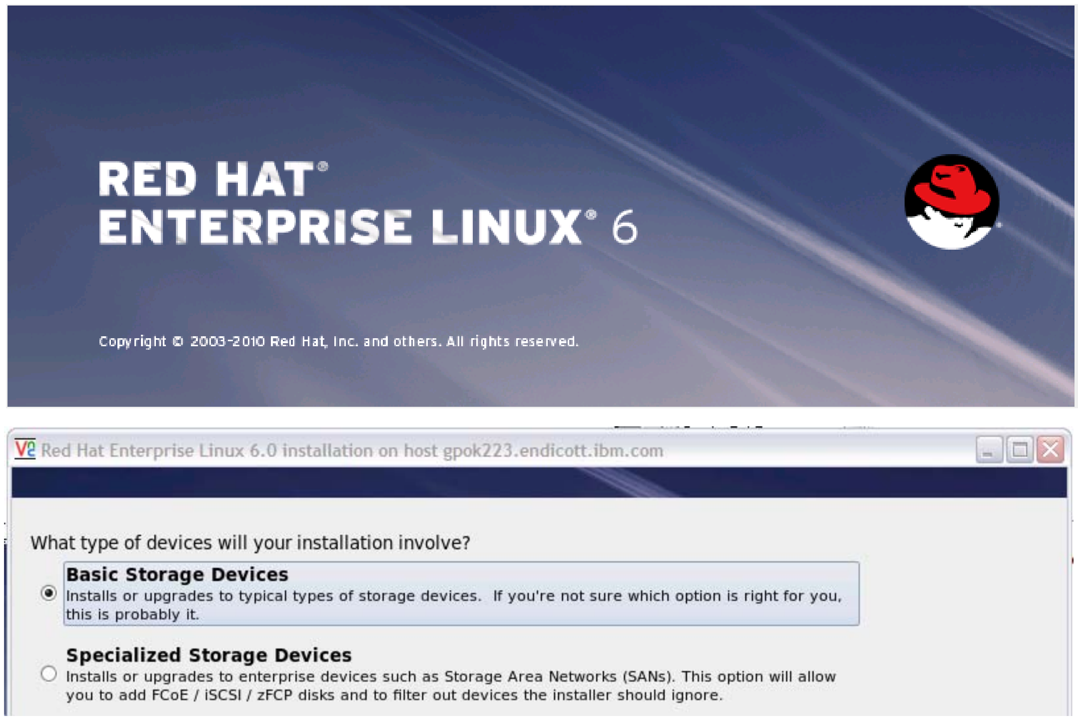


Figure 7-4 Splash screen and device type screen

- Click the button **Yes, discard any data** when prompted to initialize the virtual disk at 300 as shown in Figure 7-5 on page 119. You will still be able to use 300 and 301 as swap devices.

Figure 7-5 Re-initializing disks

- On the next window the host name, read from the configuration file, should be correct. Click **Next**.
- On the next window, select your time zone and click **Next**.
- On the next window, set the root password and click **Next**. Don't forget it!
- The installer now asks for the type of installation. Select the **Create Custom Layout** radio button as shown in Figure 7-6 on page 120. Click **Next**.

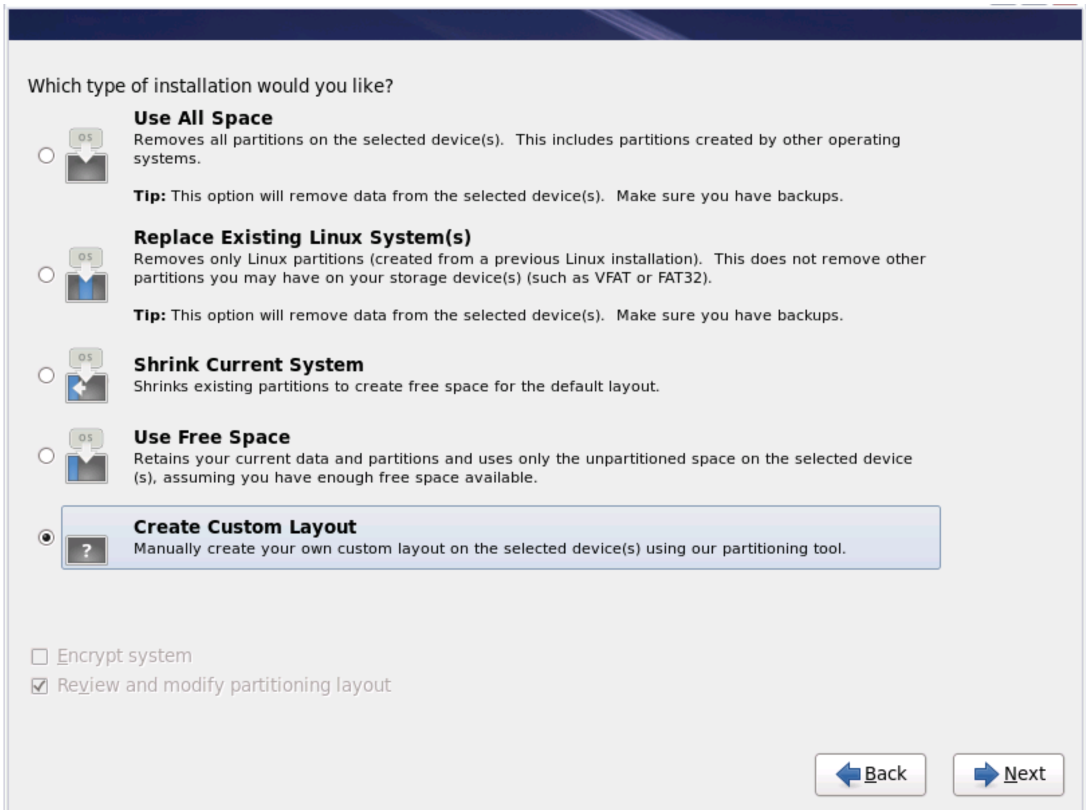


Figure 7-6 Creating custom disk layout

- The *Please Select A Device* screen allows you to set up mindisks and virtual disks. You should see all minidisks and virtual disks as shown on the left side of Figure 7-7 on page 121:

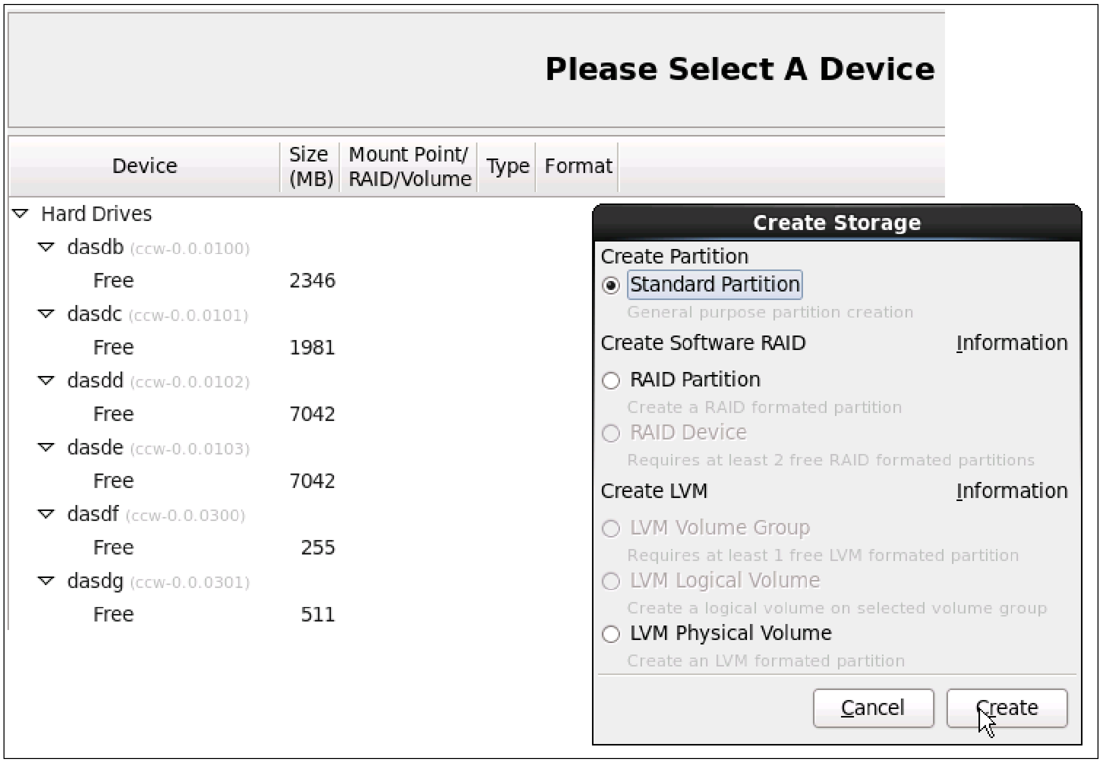


Figure 7-7 Select a device window

- Click the **Create** button and a *Create Storage* window appears as shown on the left side of Figure 7-7.
- Accept the default of **Standard Partition** and click **Create**.
- On the *Add Partition* screen, choose **swap** for *File System Type*. Deselect all *Allowable Drives* except `/dev/dasdb`. Set to a *Fixed size* to 512 (MB) as shown on the left side of Figure 7-8. This will create a swap device on minidisk. Click **OK**:

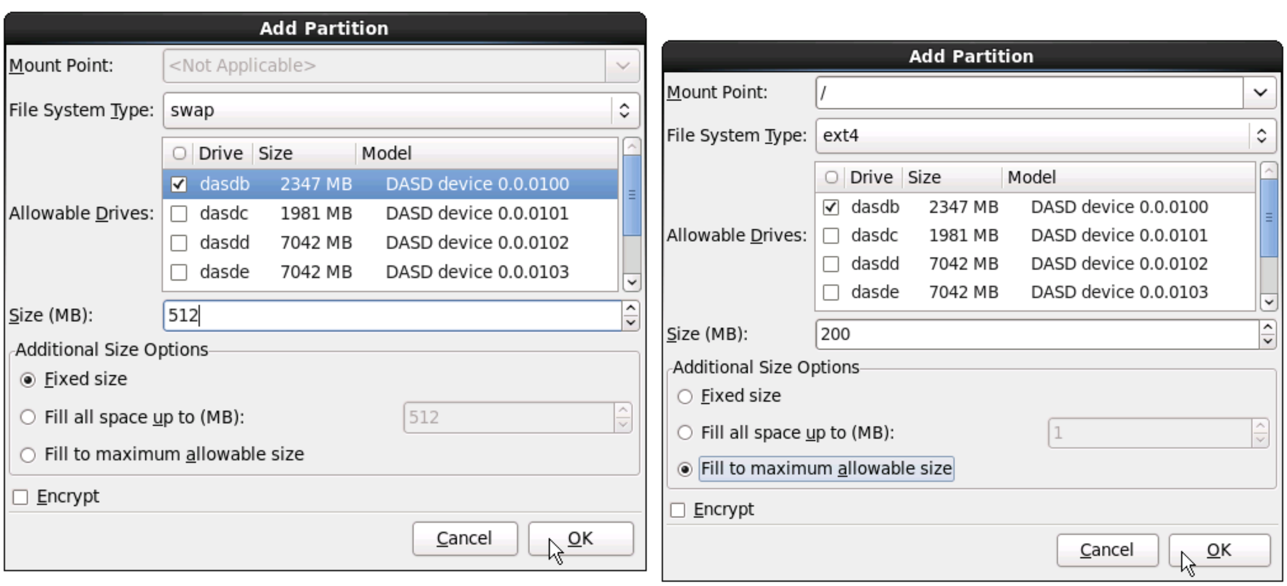


Figure 7-8 Creating a swap partition and the root file system on /dev/dasdb

- Back at the *Please Select a Device* panel, click **Create**.
- On the *Create Storage* window, accept the default of **Standard Partition**.
- In the *Add a Partition* window, use the remaining space on `/dev/dasdb` for the root file system as shown on the right side of Figure 7-8 on page 121.
- Use the **Create** button to create a LVM physical volume from `/dev/dasdc` by performing the following steps:
  – Select the **LVM Physical Volume** radio button on the *Create Storage* panel and click **Create**.
  – On the *Add a Partition* panel, deselect all *Allowable Drives* except **`dasdc`**.
  – On the *Additional Size Options*, select the radio button **Fill to maximum allowable size**
  – Click **OK**.
- Repeat the previous step and create an LVM physical volume from `/dev/dasdd` and `/dev/dasde`.
- Finally, create two more swap spaces from the virtual disks, using the maximum allowable size, on devices `/dev/dasdf` and `/dev/dasdg`. After you have done these steps, your setup should look like what is shown in Figure 7-9:

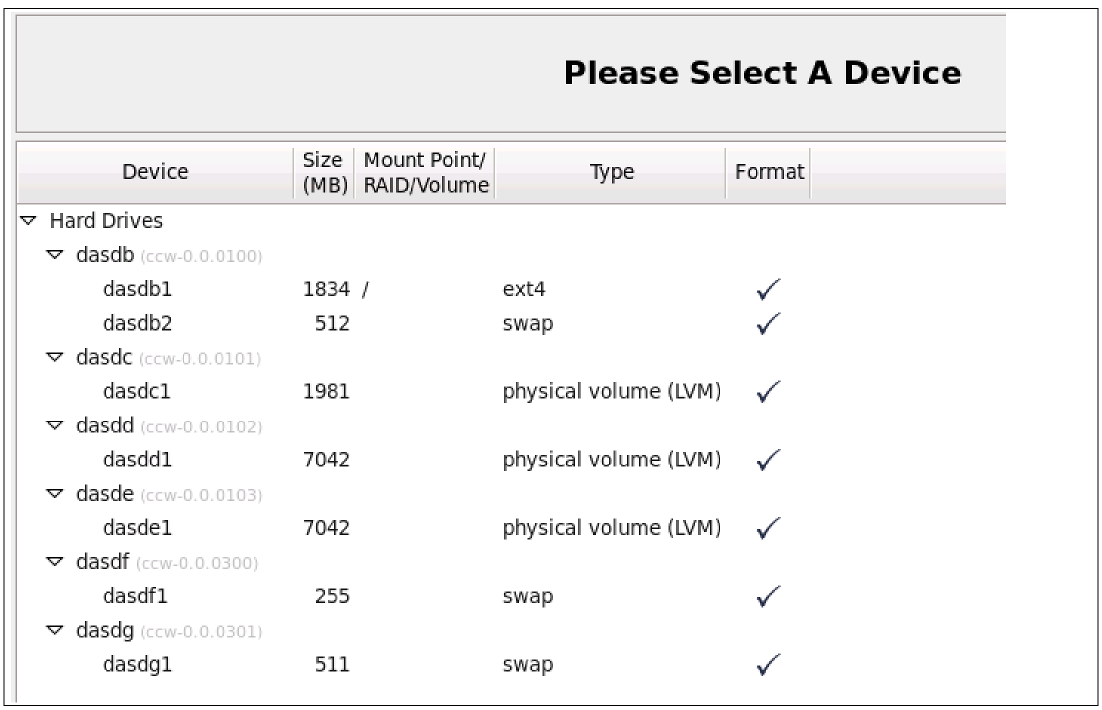


Figure 7-9 Disks and swap spaces before creating a volume group

- The next step is to set up LVM. Perform the following steps:
  – Click **Create** again on the *Please Select A Device* panel and the *Create Storage* panel appears.
  – On the *Create Storage* window, select the **LVM Volume Group** radio button and click **Create**. The *Make LVM Volume Group* panel appears as shown on the left side of Figure 7-10 on page 123.
  – Set the *Volume Group Name* to **var_vg**.

– Click **Add** under the *Logical Volumes* section. The *Make Logical Volume* panel appears.

– Set the *Mount Point* to **`/var`** and the L*ogical Volume Name* to **var_lv** as shown on the right side of Figure 7-10 on page 123. Click **OK**.

– In the *Make LVM Volume Group* panel, click **OK**.

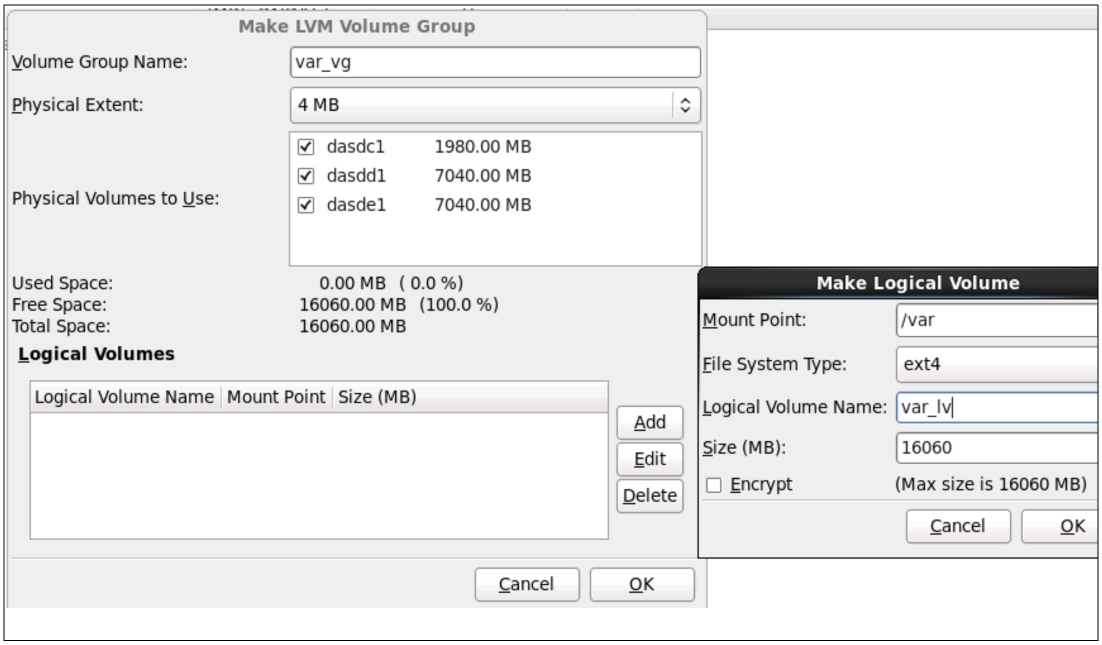


Figure 7-10 Creating a volume group and a logical volume

- You will be returned to the *Please Select A Device* panel. Click **Next**.
- On the *Format Warnings* panel, click **Format**.
- On the *Writing storage confirmation to disk* panel, click **Write changes to disk**.

Important: If you see the screen shown in Figure 7-11 on page 124, you have to start the installation over. Be sure to use **`dasdfmt`** to format the minidisks as described in 7.1.4, “Install RHEL 6.2 Linux” on page 113.

Figure 7-11 Symptom of known issue

- You will be asked for the type of software to be installed. Accept the default of **Basic Server** and click **Next**.
- The installation process will start and a progress bar will be shown. This should run for 5 - 10 minutes
- You will be prompted to reboot. Click **Reboot.**

7.1.6 Boot your new Linux system from disk

A default system should now be installed onto minidisk `100` with a large logical volume mounted over `/var/`.

- Return to your z/VM 3270 session and your newly installed system should be re-IPLing automatically.
- You system should continue to boot until a login prompt is presented.
- **Start an SSH session** as `root` to the new Linux administration system.
- Disconnect from the 3270 session:

```
==> #cp disc
```

The installation of RHEL 6.2 on the Linux administration system is now complete.

7.2 Configure the Linux administration system

Now that your Linux administration system is installed, it must be configured. The following steps are involved:

- “Copy RHEL 6.2 install tree to LNXADMIN”
- “Configure yum” on page 126
- “Turn off unneeded services” on page 127
- “Configure the VNC server” on page 128

7.2.1 Copy RHEL 6.2 install tree to LNXADMIN

Copy the RHEL 6.2 install tree to the Linux administration system, along with other files associated with this book To do so, perform the following steps:

- Create a local directory for the RHEL 6.2 install tree:

```
# cd /var/
# mkdir -p nfs/rhel62
```

- Change into that directory:

```
# cd nfs/rhel62
```

- Recursively copy the installation tree from the PC NFS server (in this example the host name is ***gpok240***) with the **scp -rp** command:

```
# scp -rp gpok240:/var/nfs/rhel62/dvd1/* .
The authenticity of host 'gpok240 (9.60.18.240)' can't be established.
RSA key fingerprint is f5:43:ce:f3:44:35:81:b2:f5:9a:5e:06:f6:fb:46:56.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'gpok240,9.60.18.240' (RSA) to the list of known hosts.
Password:
EULA                                          100% 8794     8.6KB/s   00:00
GPL                                           100%   18KB  17.7KB/s   00:00
...
repomd.xml                                    100% 4068     4.0KB/s   00:00
```

  This command will take some time, perhaps 10-15 minutes depending on a number of factors.

The RHEL 6.2 install tree should now be copied to the `LNXADMIN` system under `/var/nfs/rhel62/`.

7.2.2 Copy files associated with this book

To copy the files associated with this book to the Linux administration system, perform the following steps:

- Change directory to `/var/nfs/`:

```
# cd /var/nfs
```

- Recursively copy the files associated with this book from the PC NFS server (in this example the host name is ***gpok240***) with the **scp -rp** command:

```
# scp -rp gpok240:/var/nfs/CKB-VM62 .
Password:
...
clone-1.0-11.s390x.rpm                        100%   12KB  11.7KB/s   00:00
clone.sh                                      100%   13KB  12.6KB/s   00:00
disclaimer.txt                                100% 1461     1.4KB/s   00:00
rhel62.exec                                   100% 2424     2.4KB/s   00:00
sample.parm-rh6                               100%  168     0.2KB/s   00:00
```

```
sample.conf-rh6                                   100%  235     0.2KB/s   00:00
sample.parm-s11                                   100%  441     0.4KB/s   00:00
profile.exec                                      100% 2799     2.7KB/s   00:00
sles11s2.exec                                     100%  302     0.3KB/s   00:00
swapgen.exec                                      100%   20KB  19.9KB/s   00:00
callsm1.exec                                      100%   29KB  29.2KB/s   00:00
ssicmd.exec                                       100% 3111     3.0KB/s   00:00
chpw620.xedit                                     100% 4963     4.9KB/s   00:00
ssishutd.exec                                     100% 4260     4.2KB/s   00:00
cpformat.exec                                     100%   10KB  10.3KB/s   00:00
README.txt                                        100% 1780     1.7KB/s   00:00
```

The files associated with this book should now be copied to the Linux administration system under `/var/nfs/CKB-VM62/`.

7.2.3 Configure yum

Configure **yum** so it can install RPMs from local install tree. To do so, perform the following steps:

- Create a file named `rhel62.repo` in the `/etc/yum.repos.d` directory:

```
# cd /etc/yum.repos.d
# vi rhel62.repo
[RHEL62]
name=Red Hat Enterprise Linux 6.2
baseurl=file:///var/nfs/rhel62/Server
```

- Import the RPM key which is included in the RHEL 6.2 DVD root directory:

```
# cd /var/nfs/rhel62
# rpm --import RPM-GPG-KEY-redhat-release
```

  The RHEL 6.2 yum repository should now be configured.

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:

```
Header V3 DSA signature: NOKEY, key ID 897da07a
```

Either the correct GPG key has not been imported, or the package itself has been altered.

- Verify the new repository (often called “repo”) is accessible with the **yum grouplist** command:

```
# yum grouplist
Loaded plugins: product-id, security, subscription-manager
Updating certificate-based repositories.
Setting up Group Process
RHEL62                                                      | 4.0 kB     00:00 ...
RHEL62/primary_db                                           | 2.5 MB     00:00 ...
RHEL62/group_gz                                             | 193 kB     00:00 ...
Installed Groups:
   Additional Development
   Base
...
```

You are now ready to use **yum** to install or upgrade an RPM package. To install a package, use **yum -y install <packagename>**. Yum will conveniently install the packages specified and automatically resolve dependencies for you. Note that you should not specify the package version on the command line, only the package name.

7.2.4 Turn off unneeded services

There are a number of services which are started in a RHEL 6.2 minimum system. In order to keep the Linux administration system as lean as possible, some of these can be turned off: To do so, perform the following steps:

- Turn off the following services with the **chkconfig** command:

```
# chkconfig iptables off
# chkconfig ip6tables off
# chkconfig auditd off
# chkconfig abrtd off
# chkconfig atd off
# chkconfig cups off
# chkconfig mdmonitor off
```

> **Note:** You should 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 on configuring **iptables** for NFS traffic, see the article located at:
>
> http://www.redhat.com/magazine/010aug05/departments/tips_tricks/
>
> Also, turning on a firewall is briefly discussed in section 11.1.3, “Turn on a firewall” on page 171.

- 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 **chkconfig** command:

```
# chkconfig --list | grep 3:on
abrt-ccpp       0:off   1:off   2:off   3:on    4:off   5:on    6:off
abrt-oops       0:off   1:off   2:off   3:on    4:off   5:on    6:off
autofs          0:off   1:off   2:off   3:on    4:on    5:on    6:off
cpi             0:off   1:on    2:on    3:on    4:on    5:on    6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
dumpconf        0:on    1:on    2:on    3:on    4:on    5:on    6:on
haldaemon       0:off   1:off   2:off   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:on    3:on    4:on    5:on    6:off
mon_statd       0:off   1:off   2:on    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
portreserve     0:off   1:off   2:on    3:on    4:on    5:on    6:off
postfix         0:off   1:off   2:on    3:on    4:on    5:on    6:off
rhnsd           0:off   1:off   2:on    3:on    4:on    5:on    6:off
rhsmcertd       0:off   1:off   2:off   3:on    4:on    5:on    6:off
rpcbind         0:off   1:off   2:on    3:on    4:on    5:on    6:off
rpcgssd         0:off   1:off   2:off   3:on    4:on    5:on    6:off
rpcidmapd       0:off   1:off   2:off   3:on    4:on    5:on    6:off
rsyslog         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
sysstat         0:off   1:on    2:on    3:on    4:on    5:on    6:off
udev-post       0:off   1:on    2:on    3:on    4:on    5:on    6:off
```

7.2.5 Configure 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. To do so, perform the following steps:

- Install the VNC server and associated packages with the following **yum** command:

```
# yum -y install tigervnc-server openmotif xterm xsetroot xorg-x11-xauth
...
Installed:
  openmotif.s390x 0:2.3.3-4.el6
  tigervnc-server.s390x 0:1.0.90-0.15.20110314svn4359.el6_1.1
  xorg-x11-server-utils.s390x 0:7.5-5.2.el6
  xorg-x11-xauth.s390x 1:1.0.2-7.1.el6
  xterm.s390x 0:253-1.el6

Dependency Installed:
  libXaw.s390x 0:1.0.6-4.1.el6            libXdmcp.s390x 0:1.0.3-1.el6
  libXmu.s390x 0:1.0.5-1.el6              libXp.s390x 0:1.0.0-15.1.el6
  libXpm.s390x 0:3.5.8-2.el6              libXxf86misc.s390x 0:1.0.2-1.el6
  libmcpp.s390x 0:2.7.2-4.1.el6           libxkbfile.s390x 0:1.0.6-1.1.el6
  mcpp.s390x 0:2.7.2-4.1.el6              xkeyboard-config.noarch 0:2.3-1.el6
  xorg-x11-fonts-misc.noarch 0:7.2-9.1.el6 xorg-x11-xkb-utils.s390x 0:7.4-6.el6

Complete!
```

- The VNC server configuration file is `/etc/sysconfig/vncservers`. Edit the file by adding one line at the bottom:

```
# cd /etc/sysconfig
# vi vncservers
...
# VNCSERVERS="2:myusername"
# VNCSERVERARGS[2]="-geometry 800x600 -nolisten tcp -localhost"
VNCSERVERS="1:root"
```

- Set a VNC password with the **vncpasswd** command. This password will be needed to connect to the VNC server:

```
# vncpasswd
Password: lnx4vm
Verify: lnx4vm
```

- Stop the firewall:

```
# service iptables stop
iptables: Flushing firewall rules: [  OK  ]
iptables: Setting chains to policy ACCEPT: filter [  OK  ]
iptables: Unloading 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 'gpok151.endicott.ibm.com:1 (root)' desktop is gpok151.endicott.ibm.com:1

Creating default startup script /root/.vnc/xstartup
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/gpok151.endicott.ibm.com:1.log

[  OK  ]
```

- The directory `/root/.vnc/` is where configuration files are kept. Change to that directory and list the files:

```
# cd /root/.vnc
# ls
gpok151.endicott.ibm.com:1.log  passwd
gpok151.endicott.ibm.com:1.pid  xstartup
```

- The file `xstartup` is the script that is run when the VNC server starts and where the window manager is set. It is recommended that you change from the Tiny window manger, **twm**, to the more usable Motif window manager, **mwm**:

```
# vi xstartup   // change last line
...
xsetroot -solid grey
vncconfig -iconic &
xterm -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
mwm &
```

- Restart the VNC server with the **service** command:

```
# service vncserver restart
Shutting down VNC server: 1:root [  OK  ]
Starting VNC server: 1:root
New 'gpok144.endicott.ibm.com:1 (root)' desktop is gpok144.endicott.ibm.com:1

Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/gpok144.endicott.ibm.com:1.log

[  OK  ]
```

- You should now be able to use the VNC client to connect to the IP address of the Linux administration system with a **:1** appended. A sample session is shown in Figure 7-12.

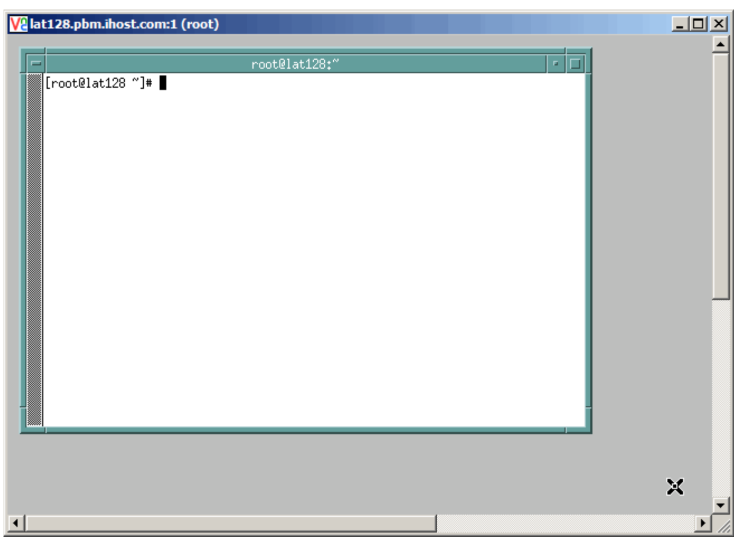


Figure 7-12 VNC client session to the VNC server

Note that the VNC server will not be started automatically across reboots. When you need a graphical environment, you can either to start the **vncserver** service manually (recommended), or you can use **chkconfig** to enable automatic startup.

7.2.6 Set system to halt on SIGNAL SHUTDOWN

By default, RHEL 6.2 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. To set the system to halt, perform the following steps:

- Edit `/etc/init/control-alt-delete.conf` changing **shutdown -r** (reboot) to **shutdown -h** (halt):

```
# cd /etc/init
# vi control-alt-delete.conf
# control-alt-delete - emergency keypress handling
#
# This task is run whenever the Control-Alt-Delete key combination is
# pressed.  Usually used to shut down the machine.

start on control-alt-delete

exec /sbin/shutdown -h now "Control-Alt-Delete pressed"
```

After that change, when the system receives a **SIGNAL SHUTDOWN** from z/VM, the following message will be displayed:

```
The system is going down for halt NOW!
```

7.2.7 Turn on the NFS server

The NFS server will be needed to export the RHEL 6.2 install tree and the files associated with this book to the other virtual servers.

Enable NFS with the following steps:

- Edit the empty file `/etc/exports` and add the following two lines:

```
# cd /etc
# vi exports
/var/nfs/rhel62            *(ro,sync)
/var/nfs/CKB-VM62          *(ro,sync)
```

These two lines will cause NFS to export:

  – The `/var/nfs/rhel62/` directory, which contains the RHEL 6.2 installation files.
  – The `/var/nfs/CKB-VM62/` directory, which has the files associated with this book.

- Set the NFS server to start at boot time and for this session.

```
# service nfs start
Starting NFS services:  [  OK  ]
Starting NFS quotas: [  OK  ]
Starting NFS daemon: [  OK  ]
Starting NFS mountd: [  OK  ]
# chkconfig nfs on
```

- Test mounting the two directories locally, then umount when successful:

```
# mount localhost:/var/nfs/rhel62 /mnt
# ls /mnt
boot.cat                  RELEASE-NOTES-es-ES.html  RELEASE-NOTES-pt-BR.html
EULA                      RELEASE-NOTES-fr-FR.html  RELEASE-NOTES-ru-RU.html
...
# umount /mnt
# mount localhost:/var/nfs/CKB-VM62 /mnt
# ls /mnt
```

```
# ls -F /mnt
disclaimer.txt  README.txt  rhel62/  sles11sp2/  vm/
# umount /mnt
```

In this section you have turned the NFS server on and exported the RHEL 6.2 install directory and the files associated with this book.

7.2.8 Configure SSH keys

SSH sessions are typically authenticated with passwords typed in from the keyboard. With SSH *key-based authentication*, sessions can be authenticated with public and private keys so that no password is needed. SSH key-based authentication can be set up from the Linux administration system (client) to the virtual servers. If the master image has a copy of Linux administration system's public key in the file `/root/.ssh/authorized_keys`, then key-based authentication will work to the cloned virtual servers.

- Create a new DSA key in the directory `/root/.ssh/`. If it does not yet exist, then first create it with the **mkdir** command.

```
# cd /root/.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:
96:19:83:28:27:84:45:01:fa:e0:c8:8e:62:b8:01:30 root@gpok222.endicott.ibm.com
The key's randomart image is:
+--[ DSA 1024]----+
|.==.             |
|o.   . .         |
|E o o . o        |
|=+ +     =       |
|oo.     S        |
|=      .         |
|=o               |
|oo               |
|.                |
+-----------------+
```

- 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 Oct 19 07:06 id_dsa
-rw-r--r--. 1 root root 619 Oct 19 07:06 id_dsa.pub
```

These files will be copied to the golden image later in the next chapter.

7.2.9 Change the order of the swap disks

It is likely that the order of swap space priority is not optimal. Perform the following commands:

- View your order with the **swapon -s** command:

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasda2                             partition       524296  0       -1
/dev/dasdb1                             partition       262132  0       -2
/dev/dasdc1                             partition       524276  0       -3
```

This shows that the minidisk swap space, /dev/dasda2, will be used before the virtual disks (which are in-memory so they should be first in the priority, from smallest to largest).

- Make a copy of the original /etc/fstab file:

```
# cd /etc
# cp fstab fstab.orig
```

- Modify the order by moving the line in /etc/fstab. with the minidisk swap space below the lines with virtual disk swap spaces:

```
# vi fstab
...
/dev/disk/by-path/ccw-0.0.0300-part1 swap                    swap    defaults        0 0
/dev/disk/by-path/ccw-0.0.0301-part1 swap                    swap    defaults        0 0
/dev/disk/by-path/ccw-0.0.0100-part2 swap                    swap    defaults        0 0
...
```

After a reboot, the minidisk swap space should come back with the lowest priority.

7.2.10 Insert vmcp module and set system to log off

The **vmcp** module is used to issue CP commands. By default it is not loaded at boot time. One way to accomplish this is to add the **modprobe vmcp** command to the file /etc/rc.d/rc.local which is run at boot time.

When Linux is shut down, the default is for the virtual machine to remain logged on even though it is not running an operating system. It is more convenient for the user ID to be logged off, both at z/VM **SHUTDOWN** time and for getting a refreshed 3270 emulator session. The file /etc/rc.d/rc.local can again be used to make two calls to the **chshut** command.

To insert the **vmcp** module and to set the virtual machine to log off at shutdown time, perform the following:

- Edit the file /etc/rc.d/rc.local and add three lines at the end as follows:

```
# cd /etc/rc.d
# vi rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local
modprobe vmcp
chshut halt vmcmd logoff
chshut poff vmcmd logoff
```

The **vmcp** command will now be available after the next reboot. The z/VM user ID should now be logged off when you halt or power off Linux.

7.2.11 Reboot the system

You should now reboot the system to test the changes:

```
# reboot
Broadcast message from root@gpok151.endicott.ibm.com
        (/dev/pts/0) at 7:27 ...

The system is going down for reboot NOW!
```

After your system comes back in a couple of minutes, start a new SSH session to the Linux administration system.

7.2.12 Verify the changes

You are now done customizing the Linux administration system. Perform the following steps to verify the configuration changes:

- Start an SSH session to LNXADMIN.
- Test the **vmcp** command with a CP command such as **QUERY NAMES**:

```
# vmcp q n
DIRMSAT2 - SSI
VSMEVSRV - DSC , VSMPROXY - DSC , VSMREQIU - DSC , VSMREQI6 - DSC
VSMREQIN - DSC , DTCSMAPI - DSC , PERSMAPI - DSC , VSMWORK3 - DSC
VSMWORK2 - DSC , VSMWORK1 - DSC , FTPSERVE - DSC , VSMGUARD - DSC
TCPIP    - DSC , DIRMAINT - DSC , DTCVSW2  - DSC , DTCVSW1  - DSC
VMSERVP  - DSC , VMSERVR  - DSC , VMSERVU  - DSC , VMSERVS  - DSC
OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
LNXADMIN - DSC
VSM     - TCPIP
```

- Confirm that three swap spaces are operational and that the minidisk swap space (/dev/dasda2) is last in priority:

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdb1                             partition       262132  0       -1
/dev/dasdc1                             partition       524276  0       -2
/dev/dasda2                             partition       524296  0       -3
```

- Verify the NFS server is running:

```
# service nfs status
rpc.svcgssd is stopped
rpc.mountd (pid 1447) is running...
nfsd (pid 1444 1443 1442 1441 1440 1439 1438 1437) is running...
rpc.rquotad (pid 1431) is running...
```

- You may also choose to test shutting down LNXADMIN from a 3270 session to **MAINT** on SSI member 1. Use the **SIGNAL SHUTDOWN** command:

```
==> signal shutdown lnxadmin
```

  From the SSH session, you should see:

```
Broadcast message from root@gpok151.endicott.ibm.com
        (unknown) at 16:57 ...

The system is going down for halt NOW!
Control-Alt-Delete pressed
```

- Wait about a minute, and query the virtual machine. That it is logged off shows that the **chshut** commands are working:

```
Ready;
... after less than one minute ...
HCPSIG2113I User LNXADMIN has reported successful termination
==> q lnxadmin
HCPCQU045E LNXADMIN not logged on
```

These steps show that the configuration changes made to the administrative Linux system have taken effect.

8

Install and configuring the RHEL 6.2 golden image

"Reality is merely an illusion, albeit a very persistent one."

— Albert Einstein

RHEL or SLES?: If you are working only with SLES 11 SP2, you can skip this chapter and follow Chapter 14, "Install the SLES 11 SP2 golden image" on page 211.

In this chapter, you will install the copy of RHEL 6.2 Linux which will be cloned. This will be referred to as the *golden image*. It should be as lean as possible so as to be a generic virtual server and to fit comfortably on two 3390-3 DASD, or about 4.4 GB.

In this section, you will perform following tasks:

- "Install the golden image" on page 135
- "Configure the golden image" on page 144

Chapters 4, 5, 6, and 7 must be completed before proceeding.

8.1 Install the golden image

The RHEL 6.2 golden image is installed onto the virtual machine `RH62GOLD`. The layer 3 virtual switch, `VSW1`, is used to demonstrate the difference from installing onto a layer 2 virtual switch.

8.1.1 Create the RH62GOLD virtual machine

In this section you will define the `RH62GOLD` virtual machine to z/VM.

- Logon to `MAINT`
- Edit the `USER DIRECT` file:

```
==> x user direct c
```

- Go to the bottom of the file and add the definition for a new virtual machine named RH62GOLD. This virtual machine is given class G privilege only. Be sure to replace the volume labels (***JM628C*** and ***JM628D*** in this example) with the labels of your DASD:

```
USER RH62GOLD LNX4VM 256M 1G G
 INCLUDE LNXDFLT
 OPTION LNKNOPAS APPLMON
 MDISK 100 3390 0001 3338 JM628C MR LNX4VM LNX4VM LNX4VM
 MDISK 101 3390 0001 3338 JM628D MR LNX4VM LNX4VM LNX4VM
*
```

This Linux virtual machine will have the following minidisks and virtual disks:

Table 8-1 Minidisks to be defined

Minidisk	Description
100-101	Minidisks used to create the root file system, and logical volumes containing the other file systems
300-301	Virtual disk swap spaces that are not defined in USER DIRECT file, but by calls to the **SWAPGEN EXEC** in the user's **PROFILE EXEC**

- Run the **DIRECTXA** command to bring the changes online:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 6 RELEASE 2.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 107 disk pages
```

You have now defined the virtual machine that will be the RHEL 6.2 golden Linux image.

8.1.2 Prepare RH62GOLD parameter files

Now that the RH62GOLD user is defined, you must create the PARM and CONF configuration files used by the RHEL 6.2 installer. To save time, you should copy the LNXADMIN PARM-RH6 and LNXADMIN CONF-RH6 files, then make the necessary changes. Perform the following steps:

- Logoff of MAINT if you are still logged on.
- Logon to LNXMAINT.
- The files LNXADMIN PARM-RH6 and LNXADMIN CONF-RH6 should exist on the 192 (D) disk as they were copied in 5.9.4, "Copy files associated with this book" on page 83. Copy the parameter and configuration files to new files with a file name of RH62GOLD. This can be done with the following **COPYFILE** command using wild cards:

```
==> copy lnxadmin * d rh62gold = =
```

- Edit the new parameter file and change the CMSCONFFILE variable to point to the new configuration file:

```
==> x rh62gold parm-rh6
root=/dev/ram0 ro ip=off ramdisk_size=40000
CMSDASD=191 CMSCONFFILE=RH62GOLD.CONF-RH6
vnc vncpassword=12345678
```

- Change the DASD, HOSTNAME and IPADDR variables in the RH62GOLD CONF-RH6 configuration file. For these values, you may want to refer to the worksheet in section 2.8.5, "Host names worksheet" on page 23. Also, add one line with the METHOD= parameter pointing to the NFS server directory you just set up on the Linux administration system. This will preclude you from having to type in the NFS server information in the **install** SSH session. Following is an example with the values used in this book:

> **Important:** If you decided to work with only layer 2 VSWITCHes instead of one layer two and one layer 3, make one additional change below
>
> ```
> LAYER2=1
> ```
>
> This will specify to Linux that the VSWITCH associated with the first NIC starting at virtual device address 0600 is Layer 2, not layer 3.

```
==> x rh62gold conf-rh6
DASD=100-101,300-301
HOSTNAME=gpok144.endicott.ibm.com
NETTYPE=qeth
IPADDR=9.60.18.144
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
NETMASK=255.255.255.128
SEARCHDNS=endicott.ibm.com
METHOD=nfs:9.60.18.151:/var/nfs/rhel62
GATEWAY=9.60.18.129
DNS=9.0.3.1
MTU=1500
PORTNAME=DONTCARE
PORTNO=0
LAYER2=0
```

You are now ready to start the golden image installation.

8.1.3 Install RHEL 6.2 on the golden image

Install Linux onto the RH62GOLD virtual machine using the new installation tree exported from the Linux administration system.

Perform the following steps:

- Logon to RH62GOLD. You should see authorization being granted to the virtual switches and virtual NICs at 600 and 700 being created and connected. The **PROFILE EXEC** from the LNXMAINT 192 disk should prompt you to IPL minidisk 100. Since there is nothing installed yet, answer **no**.

```
LOGON RH62GOLD
00: z/VM Version 6 Release 2.0, Service Level 0000 (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 09:53:35 EDT FRIDAY 09/09/11
00: Command complete
00: NIC 0600 is created; devices 0600-0602 defined
00: NIC 0600 is connected to VSWITCH SYSTEM VSW1
00: Command complete
00: NIC 0700 is created; devices 0700-0702 defined
00: NIC 0700 is connected to VSWITCH SYSTEM VSW2
z/VM V6.2.0    2011-07-19 16:53

DMSACP723I A (191) R/O
DMSACP113S C(592) not attached or invalid device address
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
```

- The default memory size of 256 MB is not enough to install RHEL 6.2. Set the memory size to 1 GB with the CP **DEFINE STORAGE** command:

```
==> def stor 1g
00: STORAGE = 1G
00: Storage cleared - system reset.
```

- IPL CMS, and again answer **no**:

```
==> ipl cms
z/VM V6.2.0    2011-07-19 16:53

DMSACP723I A (191) R/O
DMSACP113S C(592) not attached or invalid device address
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
```

- To begin the install program, run the **RHEL62 EXEC**:

```
==> rhel62
RDR FILE 0001 SENT FROM RH62GOLD  PUN WAS 0004 RECS 100K CPY  001 A NOHOLD NOKEEP
RDR FILE 0002 SENT FROM RH62GOLD  PUN WAS 0005 RECS 0003 CPY  001 A NOHOLD NOKEEP
RDR FILE 0003 SENT FROM RH62GOLD  PUN WAS 0006 RECS 296K CPY  001 A NOHOLD NOKEEP
0000003 FILES CHANGED
0000003 FILES CHANGED
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 2.6.32-202.el6.s390x (mockbuild@s390-009.build.bos.redhat.com) (gc
c version 4.4.5 20110214 (Red Hat 4.4.5-6) (GCC) ) #1 SMP Wed Sep 21 15:34:33 ED
T 2011
setup: Linux is running as a z/VM guest operating system in 64-bit mode
...
Kernel command line: root=/dev/ram0 ro ip=off ramdisk_size=40000
                     CMSDASD=191 CMSCONFFILE=RH62GOLD.CONF-RH6
                     vnc vncpassword=12345678
...
```

- There can be many, many screens of DASD I/O messages. Use the CP **TERM MORE** command to make the 3270 screens clear instantly:

```
==> #cp term more 0 0
```

- The install system should continue to boot and you should see the following messages:

```
...
Initial configuration completed.

Starting sshd to allow login over the network.

Connect now to 9.60.18.144 and log in as user 'install' to start the installatio
n.
E.g. using: ssh -x install@9.60.18.144
For VNC or text mode, disable X11 forwarding (recommended) with 'ssh -x'.
For X11, enable X11 forwarding with 'ssh -X'.

You may log in as the root user to start an interactive shell.
```

- **Important!** The message says to log in as the user **install**, however, there is an intermediate step. There is an issue where the Red Hat installer does not recognize disks that were have been formatted with **CPFMTXA**. If you have followed all the steps in this book, this will be the case (if you have previously used **dasdfmt** to format these minidisks, you can skip this step) You must first start an SSH session, login as root and dasdfmt the disks. To do so, perform the following

– Start SSH session to the install system and log in as **root**. A password will not be required.

– Invoke the lsdasd command and observe the disks:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasdb     94:4    ECKD  ???    2347MB    ???
0.0.0101   active      dasdc     94:8    ECKD  ???    1981MB    ???
0.0.0300   active      dasdd     94:12   FBA   ???    256MB     ???
0.0.0301   active      dasde     94:16   FBA   ???    512MB     ???
```

Here, the mindisks `100-101` correspond to `dasdb` and `dasdc`.

– Format the minidisks in parallel with the following **for** loop:

```
# for i in b c
> do
>   dasdfmt -b 4096 -y -f /dev/dasd$i &
> done
[1] 615
[2] 616
```

– You may need to press **Enter** after a few minutes to see the jobs in the background complete.

```
Finished formatting the device.
Rereading the partition table... ok
Finished formatting the device.
Rereading the partition table... ok

[1]-  Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
[2]+  Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
```

- Start an SSH session to the new in-memory Linux installer and login as **install**.

```
login as: install
Welcome to the anaconda install environment 1.2 for zSeries
...
```

- On the *Choose a Language* panel, set your language, tab to *Next* and press **Enter**.
- On the *Installation Method* screen, choose **NFS directory** for the install method, and select **OK**.
- The *NFS Setup* screen should appear. Enter the IP address of the PC NFS server on the *NFS server name* line, and the path to the installation tree on the second line: `/var/nfs/rhel62/dvd1`.
- Start a VNC client session to the name in the message and enter the password set in the configuration file (**lnx4vm** in the examples):

```
14:30:11 Starting VNC...
14:30:12 The VNC server is now running.
14:30:12

You chose to execute vnc with a password.


14:30:12 Please manually connect your vnc client to gpok144.endicott.ibm.com:1
(9.60.18.144) to begin the install.
```

```
14:30:12 Starting graphical installation.
```

- A splash screen should appear. Click `Next`.
- At the screen asking for the type of devices, select **Basic Storage Devices** and click **Next**.
- A *Storage Device Warning* screen should appear as shown in Figure 8-1. Click **Yes, Discard any data**. You will still be able to use the two virtual disk swap spaces at `300` and `301`.

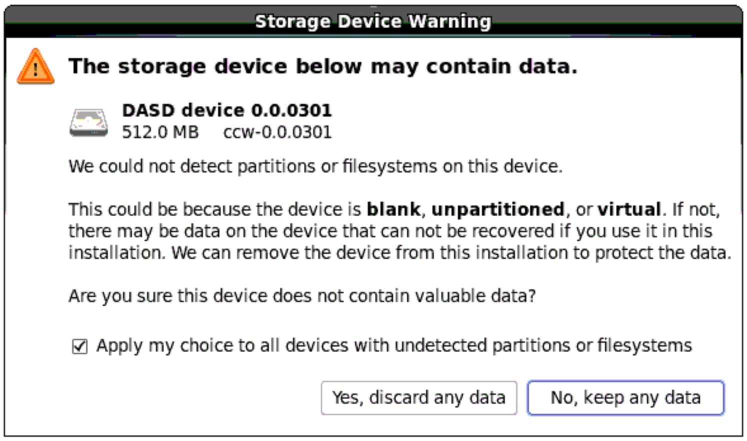


Figure 8-1 Disk initialization screen

- At the screen that sets the host name, the value read from the configuration file should be correct. Click **Next**.
- Set the time zone and click **Next**.
- Set the root password and click **Next**.
- At the type of installation screen, select **Create Custom Layout** and click **Next**. It is very important the you choose this option as described earlier.
- On the resulting windows which shows all minidisks and virtual disks, click **Create**.
- At the *Create Storage screen*, choose **Standard Partition** and click **Create**.
- At the *Add Partition screen* as shown in Figure 8-2, set the *Mount Point* to the root file system (**/**), deselect all drives except **dasdb**, and set the *Size (MB)* to **512**. Click **OK**.

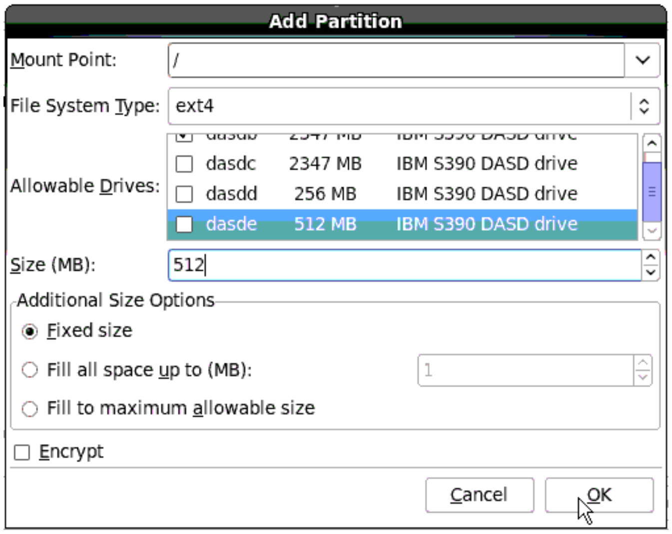


Figure 8-2 Defining the root file system

- At the *Create Storage screen*, choose **Standard Partition** and click **Create** again and create a 512 MB swap space, also on **dasdb**.
- Click **Create** again.
- At the *Create Storage* screen, select the **LVM Physical Volume** radio button and click **Create**.
- On the Add Partition window, deselect all *Allowable Drives* except **dasdb** (minidisk 100) and click the **Fill to maximum allowable size** radio button. Click **OK**.
- Repeat the previous steps so that **dasdc** (minidisk 101) is also a physical volume.
- **Create** two more **Standard partitions** with a *File System Type* of **swap** with **dasdd** (virtual disk 300) and **dasde** (virtual disk 301).
- When you return to the *Please Select A Device* screen, you should see the window shown in Figure 8-3:

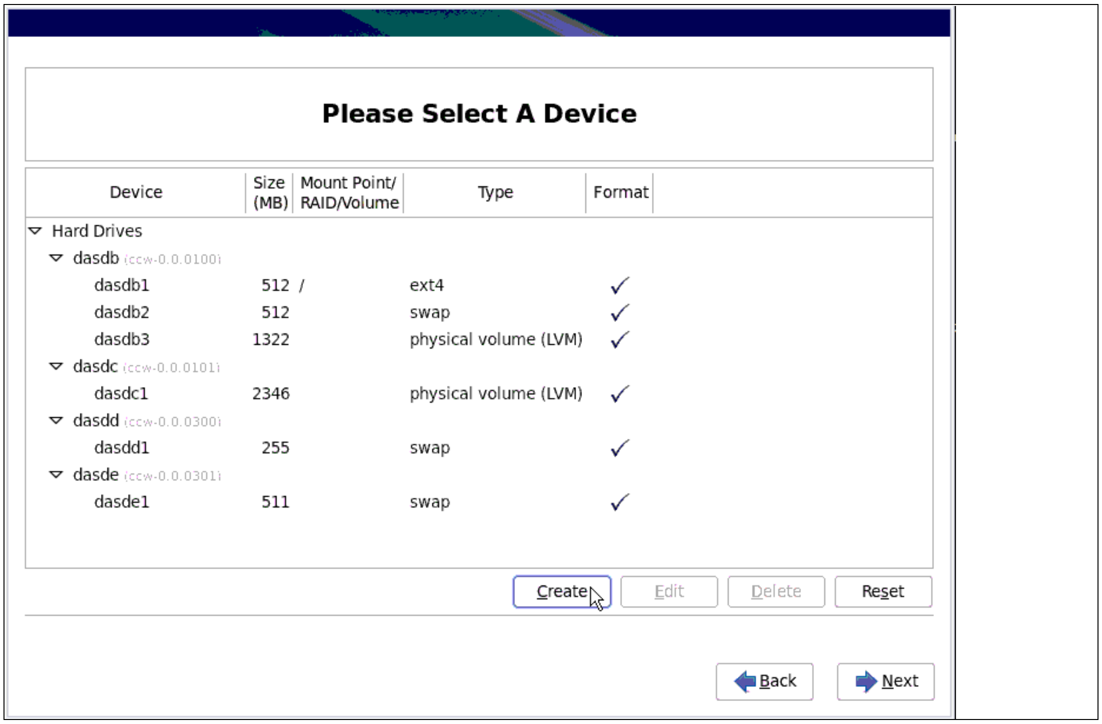


Figure 8-3 Defining file systems for logical volumes and swap spaces

- At the *Please Select a Device* window, click **Create**.
- On the *Create Storage* window choose **LVM Volume Group** and click **Create**.
- On the *Make LVM Volume Group* window, perform the following steps:
  - Set the *Volume Group Name* to **system_vg**.
  - Click **Add** in the *Logical Volumes* section.
  - In the *Make Logical Volume* window, choose a mount point of `/tmp` and a size of 256 MB.
- Repeat the above step for file other file systems mounted at `/opt`, `/var` and `/usr`. See Table 8-2 below for the recommended logical volume layout and sizes.

Table 8-2 LVM logical volume layout

Mount point	Logical Volume Name	Size (MB)
`/tmp/`	tmp_lv	256
`/opt/`	opt_lv	256
`/var/`	var_lv	384
`/usr/`	usr_lv	2048

- This results in about 720 MB of free space remaining in the volume group as shown in Figure 8-4:

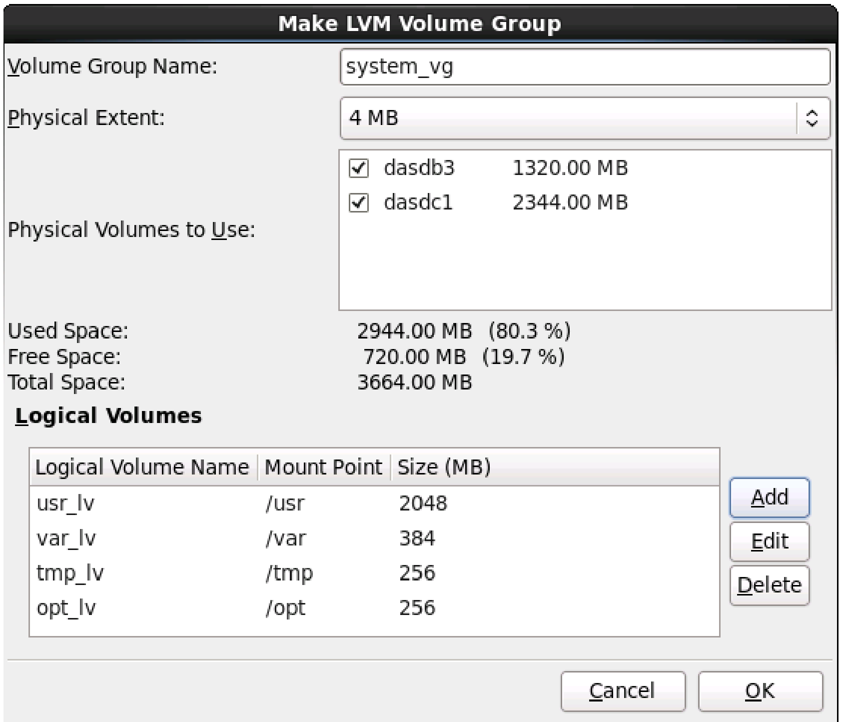


Figure 8-4 Defining a volume group and logical volumes

- At the *Please Select A Device* window, click **Next**. You will see a *Format Warnings* window. Click **Format**.

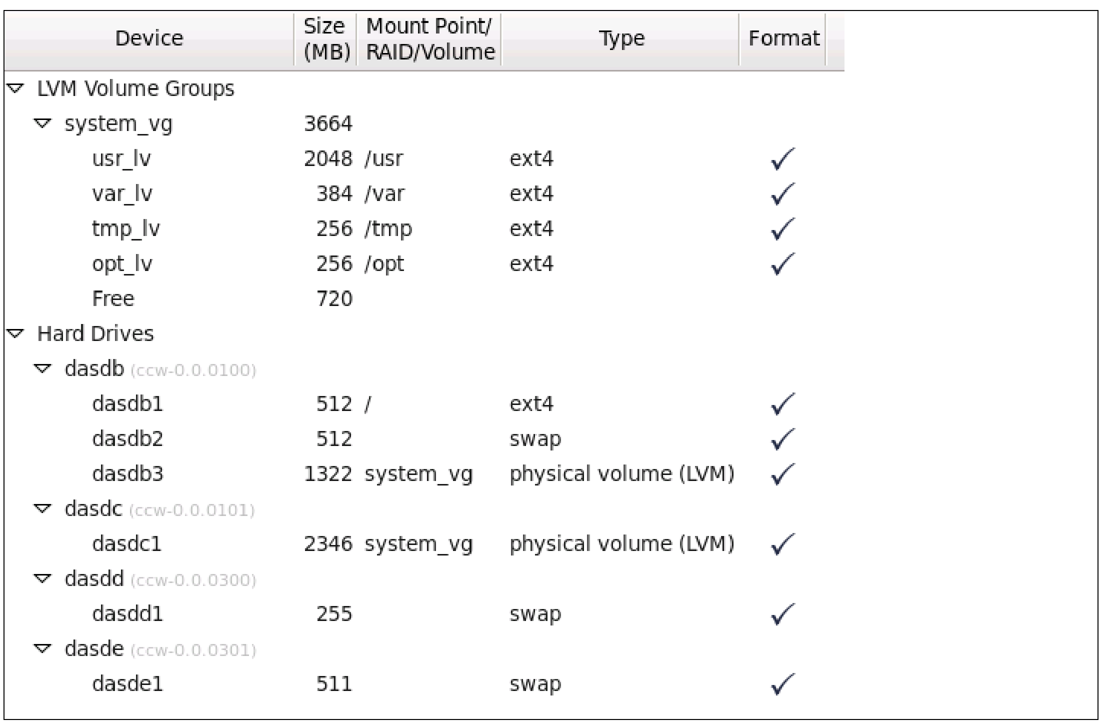


Figure 8-5 Summary of file systems and swap spaces

- On the *Format Warnings* panel, click **Format**.

- At the *Writing storage configuration to disk* window, click **Write changes to disk**. The partitions, logical volumes and file systems will be created on disk.
- At the Software options section, accept the default of a **Basic Server** and click **Next**.
- The installer will take about 5-10 minutes to install Linux. When complete, click **Reboot**. The system should be restarted from disk.
- Start an new SSH session or restart the existing one to the RHEL 6.2 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.

8.1.4 Verify the installation

Verify some settings with the following commands. You should see output similar to the following:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0300   active      dasdb     94:4    FBA   512    256MB     524288
0.0.0301   active      dasdc     94:8    FBA   512    512MB     1048576
0.0.0101   active      dasdd     94:12   ECKD  4096   2347MB    600840
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasda2                             partition       524296  0       -1
/dev/dasdb1                             partition       262132  0       -2
/dev/dasdc1                             partition       524276  0       -3
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1           504M  172M  307M  36% /
tmpfs                 498M     0  498M   0% /dev/shm
/dev/mapper/system_vg-opt_lv
                      248M   17M  220M   7% /opt
/dev/mapper/system_vg-tmp_lv
                      248M   17M  220M   7% /tmp
/dev/mapper/system_vg-usr_lv
                      2.0G  1.2G  719M  63% /usr
/dev/mapper/system_vg-var_lv
                      372M   54M  299M  16% /var
```

This shows that the two minidisks and two virtual disks are enabled devices, that the three swap spaces are active, and that all file systems are about five eighths full or less.

8.2 Configure the golden image

Customize the golden image before cloning. The following high level steps are recommended though you may add or omit some steps:

- “Configure automount of install tree” on page 145
- “Configure yum for online updates” on page 146
- “Turn off unneeded services” on page 146
- “Configure the VNC server” on page 147
- “Set system to halt on SIGNAL SHUTDOWN” on page 147
- “Configure settings at boot time” on page 147
- “Configure SSH keys” on page 148

- “Change the order of the swap disks” on page 149
- “Reboot the system” on page 149
- “Verify the changes” on page 149

8.2.1 Configure automount of install tree

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.

To configure `automount`, perform the following steps:

- Make a backup copy of the file `/etc/auto.master`,

  ```
  # cd /etc
  # cp auto.master auto.master.orig
  ```

- Edit the file and add the following line at the bottom:

  ```
  # vi auto.master   // add one line at the bottom
  ...
  #
  +auto.master
  /var/nfs    /etc/auto.nfs
  ```

  The new line specifies that the file system mounted beneath the directory `/var/nfs/` will be configured in the file `/etc/auto.nfs`.

- Create that file `auto.nfs` and add one line which points to the RHEL 6.2 install tree that is NFS-exported:

  ```
  # vi auto.nfs
  rhel62     -ro,hard,intr 9.60.18.151:/var/nfs/rhel62
  ```

  This line specifies that beneath `/var/nfs/` (in `auto.master`), when the directory `rhel62/` (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 `/var/nfs/rhel62/` directory. Even though this directory does not exist as a local file system, it is automatically mounted when referenced:

  ```
  # mkdir /var/nfs
  # service autofs reload
  Reloading maps
  ```

- Invoke the following two commands:

  ```
  # ls /var/nfs
  # ls /var/nfs/rhel62
  boot.cat                  RELEASE-NOTES-en-US.html  RELEASE-NOTES-pt-BR.html
  ...
  ```

  Do you notice anything unusual in these two commands? This is an attribute of `automount`: the first command suggests that there are no files nor directories under `/var/nfs/`, however, the second command lists the contents of `/var/nfs/rhel62/`. This behavior is expected because the directory that is *auto-mounted* is not accessed with NFS until it is referenced.

The directory `/var/nfs/rhel62/` should now be automatically mounted when referenced.

8.2.2 Configure yum for online updates

Configure **yum** so it can install RPMs from the *auto-mounted* install tree. The configuration is identical to the Linux administration system because in both instances the install tree is in the directory /var/nfs/rhel62/. However on the Linux administration system this directory is local, while on the golden image (and later the clones) the directory is *auto-mounted*. To configure **yum**, perform the following steps:

- You could create a file named rhel62.repo in the /etc/yum.repos.d directory again, or you could copy the same file from the Linux administration system that you created previously. Use the **scp** command to copy the existing file:

```
# cd /etc/yum.repos.d
# scp gpok151:/etc/yum.repos.d/rhel62.repo .
The authenticity of host 'gpok151 (9.60.18.151)' can't be established.
RSA key fingerprint is a0:a8:85:f6:01:62:11:de:77:1d:8b:5e:b9:c6:5f:b1.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'gpok151,9.60.18.151' (RSA) to the list of known hosts.
root@gpok151's password:
rhel62.repo                                   100%   81     0.1KB/s   00:00
```

- Type the file to verify the contents with the **cat** command:

```
# cat rhel62.repo
[RHEL62]
name=Red Hat Enterprise Linux 6.2
baseurl=file:///var/nfs/rhel62/Server
```

- 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 /var/nfs/rhel62/RPM-GPG-KEY-redhat-release
```

The last command again shows that automount is working. The **yum** tool should now be configured. It will be tested in the next section.

8.2.3 Turn off unneeded services

As with the golden image, follow the steps in 7.2.4, “Turn off unneeded services” on page 127. Following is a summary:

```
# chkconfig iptables off
# chkconfig ip6tables off
# chkconfig auditd off
# chkconfig abrtd off
# chkconfig atd off
# chkconfig cups off
# chkconfig mdmonitor off
```

Verify these service are turned off with the **chkconfig --list** command:

```
# chkconfig --list | grep 3:on
abrt-ccpp       0:off   1:off   2:off   3:on    4:off   5:on    6:off
abrt-oops       0:off   1:off   2:off   3:on    4:off   5:on    6:off
autofs          0:off   1:off   2:off   3:on    4:on    5:on    6:off
cpi             0:off   1:on    2:on    3:on    4:on    5:on    6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
dumpconf        0:on    1:on    2:on    3:on    4:on    5:on    6:on
haldaemon       0:off   1:off   2:off   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:on    3:on    4:on    5:on    6:off
```

```
mon_statd       0:off   1:off   2:on    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
portreserve     0:off   1:off   2:on    3:on    4:on    5:on    6:off
postfix         0:off   1:off   2:on    3:on    4:on    5:on    6:off
rhnsd           0:off   1:off   2:on    3:on    4:on    5:on    6:off
rhsmcertd       0:off   1:off   2:off   3:on    4:on    5:on    6:off
rpcbind         0:off   1:off   2:on    3:on    4:on    5:on    6:off
rpcgssd         0:off   1:off   2:off   3:on    4:on    5:on    6:off
rpcidmapd       0:off   1:off   2:off   3:on    4:on    5:on    6:off
rsyslog         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
sysstat         0:off   1:on    2:on    3:on    4:on    5:on    6:off
udev-post       0:off   1:on    2:on    3:on    4:on    5:on    6:off
```

8.2.4 Configure the VNC server

Configure the VNC server the same way as on the Linux administration system. Follow the same steps as described in 7.2.5, “Configure the VNC server” on page 128.

8.2.5 Set system to halt on SIGNAL SHUTDOWN

Again, RHEL 6.2 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 (shutdown).

Edit `/etc/init/control-alt-delete.conf` changing **shutdown -**r (reboot) to **shutdown -h** (halt):

```
# cd /etc/init
# vi control-alt-delete.conf
# control-alt-delete - emergency keypress handling
#
# This task is run whenever the Control-Alt-Delete key combination is
# pressed.  Usually used to shut down the machine.

start on control-alt-delete

exec /sbin/shutdown -h now "Control-Alt-Delete pressed"
```

This change will be picked up when the system is rebooted.

8.2.6 Configure settings at boot time

When Linux is shut down, the default is for the virtual machine to remain logged on even though it is not running an operating system. It is more convenient for the virtual machine to be logged off, both at z/VM **SHUTDOWN** time and for getting a refreshed 3270 emulator session. This can be modified with the **chshut** command and the parameters **halt** and **poff** (power off).

In order for z/VM 6.2 to relocate guests between SSI members, there must not be any links to CMS disks. The `LNXDFLT PROFILE` creates links to mindisks and assigns virtual device numbers `190`, `191`, `19D` and `19E`. These addresses must be detached. The **vmcp** module is used to accomplish this.

To do these two tasks, perform the following steps:

- Edit the file /etc/rc.d/rc.local, which is run at boot time, and the following lines:

```
# cd /etc/rc.d
# vi rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local
chshut halt vmcmd logoff
chshut poff vmcmd logoff
modprobe vmcp
vmcp det 190
vmcp det 191
vmcp det 19d
vmcp det 19e
rmmod vmcp
```

The z/VM virtual machine should now be logged off when you halt or power off Linux.

8.2.7 Configure SSH keys

Recall that you generated SSH public and private keys on the Linux administration system in section 7.2.8, "Configure SSH keys" on page 131. Now it is time to copy these keys from the Linux administration system to the golden image.

- Change into the /root/.ssh/ directory:

```
# cd /root/.ssh
```

  If it does not exist, create it with the **mkdir** command.

- Copy the Linux administration system's public key to authorized_keys using **scp**. In this example the IP address of the Linux administration system is ***9.60.18.151***:

```
# scp 9.60.18.151:/root/.ssh/id_dsa.pub authorized_keys
root@9.60.18.151's password:
id_dsa.pub                                    100%  619     0.6KB/s   00:00
```

  This will allow the Linux administration system to initiate an encrypted SSH connection to the Linux server using *key-based authentication* (so there will be no need to type the root password).

- Switch to, or start and SSH session on the Linux administration system. Use the **ssh** command to the golden image to issue the **hostname** command. You should not be prompted for a password:

```
[root@gpok151 .ssh]# ssh gpok144 hostname
The authenticity of host 'gpok144 (9.60.18.144)' can't be established.
RSA key fingerprint is d3:e9:1c:30:5e:2c:db:37:2d:07:7d:0a:5e:47:8a:2c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'gpok144,9.60.18.144' (RSA) to the list of known hosts.
gpok144.endicott.ibm.com
```

  You should not have to type the RHEL 6.2 golden image's password.

You should now have key-based authentication from the Linux administration system to the golden image configured.

8.2.8 Change the order of the swap disks

It is likely that the order of swap space priority is not optimal. Perform the following commands:

- View your order with the **swapon -s** command:

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasda2                             partition       524296  0       -1
/dev/dasdb1                             partition       262132  0       -2
/dev/dasdc1                             partition       524276  0       -3
```

  This shows that the minidisk swap space will be used before the virtual disk.

- Make a copy of the original /etc/fstab file:

```
# cd /etc
# cp fstab fstab.orig
```

- Modify the order by moving the line in /etc/fstab. with the minidisk swap space below the lines with virtual disk swap spaces:

```
# vi fstab
...
/dev/disk/by-path/ccw-0.0.0300-part1 swap                  swap    defaults        0 0
/dev/disk/by-path/ccw-0.0.0301-part1 swap                  swap    defaults        0 0
/dev/disk/by-path/ccw-0.0.0100-part2 swap                  swap    defaults        0 0
...
```

After a reboot, the minidisk swap space should come back with the lowest priority.

8.2.9 Other configuration changes

You may consider other configuration changes. Of course you can take an iterative approach: start with this set of changes, clone some Linux images and test, then bring the golden image back up, make more changes and re-clone.

Whether you're on the first pass of configuration or not, refer to the following sections to consider other changes for performance and availability related issues:

- 12.1, "Register your system with RHN" on page 185
- 20.6, "Set up Memory Hotplugging" on page 342
- 20.8, "Hardware cryptographic support for OpenSSH" on page 348

8.2.10 Reboot 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!
```

8.2.11 Verify the changes

You are now done customizing the RHEL 6.2 golden image. When the system comes back, perform the following steps to verify the changes that you made.

- Start an SSH session to the golden image as **root** and check a few settings:

- Use the **df** command to display your file systems:

```
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1           504M  161M  318M  34% /
tmpfs                 498M     0  498M   0% /dev/shm
/dev/mapper/system_vg-opt_lv
                      248M   17M  220M   7% /opt
/dev/mapper/system_vg-tmp_lv
                      248M   17M  219M   7% /tmp
/dev/mapper/system_vg-usr_lv
                      2.0G  1.2G  703M  64% /usr
/dev/mapper/system_vg-var_lv
                      372M   82M  272M  24% /var
```

- Confirm that your swap spaces are operational and that the two virtual disks have priority over the one minidisk:

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdb1                             partition       262132  0       -1
/dev/dasdc1                             partition       524276  0       -2
/dev/dasda2                             partition       524296  0       -3
```

- Verify the shutdown settings with the **lsshut** command:

```
# lsshut
Trigger          Action
========================
Halt             vmcmd ("logoff")
Panic            stop
Power off        vmcmd ("logoff")
Reboot           reipl
```

You may choose to confirm other settings.

Congratulations! You have now successfully installed and configured 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

Configure RHEL 6.2 for cloning

“It has become appallingly obvious that our technology has exceeded our humanity.”

--Albert Einstein

RHEL or SLES?: If you are working only with SLES 11 SP2, you can skip this chapter and follow Chapter 15, “Clone SLES 11 SP2” on page 237.

By now you should have installed and configured `LNXADMIN`, the Linux administration system, and `RH62GOLD`, the *golden image*. The Linux administration system must be up and running. In this chapter, the following tasks are described:

- “Define two new virtual machines” on page 151
- “Clone a virtual server manually” on page 152
- “Clone a virtual server automatically” on page 157
- “Review system status” on page 161

A virtual server is cloned manually so you will better understand the steps, and it is cloned automatically to speed up the process.

9.1 Define two new virtual machines

In this section you will define two new virtual machines that will be targets to clone to. In this example, they are named `LINUX153` and `LINUX157` because the last dotted decimal octet in the IP address associated with them are 153 and 157.

To do so, perform the following steps:

- Logon to `MAINT` and edit the `USER DIRECT` file to add new virtual machines for Linux.

  ```
  ==> x user direct c
  ```

- Go to the bottom of the file and add the following lines. In this example the user IDs will be `LINUX153` and `LINUX157` with a passwords of `LNX4VM`. They will default to have 256MB of memory but can be set up to 1GB. They will have only G privilege class (General user)

They will each have two 3338 cylinder (about 2.2 GB each) minidisks. In this example, the 3390-3s assigned have labels of ***JM6294***, ***JM6327***, ***JM6328*** and ***JM6339***. These disks were formatted in section 5.8.4, “Format DASD for minidisks” on page 75

```
*
USER LINUX153 LNX4VM 256M 1G G
 INCLUDE LNXDFLT
 MDISK 100 3390 0001 3338 JM6294 MR LNX4VM LNX4VM LNX4VM
 MDISK 101 3390 0001 3338 JM6327 MR LNX4VM LNX4VM LNX4VM
*
USER LINUX157 LNX4VM 256M 1G G
 INCLUDE LNXDFLT
 MDISK 100 3390 0001 3338 JM6328 MR LNX4VM LNX4VM LNX4VM
 MDISK 101 3390 0001 3338 JM6339 MR LNX4VM LNX4VM LNX4VM
```

- You may need to add the new volumes to the `$ALLOC$` virtual machine so cylinder 0 won’t show up in the disk map as a gap.
- Again check for gaps and overlaps.

```
==> diskmap user
==> x user diskmap
====> all /gap/|/overlap/
...
```

- Bring the changes online with the **DIRECTXA** command:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 6 RELEASE 2.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 107 disk pages
```

Two new virtual machines have now been created.

9.2 Clone 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 virtual machine, `RH62GOLD` in this example, has a root file system on LVM, located on minidisks `100-101`.
- The target virtual machine, `LINUX153`, 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 the Linux **dasdfmt** and **dd** commands will also work.

Given these assumptions, one set of steps that can be used to clone Linux is as follows:

- “Link the source and target disks” on page 153.
- “Copy the source to the target disks” on page 153
- “Modify the new root file system” on page 154
- “IPL the target system” on page 156

9.2.1 Link the source and target disks

To link the source and target disks, perform the following steps:

- Shutdown the RHEL 6.2 golden image if it is still running:

```
# shutdown -h now
Broadcast message from root@gpok144.endicott.ibm.com
        (/dev/pts/0) at 13:39 ...

The system is going down for halt NOW!
```

  The virtual machine should automatically be logged off.

- Start an SSH session to the Linux administration system, `gpok151` in this example, as `root`.
- Link the source disks, `RH62GOLD 100` and `101` read-only as virtual devices `1100` and `1101` with the **`LINK`** command:

```
# vmcp link rh62gold 100 1100 rr
# vmcp link rh62gold 101 1101 rr
```

Link The target disks, `LINUX153 100` and `101`, multi-read (read-write if no other virtual machine has write access) as virtual devices `2100` and `2101`:

```
# vmcp link linux153 100 2100 mr
# vmcp link linux153 101 2101 mr
```

9.2.2 Copy the source to the target disks

To copy the source to the target disks, perform the following steps:

- **If you have** the **`FLASHCOPY`** feature, use it with the **`vmcp`** command:

```
# vmcp flashcopy 1100 0 end to 2100 0 end
Command complete: FLASHCOPY 1100 0 3337 TO 2100 0 3337
# vmcp flashcopy 1101 0 end to 2101 0 end
Command complete: FLASHCOPY 1100 0 3337 TO 2100 0 3337
```

- **If you do not have** the **`FLASHCOPY`** feature, enable the `1100-1101` and `2100-2101` disks with the **`chccwdev -e`** command, then determine the newly created device nodes with the **`lsdasd`** command:

```
# chccwdev -e 1100-1101,2100-2101
Setting device 0.0.1100 online
Done
...
# lsdasd
...
0.0.1100   active     dasdf     94:20   ECKD  4096   2347MB    600840
0.0.1101   active     dasdg     94:24   ECKD  4096   2347MB    600840
0.0.2100   active     dasdh     94:28   ECKD  4096   2347MB    600840
0.0.2101   active     dasdi     94:32   ECKD  4096   2347MB    600840
```

  – In this example the source minidisks (`1100-1101`) are named `/dev/dasdf` and `/dev/dasdg`, while the target minidisks (`2100-2101`) are named `/dev/dasdh` and `/dev/dasdi`. Format the target devices with the **`dasdfmt`** command using a 4096 byte (4KB) block size:

```
# dasdfmt -b 4096 -y -f /dev/dasdh
Finished formatting the device.
Rereading the partition table... ok
# dasdfmt -b 4096 -y -f /dev/dasdi
```

```
...
```

– Now that the devices have been formatted, you can copy the volumes of the golden image with the **dd** command, again using a block size of 4K (4096) bytes:

```
# dd if=/dev/dasdf of=/dev/dasdh bs=4096
...
# dd if=/dev/dasdg of=/dev/dasdi bs=4096
...
```

– Bring the devices offline so the new file systems will be recognized when brought back online:

```
# chccwdev -d 1100-1101,2100-2101
...
```

- Detach the source disks as they are no longer needed:

```
# vmcp det 1100-1101
1100-1101 DETACHED
```

9.2.3 Modify the new root file system

To modify the newly copied root file system, perform the following steps:

- The target root file system is on the `LINUX153 100` disk which is linked as virtual address `2100`. Activate it with the **chccwdev -e** command:

```
# chccwdev -e 2100
Setting device 0.0.2100 online
Done
```

- Use the **lsdasd** command to show the minidisks that are accessible. The target root file system is on the disk accessed as virtual device address `2100`:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0300   active      dasdb     94:4    FBA   512    256MB     524288
0.0.0301   active      dasdc     94:8    FBA   512    512MB     1048576
0.0.0103   active      dasdd     94:12   ECKD  4096   7042MB    1802880
0.0.0101   active      dasde     94:16   ECKD  4096   1981MB    507240
0.0.0102   active      dasdf     94:20   ECKD  4096   7042MB    1802880
0.0.2100   active      dasdg     94:24   ECKD  4096   2347MB    600840
```

- The target disk device is `/dev/dasdg` and the target disk first partition is `/dev/dasdg1`. Mount the target root file system over the directory `/mnt/`:

```
# mount /dev/dasdg1 /mnt
```

- Observe that this appears to be a root file system:

```
# ls /mnt
bin   cgroup  etc   lib    lost+found  mnt  proc  sbin     srv  tmp  var
boot  dev     home  lib64  media       opt  root  selinux  sys  usr
```

- The networking values that must be changed are the IP address and host name in the files `/etc/sysconfig/network` and `/etc/sysconfig/network-scripts/ifcfg-eth0`.
- Observe the contents of these files on the newly copied disks:

```
# cat /mnt/etc/sysconfig/network
NETWORKING=yes
HOSTNAME=gpok144.endicott.ibm.com
GATEWAY=9.60.18.129
# cat /mnt/etc/sysconfig/network-scripts/ifcfg-eth0
```

```
DEVICE="eth0"
BOOTPROTO="static"
DNS1="9.0.3.1"
DOMAIN="endicott.ibm.com"
GATEWAY="9.60.18.129"
IPADDR="9.60.18.144"
MTU="1500"
NETMASK="255.255.255.128"
NETTYPE="qeth"
NM_CONTROLLED="yes"
ONBOOT="yes"
OPTIONS="layer2=1 portno=0"
PORTNAME="DONTCARE"
SUBCHANNELS="0.0.0700,0.0.0701,0.0.0702"
TYPE="Ethernet"
```

- Change the host name on the target disks in the file `/etc/sysconfig/network`:

```
# cd /mnt/etc/sysconfig
# vi network
NETWORKING=yes
HOSTNAME=gpok153.endicott.ibm.com
GATEWAY=9.60.18.129
```

- Change the IP address in the file `/etc/sysconfig/network-scripts/ifcfg-eth0`:

```
# cd network-scripts
# vi ifcfg-eth0
DEVICE="eth0"
BOOTPROTO="static"
DNS1="9.0.3.1"
DOMAIN="endicott.ibm.com"
GATEWAY="9.60.18.129"
IPADDR="9.60.18.153"
MTU="1500"
NETMASK="255.255.255.128"
NETTYPE="qeth"
NM_CONTROLLED="yes"
ONBOOT="yes"
OPTIONS="layer2=0 portno=0"
PORTNAME="DONTCARE"
SUBCHANNELS="0.0.0600,0.0.0601,0.0.0602"
```

- Now that the target disks have been copied and modified, they can be detached. Change to the default directory with the **cd** command, use the **sync** command to flush the disks and the **umount** command to unmount the modified root file system:

```
# cd
# sync
# umount /mnt
```

- Set the `LINUX153 100` disks offline with the **chccwdev** command:

```
# chccwdev -d 2100
Setting device 0.0.2100 offline
Done
```

- Detach the two target disks using the CP **DETACH** command:

```
# vmcp det 2100-2101
2100 DETACHED
```

You should now be ready to IPL the manually cloned system.

9.2.4 IPL the target system

The modified target system should now be ready to IPL. To do so, perform the following steps:

- Logon to LINUX153.

```
LOGON LINUX153
00: z/VM Version 6 Release 2.0, Service Level 0000 (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 14:42:30 EDT FRIDAY 09/09/11
00: Command complete
00: NIC 0600 is created; devices 0600-0602 defined
00: NIC 0600 is connected to VSWITCH SYSTEM VSW1
00: Command complete
00: NIC 0700 is created; devices 0700-0702 defined
00: NIC 0700 is connected to VSWITCH SYSTEM VSW2
z/VM V6.2.0    2011-07-19 16:53
```

- Press **Enter** to IPL CMS. The **PROFILE EXEC** will ask you if you want to IPL from minidisk 100. Type **y** for yes and Linux should boot:

```
DMSACP723I A (191) R/O
DMSACP113S C(592) not attached or invalid device address
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
y
000: zIPL v1.8.2-40.el6 interactive boot menu
00:
00:  0. default (linux-2.6.32-202.el6.s390x)
00:
00:  1. linux-2.6.32-202.el6.s390x
00:
00: Note: VM users please use '#cp vi vmsg <input>'
00:
00: Please choose (default will boot in 5 seconds):
00: Booting default (linux-2.6.32-202.el6.s390x)...
...
Red Hat Enterprise Linux Server release 6.2 Beta (Santiago)
Kernel 2.6.32-202.el6.s390x on an s390x

gpok153 login:
```

Your new system should come up cleanly using the modified IP address and host name. If it does, then congratulations! 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 do it automatically. You will use the LINUX153 virtual machine again. To clone, the target virtual machine 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 LINUX153 virtual machine:

```
==> #cp log
```

9.3 Clone a virtual server automatically

Now that you have cloned a server manually and better understand the steps, you can use the **clone** script to clone automatically. To do so, perform the following steps:

- "Install the clone RPM" on page 157
- "Create a configuration file for cloning" on page 157
- "Use the clone script" on page 158

9.3.1 Install the clone RPM

Perform the following steps on the LNXADMIN installation server:

- Open an SSH session as root to the system running on LNXADMIN.
- Install the clone script RPM:

```
# rpm -ivh /var/nfs/CKB-VM62/rhel62/clone-1.0-11.s390x.rpm
Preparing...                ########################################### [100%]
   1:clone                  ########################################### [100%]
```

9.3.2 Create a configuration file for cloning

For each Linux guest you want to clone, you must create a configuration file that you can use to customize the image after cloning. To do that for LINUX153, perform the following steps:

- Review the configuration file /etc/sysconfig/clone which is used to set variables:

```
# cat /etc/sysconfig/clone
# AUTOLOG - If set to "y" the script will autolog the cloned
#           image after the cloning is completed.  If it is
#           set to "n" the image will not autolog the cloned
#           image.
AUTOLOG=y

# PROMPT  - This will set if the script should prompt the user for
#           confirmation before cloning.  If set to "y" the user
#           will be prompted to continue.  If set to "n" the script
#           will run without confirmation.
PROMPT=y

# CLONE_MNT_PT - This specifies the location on the filesystem
#                that the cloned root filesystem should be mounted
#                to.  If the directory does not exist it will be
#                created the first run.
CLONE_MNT_PT=/mnt/clone

# CLONE_METHOD - This is used to determine what method you want to use
#                for cloning.  It can have a value of AUTO, which will first
#                attempt FLASHCOPY then fall back to dd, or DD which will
#                only try to perform a Linux dd command.
CLONE_METHOD=auto

# BLACKLIST    - List of z/VM user IDs forbidden to be used as clone targets.
#                It's a good idea to add your master server here, so it doesn't
#                become a clone target by mistake.
#                Format: BLACKLIST="userA userB userC ..."
BLACKLIST=""
```

In the following example this configuration file is not modified as all defaults are acceptable.

- Copy and then edit the supplied sample configuration file to reflect the values of the new Linux system:

```
# cd /etc/clone
# cp rhel.conf.sample linux153.conf
```

- Edit the new configuration file with the appropriate values for your system. 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 IP address (`IPADDR`) and the DNS host name (`HOSTNAME`). In the following example, the IP address is set to ***9.60.18.153*** and the host name to ***gpok153.endicott.ibm.com***.

```
# vi linux153.conf
# Define the DASD that should be included as a part
# of the clone.
DASD=100,101
DASD_ROOT=100
VG_NAME=
LV_ROOT=
AUTOLOG=y

# Define networking information that will be used for the host.
IPADDR=9.60.18.153
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
HOSTNAME=gpok153.endicott.ibm.com
NETTYPE=qeth
NETMASK=255.255.255.128
NETWORK=9.60.18.128
SEARCHDNS=endicott.ibm.com
BROADCAST=9.60.18.255
GATEWAY=9.60.18.129
DNS=9.0.2.11
MTU=1500
```

Notes on the previous values:

  – The `DASD=` line sets the range of minidisks that will be copied. You can enter dashes (**-**) or commas (**,**) to specify address ranges or specific disks, respectively. Make the range following `DASD=` is one continuous block of text with no spaces added.
  – The `DASD_ROOT=` line sets the minidisk that contains the root file system.
  – The `VG_NAME=` line sets the volume group name if the root file system of the source system is on a logical volume.
  – The `LV_ROOT=` line sets the logical volume name of the root file system.
  – The `AUTOLOG=y` line specified that the target virtual machine will be started after cloning.

You are now be ready to clone to this new virtual machine.

9.3.3 Use the clone script

To use the clone script, perform the following steps:

- Go back to your an SSH session to the Linux administration system.
- Verify that the **`clone`** script is in your PATH with the **`which`** command:

```
# which clone
/usr/sbin/clone
```

- The **`clone`** script can operate in two modes. The first where the DASD information is provided on the command line, and the second where the DASD information is included in the new virtual machine's configuration file. Running clone with no arguments prints the following help information:

```
# clone
Usage: clone [-v] sourceID targetID [rootMinidisk [minidisk1 minidisk2..]]
    Switches
       -v Verbose output
    Required
       sourceID the z/VM user id you want to clone from
       targetID  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 `sourceID` is the z/VM virtual machine of the RHEL 6.2 golden image (`RH62GOLD` in this example) and `targetID` is the z/VM ID of the target (`LINUX153` in this example). These values are *always* required.

In this example, DASD is set to `100-101`, which implies that minidisks located at virtual addresses `100` and `101` are copied. The `300` and `301` virtual disks 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 file system (/).

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** via 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 via the **xautolog** command. It takes less than a minute to clone with **FLASHCOPY** support and 5-15 minutes with **dd**.

- Run the **clone** script with the verbose switch (**-v**) to add output:

```
# clone -v rh62gold linux153
Invoking CP command: QUERY rh62gold
Invoking CP command: QUERY linux153

This will copy disks from rh62gold to linux153
Host name will be:  gpok153.endicott.ibm.com
IP address will be: 9.60.18.153
Do you want to continue? (y/n): y
```

The script makes sure the golden image (source) virtual machine and the target virtual machine exist and are logged off. Then, it confirms the order of the cloning and displays information collected from the `/etc/clone/linux153.conf` file. Following this, it asks if you are sure you want to overwrite the disks on the target virtual machine.

Next, the script links to the golden image minidisk and the target minidisk. The golden image minidisks are linked 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 source `100` and `101` minidisks to the target `100` and `101` minidisks. The script then detaches the links. If **FLASHCOPY** fails, the script falls back to the Linux **dasdfmt** and **dd** commands.

```
Cloning rh62gold to linux153 ...
Copying minidisks...
Invoking CP command: QUERY VIRTUAL fffe
Invoking CP command: LINK rh62gold 100 fffe RR
Invoking CP command: QUERY VIRTUAL ffff
Invoking CP command: LINK linux153 100 ffff W
Invoking CP command: FLASHCOPY fffe 0 END ffff 0 END
100 disk copied ...
Invoking CP command: DETACH fffe
Invoking CP command: DETACH ffff
Invoking CP command: QUERY VIRTUAL fffe
```

```
Invoking CP command: LINK rh62gold 101 fffe RR
Invoking CP command: QUERY VIRTUAL ffff
Invoking CP command: LINK linux153 101 ffff W
Invoking CP command: FLASHCOPY fffe 0 END ffff 0 END
101 disk copied ...
Invoking CP command: DETACH fffe
Invoking CP command: DETACH ffff
```

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 linux153 100 ffff W
Modifying networking info under /mnt/clone...
Regenerating SSH keys in /mnt/clone/etc/ssh/ ...
Invoking CP command: DETACH ffff
Invoking CP command: XAUTOLOG linux153
Booting linux153
Successfully cloned rh62gold to linux153
```

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: In the final section, the `LINUX153` virtual machine is logged on via **`XAUTOLOG`**. Because the shared `PROFILE EXEC` detects that the virtual machine is in a disconnected mode, it carries out an IPL of Linux from minidisk `100`.

- Start an SSH session to the newly cloned Linux server.

> **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.

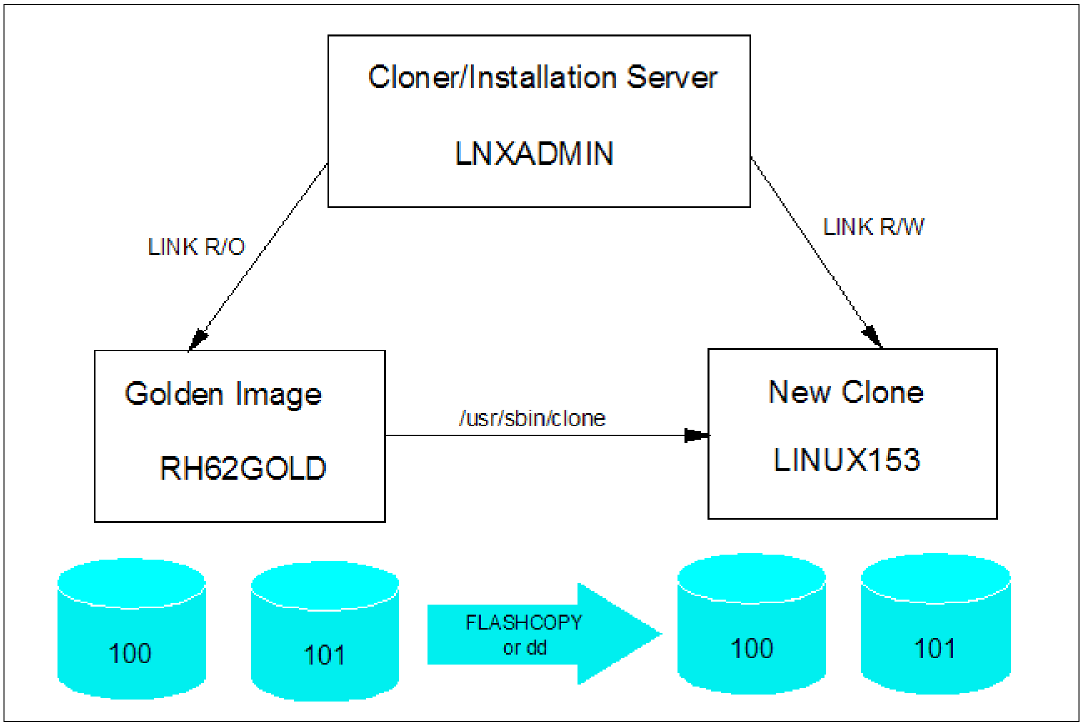


Figure 9-1 Cloning block diagram

The top of the figure shows the Linux administration system that is running on the `LNXADMIN` virtual machine. In order to **`FLASHCOPY`** or **`dd`**, the `LNXADMIN` virtual machine requires a **`LINK`** to the source minidisks that `RH62GOLD` owns and the destination minidisks that `LINUX153` owns. The figure shows that the **`LINK`** statement is issued as read-only (**`RR`**) for the source and read-write (**`W`**) for the target. The virtual disk-based swap spaces at addresses `300` and `301` are defined in-memory, therefore, they do not need to be copied.

9.4 Review system status

You can step back now and view your system from a DASD point of view as shown in Figure 9-2 on page 162. If you have followed all sections in this book you should have used the equivalent of 28 3390-3 volumes and 2 3390-9 volumes.

You can also view the system from an administrator's and end user point of view as shown by the horizontal lines and the italicized text on the right side of the figure. 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 end users may not care that their servers are virtual machines and may be oblivious to the fact that they might have been cloned in a matter of minutes.

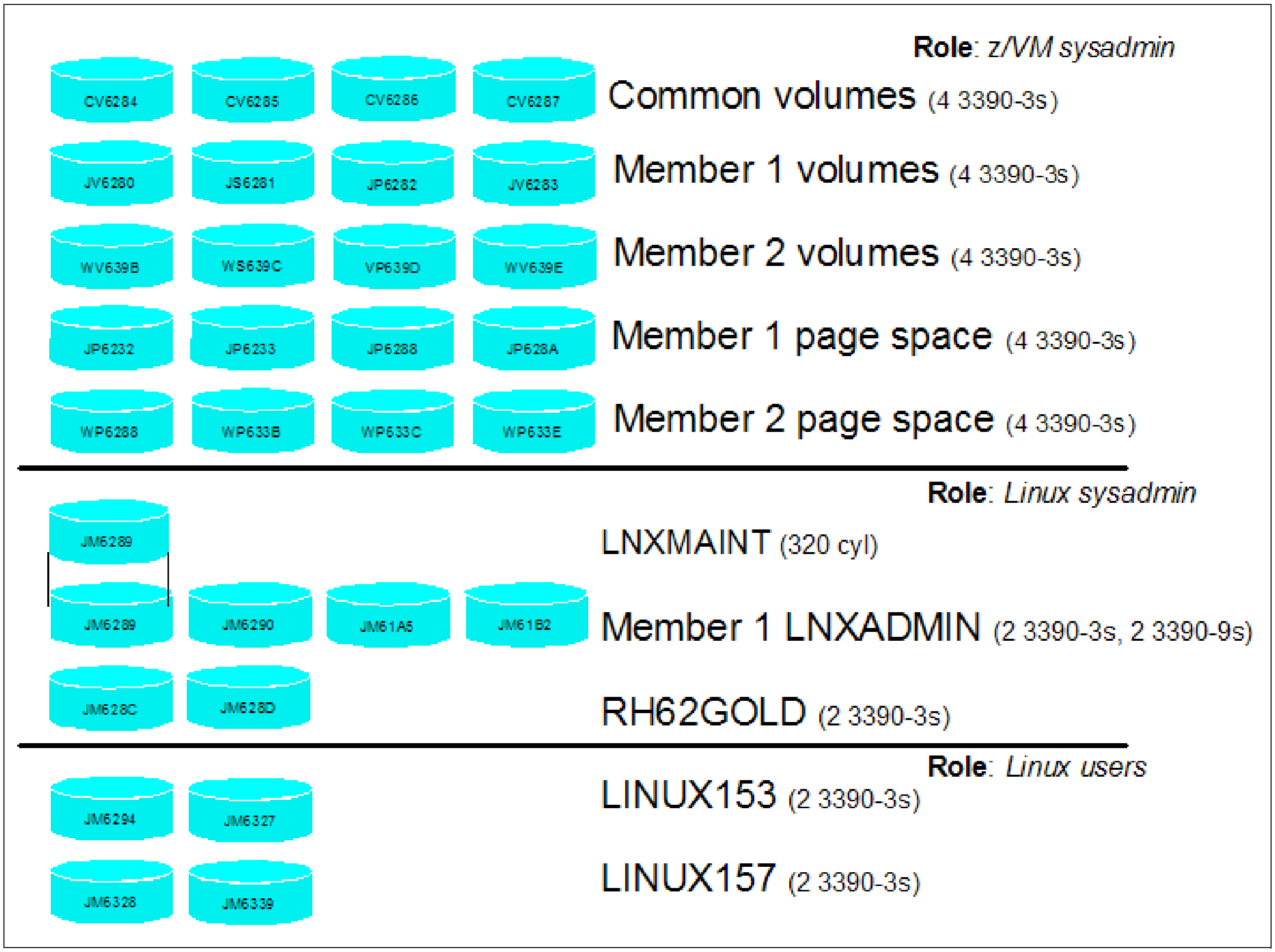


Figure 9-2 Linux virtual server system - DASD view and role view

10

Installing Linux with kickstart

“We still don’t know one thousandth of one percent of what nature has revealed to us.”

--Albert Einstein

Kickstart is an automated way of installing RHEL 6.2. Using kickstart, you can create a single file that answers all of the questions usually asked during an interactive installation.

In the previous chapter, you cloned to `LINUX153` and created three new virtual machines for virtual servers. In this chapter you will *kickstart* a RHEL 6.2 system to `LINUX153`. In comparison, cloning a server is faster, assuming the `FLASHCOPY` command is available. However, kickstart-ing a server is more flexible, as it allows for different package configurations as well as pre-install and post-install scripting.

The Linux administration system is now configured as an installation server using NFS to share the installation tree. Configure it as a kickstart server to perform automated installations over the network. The following steps are involved in installing Linux with kickstart:

- Configure the Linux administration system for kickstart
- “Configure LINUX153 for kickstart” on page 165
- “Kickstart the LINUX153 user” on page 166

10.1 Configure the Linux administration system for kickstart

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. This file on `LNXADMIN` will be used as a template for `LINUX153`.

Perform the following steps:

- Start an SSH session on the Linux administration system (`LNXADMIN`) as root.
- Start the golden image (`RH6GOLD`). You could log on to a 3270 session, but you can also start it from the Linux administration system with the CP **`XAUTOLOG`** command:

```
# vmcp xautolog rh62gold
Command accepted
```

- Create the directory `/nfs/ks/` for the kickstart file:

```
# cd /nfs
# mkdir ks
# cd ks
```

- Copy the sample kickstart file from the golden image:

```
# scp 9.60.18.144:/root/anaconda-ks.cfg linux153-ks.cfg
anaconda-ks.cfg                              100% 1813     1.8KB/s   00:00
```

- Set the read bits on the file with the **chmod +r** command:

```
# chmod +r linux153-ks.cfg
```

- Edit the kickstart configuration file as follows. After the first four changes which are in bold, remove the comments from the part, `volgroup` and `logvol` lines. Edit the lines in bold in to customize this kickstart for `LINUX153`:

```
# vi linux153-ks.cfg
# Kickstart file automatically generated by anaconda.

#version=RHEL6
install
reboot
nfs --server=9.60.18.240 --dir=/var/nfs/rhel62
lang en_US.UTF-8
rootpw  --iscrypted
$6$jiFGqyU1FwxWWQ6t$7qnsOSsUsNOyGnjtIpR63z2O4RDjL1q6M//lxfA.E5SbQ.M2gNKCJpahQ.mO7JCm.56y
H3vKbxc5bVtvRERwdO
firewall --disabled
authconfig --enableshadow --passalgo=sha512 --enablefingerprint
selinux --enforcing
timezone --utc America/New_York
bootloader --location=mbr --driveorder=dasdb,dasdc,dasdd,dasde
--append="crashkernel=auto"
# 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
clearpart --all --initlabel --drives=dasdb,dasdc,dasdd,dasde


part / --fstype=ext4 --size=512
part swap --size=512
part pv.Al9FUC-feWq-uHGF-Jaui-RxZQ-Kq9t-pi5zlC --grow --size=200
part pv.uB82Dq-ajP3-QEln-dcsJ-XHds-tCxx-BRjxOc --grow --size=200
part swap --grow --size=200
part swap --grow --size=200
volgroup system_vg --pesize=4096 pv.Al9FUC-feWq-uHGF-Jaui-RxZQ-Kq9t-pi5zlC
pv.uB82Dq-ajP3-QEln-dcsJ-XHds-tCxx-BRjxOc
logvol /opt --fstype=ext4 --name=opt_lv --vgname=system_vg --size=384
logvol /tmp --fstype=ext4 --name=tmp_lv --vgname=system_vg --size=384
logvol /usr --fstype=ext4 --name=usr_lv --vgname=system_vg --size=1536
logvol /var --fstype=ext4 --name=var_lv --vgname=system_vg --size=384
%packages
@base
...
%end
```

Following are clarifications to some of the values:

– The line `reboot` is added to set the server to automatically shutdown after kickstart.
– The line starting with `nfs --server=` sets the IP address of installation server and path to install tree.
– The line starting with `firewall` disables the firewall. This is not recommended if the server is on an external network.
– The line starting with `bootloader` removes references to additional drives only available to the Linux administration system.
– The line starting with `clearpart --all` specifies to remove all existing partitions.
– The line starting with `part /` defines the root partition to be 512 MB of type ext4.
– The line starting with `part swap` defines a swap partition of size 512 MB.
– The two lines starting with `part pv` specify to make physical volumes.
– The next two lines starting with `part swap` define partitions. Since they have the `--grow` parameter, all of the virtual disks will be used for swap, regardless of the size specified. Anaconda creates the swap devices based on the order in the kickstart file, so the first 512 MB swap space will be created on the first minidisk while the last two will be created on virtual disks `300` and `301`.
– The line starting with `volgroup` specifies to create a volume group.
– The next four lines starting with `logvol` defines the logical volumes created when installing.
– The line `@base` specifies a default set of packages for the install. These can be customized later by adding or removing specific packages from the `%packages` section.

- Add the path to the kickstart folder to `/etc/exports`:

```
# vi /etc/exports
/var/nfs/rhel62/           *(ro,sync)
/var/nfs/CKB-VM62          *(ro,sync)
/var/nfs/ks                *(ro,sync)
```

- Restart the NFS service on the Linux administration system. The **showmount -e** command should show the exported file systems:

```
# service nfs reload
# showmount -e
/var/nfs/ks       *
/var/nfs/CKB-VM62  *
/var/nfs/rhel62    *
```

You should now be ready to configure kickstart for a virtual machine.

10.2 Configure LINUX153 for kickstart

Earlier you should have created the virtual machine `LINUX153`. It is now time to configure it for kickstart. `LINUX153` must have its own parameter and configuration files, which are again based on the `RH6GOLD` virtual machine. Perform the following steps:

- **LOGOFF** of `MAINT` and logon to `LNXMAINT`. Copy the parameter and configuration files from `RH6GOLD` to `LINUX153` as follows:

```
==> copy rh62gold * d linux153 = =
```

- Edit the `LINUX153 PARM-RH6` file. Because this is a non-interactive installation, the vnc options are no longer required. The **ks=** line directs the installer to get the kickstart file from the installation server. **RUNKS=1** is required for “kick starts”, and the **cmdline** option prevents the installer’s text-based user interface from opening on the 3270 console:

```
==> x linux153 parm-rh6 d
ramdisk_size=40000 root=/dev/ram0 ro ip=off
CMSDASD=191 CMSCONFFILE=linux153.conf-rh6
ks=nfs:9.60.18.151:/var/nfs/ks/linux153-ks.cfg
RUNKS=1 cmdline
====> file
```

- Next, edit the LINUX153 CONF file, and change the DASD range and networking information:

```
==> x linux153 conf-rh6 d
DASD=100-101,300-301
HOSTNAME=gpok153.endicott.ibm.com
NETTYPE=qeth
IPADDR=9.60.18.153
...
====> file
```

- Logoff of LNXMAINT.

You should now be ready to *kick start* RHEL 6.2 onto a virtual machine.

10.3 Kickstart the LINUX153 user

Perform the following steps to kickstart the LINUX153 user:

- Logon to LINUX153. When asked to IPL from disk 100, answer n:

```
LOGON LINUX153
...
Do you want to IPL Linux from minidisk 100? y/n
n
```

- Add more memory for the install process. Temporarily modify the storage up to 1 GB with the **DEFINE STORAGE** command. Then **IPL CMS** and again answer **n** to the question of IPLing Linux:

```
==> def stor 1g
00: STORAGE = 1G
00: Storage cleared - system reset.
==> ipl cms
...
Do you want to IPL Linux from minidisk 100? y/n
n
```

  Verify that you have a 1 GB virtual machine:

```
==> q v stor
00: STORAGE = 1G
```

  This change is for the duration of the virtual machine session. When you logoff and log back on this virtual machine, the storage will go back to 256MB.

- Run **RHEL62 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 screen. This is because the 3270 console cannot display the progress meter during the post installation phase. To automatically clear the 3270 console and avoid multiple screens of unreadable messages, issue the **#cp term more 0 0** command before running **RHEL62 EXEC**.

```
==> rhel62
...
Kernel command line: ramdisk_size=40000 root=/dev/ram0 ro ip=off
                     CMSDASD=191 CMSCONFFILE=linux153.conf-rh6
                     ks=nfs:9.60.18.151:/var/nfs/ks/linux153-ks.cfg
                     RUNKS=1 cmdline
...
```

- The first time 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...
```

- At the end of the kickstart, IPL the 100 disk to make any changes to your RHEL 6.2 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
...
```

Congratulations! You have now installed Linux onto the virtual server using kickstart. This process can be repeated in the future for other Linux guests. This chapter has shown a minimal installation with kickstart. However, 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 on **Installation Guide**, then **28. Kickstart Installations**.

11

Create RHEL 6.2 appliances

“The secret to creativity is knowing how to hide your sources.”

--Albert Einstein

Once you have the ability to clone generic Linux servers, it is relatively easy to customize them for specific tasks given the variety of open source tools included with Linux distributions. These customized virtual servers can be thought of as *appliances*.

This chapter describes four types of appliances:

- “Create a Web server appliance”
- “Create an application development appliance” on page 173
- “Create an LDAP appliance” on page 175
- “Create a file and print serving appliance” on page 181

The sections that follow don’t go into the theory nor detail on the four types of servers. Rather, they are just a reference to get the servers quickly installed and configured. There are many other resources that go into depth on these types of servers.

11.1 Create a Web server appliance

This section describes how to install and configure a virtual Web server using the following tasks:

- “Install Apache RPMs”
- “Test Apache” on page 170
- “Turn on a firewall” on page 171
- “Configure SSL” on page 172
- “Populate the Web site” on page 173

11.1.1 Install Apache RPMs

To accomplish this task, perform the following steps:

- Start an SSH session as root to the first target server that you cloned. In this example, it is ***gpok153*** running on `LINUX153`.
- Verify that the Apache RPMs are not installed:

```
# rpm -qa | grep httpd
```

  No output shows that no RPMs with the string `httpd` are installed.
- **SSH** into the IP address of the new `LINUX153` server. Install the following Apache RPMs with the **`yum -y install`** command. The **`-y`** flag prevents the “Is this OK” question:

```
# yum -y install httpd httpd-manual
...
Installed:
  httpd.s390x 0:2.2.15-13.el6        httpd-manual.noarch 0:2.2.15-13.el6

Dependency Installed:
  apr.s390x 0:1.3.9-3.el6_1.2              apr-util.s390x 0:1.3.9-3.el6_0.1
  apr-util-ldap.s390x 0:1.3.9-3.el6_0.1    httpd-tools.s390x 0:2.2.15-13.el6

Complete!
```

- Verify that the RPMs were installed:

```
# rpm -qa | grep httpd
httpd-2.2.15-13.el6.s390x
httpd-tools-2.2.15-13.el6.s390x
httpd-manual-2.2.15-13.el6.noarch
```

- 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
```

- Start the Apache Web server:

```
# service httpd start
Starting httpd: [  OK  ]
```

The Apache RPMs should now be installed with the service running.

11.1.2 Test Apache

To verify that Apache is installed correctly, after it’s been started, **bring up a Web browser** and point it to the server. For example, the virtual server running on `LINUX153` can be reached with the following URL:

`http://`***`9.60.18.153`*****`/`**

You should see the following test page to verify the Web server is working:

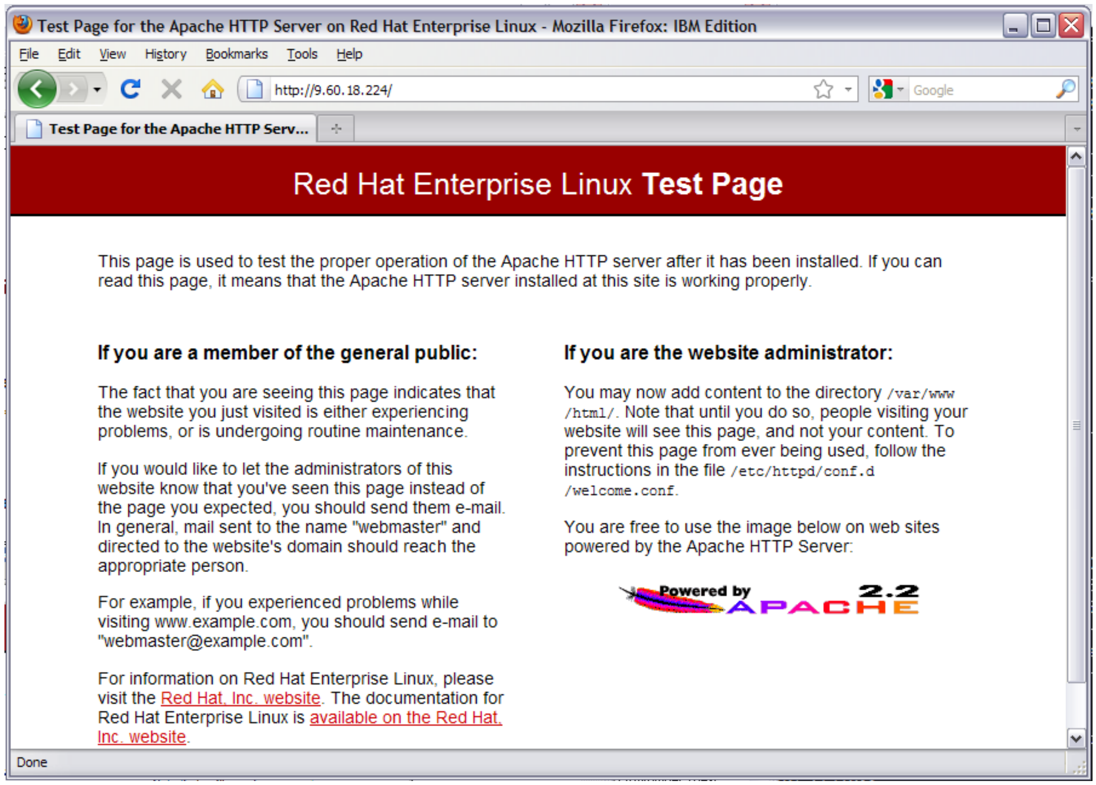


Figure 11-1 Apache test page

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 can't reach the test page from a browser, try accessing it using the IP address rather than the DNS name.

11.1.3 Turn on a firewall

RHEL 6.2 comes with an IP tables firewall. In section 8.2.3, "Turn off unneeded services" on page 146, it was recommended that you turn off the `iptables` service. If you did this on the golden image, the firewall is turned off on this clone. This section describes how to quickly enable an IP tables firewall and configure it to allow Web traffic through. Perform the following steps:

- Verify that the firewall is off with 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 (or **Unable to connect**). This is because packets for ports for http: and https: (80 and 443) are dropped by default.

- To allow Web traffic through, modify the file `/etc/sysconfig/iptables`. First make a copy of the original

```
# cd /etc/sysconfig
# cp iptables iptables.orig
```

- Add two rules (in bold) to allow these ports then save your changes:

```
# vi iptables
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

- Restart the firewall to pick up the new rules:

```
# service iptables restart
iptables: Flushing firewall rules: [  OK  ]
iptables: Setting chains to policy ACCEPT: filter [  OK  ]
iptables: Unloading modules: [  OK  ]
iptables: Applying firewall rules: [  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 to pass through.

11.1.4 Configure SSL

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. Perform the following steps:

- To use SSL, the `mod_ssl` package is required. Show that SSL communications *do not work* by changing `http` to **https** in the URL in your browser:

  https://*9.60.18.153*/

  You should see some type of communications error.

- Install the `mod_ssl` RPM with the **`yum -y install`** command:

```
# yum -y install mod_ssl
...
Installed:
  mod_ssl.s390x 1:2.2.15-13.el6

Complete!
```

- Verify that the RPM was added:

```
# rpm -qa | grep mod_ssl
mod_ssl-2.2.15-13.el6.s390x
```

- 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, which is acceptable for a test system. For a production Web site you will probably want to obtain a certificate signed by a certificate authority.

11.1.5 Populate the Web site

Web pages are stored in the directory `/var/www/html/` which is the default Web root. For example:

```
# cd /var/www/html/
# echo "<h2>Our new Web site</h2>" > index.html
```

Click refresh again and accept the certificate. You should see a Web page similar to that in Figure 11-2:

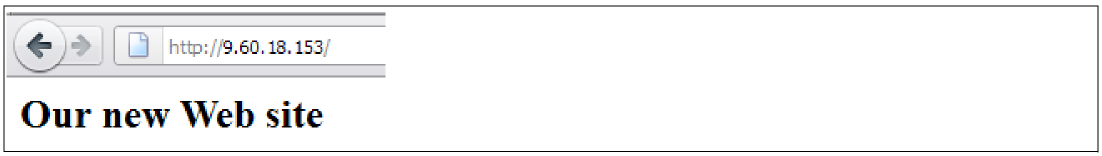


Figure 11-2 Testing the populating of your Web site

This section has shown how to install and configure Apache as a Web server.

The following Web sites contain additional information on 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
```

11.2 Create an application development appliance

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 are software available on Linux to help form a development system for developers to create integrated applications. 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.

You may choose to clone a new Linux virtual server, or continue to use the one you just installed Apache on to. In this example, the system running on `LINUX153`, which is now a Web server, is used.

- Start or an SSH session as `root` to the Linux system which will be used.
- Before installing the development tools, note how full the root and `/usr/` file systems are:

```
# df -h
Filesystem            Size  Used Avail Use% Mounted on
```

```
/dev/dasda1           504M  161M  319M  34% /
tmpfs                 121M     0  121M   0% /dev/shm
/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-usr_lv
                      2.0G  1.2G  723M  63% /usr
/dev/mapper/system_vg-var_lv
                      372M   76M  278M  22% /var
9.60.18.151:/var/nfs/rhel62
                       16G  5.8G  8.9G  40% /var/nfs/rhel62
```

In this example, they are 34% and 63% full

- Use the **`yum -y groupinstall`** command to install the groups named **development-tools** and **development-libs**. This will add about 45 packages which requires a number of minutes to install:

```
# yum -y groupinstall "Development tools" "Development libs"
...
Installed:
  autoconf.noarch 0:2.63-5.1.el6           automake.noarch 0:1.11.1-1.2.el6
  bison.s390x 0:2.4.1-5.el6                byacc.s390x 0:1.9.20070509-6.1.el6
  cscope.s390x 0:15.6-6.el6                ctags.s390x 0:5.8-2.el6
  diffstat.s390x 0:1.51-2.el6              doxygen.s390x 1:1.6.1-6.el6
  flex.s390x 0:2.5.35-8.el6                gcc.s390x 0:4.4.6-3.el6
  gcc-c++.s390x 0:4.4.6-3.el6              gcc-gfortran.s390x 0:4.4.6-3.el6
  git.s390x 0:1.7.1-2.el6_0.1              indent.s390x 0:2.2.10-5.1.el6
  intltool.noarch 0:0.41.0-1.1.el6         libtool.s390x 0:2.2.6-15.5.el6
  patchutils.s390x 0:0.3.1-3.1.el6         rcs.s390x 0:5.7-37.el6
  redhat-rpm-config.noarch 0:9.0.3-34.el6   rpm-build.s390x 0:4.8.0-17.el6
  subversion.s390x 0:1.6.11-2.el6_1.4      swig.s390x 0:1.3.40-6.el6
  systemtap.s390x 0:1.6-2.el6

Dependency Installed:
  cloog-ppl.s390x 0:0.15.7-1.2.el6      cpp.s390x 0:4.4.6-3.el6
  gettext-devel.s390x 0:0.17-16.el6     gettext-libs.s390x 0:0.17-16.el6
  glibc-devel.s390x 0:2.12-1.43.el6     glibc-headers.s390x 0:2.12-1.43.el6
  kernel-devel.s390x 0:2.6.32-202.el6   kernel-headers.s390x 0:2.6.32-202.el6
  libart_lgpl.s390x 0:2.3.20-5.1.el6    libgcj.s390x 0:4.4.6-3.el6
  libproxy.s390x 0:0.3.0-2.el6          libproxy-bin.s390x 0:0.3.0-2.el6
  libproxy-python.s390x 0:0.3.0-2.el6   libstdc++-devel.s390x 0:4.4.6-3.el6
  mpfr.s390x 0:2.4.1-6.el6              neon.s390x 0:0.29.3-1.2.el6
  pakchois.s390x 0:0.4-3.2.el6          perl-Error.noarch 1:0.17015-4.el6
  perl-Git.noarch 0:1.7.1-2.el6_0.1     ppl.s390x 0:0.10.2-11.el6

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 your file systems. In this example, the root file system was not changed, but `/usr/` is now 76% full:

```
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1           504M  161M  318M  34% /
tmpfs                 121M     0  121M   0% /dev/shm
/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-usr_lv
                      2.0G  1.5G  476M  76% /usr
/dev/mapper/system_vg-var_lv
                      372M   84M  269M  24% /var
9.60.18.151:/var/nfs/rhel62
                       16G  5.8G  8.9G  40% /var/nfs/rhel62
```

The Linux system should now have many application development tools installed and ready to use.

11.2.1 Additional resources

The following Web sites are resources for additional information on application development topics:

Scripting languages

http://www.perl.com/
http://www.python.org/
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.htmll
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/

11.3 Create an LDAP appliance

The Lightweight Directory Access Protocol (LDAP) is commonly implemented with 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 steps in this section are as follow:

- "Install the OpenLDAP server" on page 176
- "Configure the OpenLDAP server" on page 176

- "Start the LDAP service" on page 177
- "Add an LDAP user" on page 177
- "Configure an LDAP client" on page 179

11.3.1 Install the OpenLDAP server

In this example, a RHEL 6.2 server was installed on LINUX157 using kickstart. This will not have **yum** configured for online updates.

Perform the following steps to create an OpenLDAP server:

- Update the Linux system running on LINUX157 to configure **yum** as described in 8.2.2, "Configure yum for online updates" on page 146. You could also use the **clone** script to clone the golden image over the kickstart-ed Linux.
- Start an SSH session to the IP address of the new virtual server running on LINUX153. Use the **yum** command to install the OpenLDAP client and server RPMs:

```
# yum -y install openldap-clients openldap-servers
...
Installed:
  openldap-clients.s390x 0:2.4.23-19.el6 openldap-servers.s390x 0:2.4.23-19.el6

Dependency Installed:
  libtool-ltdl.s390x 0:2.2.6-15.5.el6

Complete!
```

OpenLDAP should now be installed on LINUX157.

11.3.2 Configure the OpenLDAP server

Any detailed description of LDAP is outside the scope of this book. Rather, short configuration recommendations are given in this section.

There are two important configuration values that must be chosen.

1. 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.
2. The LDAP administrator or root name and password.

Perform the following steps:

- Choose an administrative password and run the **slappasswd** command which displays an encrypted version of it. The output of this command will be used shortly in a configuration file so you may want to make a copy of it.

```
# slappasswd
New password: lnx4vm
Re-enter new password: lnx4vm
{SSHA}th6d+bfXgap5N+Pzf97AewivW4FWEXsO
```

- The OpenLDAP server configuration file that will contain the LDAP manager (root) password is /etc/openldap/slapd.d/cn=config/olcDatabase={1}bdb.ldif. Make a backup copy of that file:

```
# cd /etc/openldap/slapd.d/cn=config
# cp olcDatabase={2}bdb.ldif olcDatabase={2}bdb.ldif.orig
```

- Edit the file and add one line to set the LDAP manager's password. Use the variable olcRootPW and set the password to the output of the previous **slappasswd** command:

```
# vi olcDatabase={2}bdb.ldif
dn: olcDatabase={2}bdb
objectClass: olcDatabaseConfig
objectClass: olcBdbConfig
olcDatabase: {2}bdb
olcSuffix: dc=my-domain,dc=com
olcAddContentAcl: FALSE
olcLastMod: TRUE
olcMaxDerefDepth: 15
olcReadOnly: FALSE
olcRootDN: cn=Manager,dc=my-domain,dc=com
olcRootPW: {SSHA}th6d+bfXgap5N+Pzf97AewivW4FWEXsO
olcSyncUseSubentry: FALSE
olcMonitoring: TRUE
olcDbDirectory: /var/lib/ldap
...
```

- Save the file.

Your LDAP server should now be minimally configured.

11.3.3 Start the LDAP service

To start the LDAP server, perform the following steps:

- Start LDAP at boot time with the **chkconfig** command and for this session with the **service** command:

```
# chkconfig slapd on
# service slapd start
Starting slapd: [  OK  ]
```

- Query the LDAP database with 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 as no data has been added to it.

11.3.4 Add an LDAP user

Define a new LDAP user. To do so, perform the following steps:

- Choose an LDAP user name. In this example, `mikemac` will be used. Verify there is no such local user with the **id** command:

```
# id mikemac
id: mikemac: No such user
```

- An LDIF (LDAP Interchange Format) file is created to add an organizational unit named **People** and a user named **mikemac**. Create a similar file for your system's values.

```
# cd /tmp
# vi initial.ldif // create the input file ...
dn: dc=my-domain,dc=com
objectClass: dcObject
objectClass: organization
description: my-domain domain
o: my-domain
dc: my-domain

dn: cn=Manager,dc=my-domain,dc=com
objectClass: organizationalRole
cn: Manager

dn: ou=People,dc=my-domain,dc=com
ou: People
objectClass: top
objectClass: organizationalUnit

dn: uid=mikemac,ou=People,dc=my-domain,dc=com
uid: mikemac
cn: mikemac
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
loginShell: /bin/bash
uidNumber: 10000
gidNumber: 10000
homeDirectory: /home/mikemac

dn: ou=Group,dc=my-domain,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Group

dn: cn=mikemac,ou=Group,dc=my-domain,dc=com
objectClass: posixGroup
objectClass: top
cn: mikemac
userPassword: {crypt}x
gidNumber: 10000
```

- Add the contents of the LDIF file to the LDAP server with the `ldapadd` command:

```
# ldapadd -x -h localhost -D "cn=Manager,dc=my-domain,dc=com" -f initial.ldif -W
Enter LDAP Password:
adding new entry "dc=my-domain,dc=com"

adding new entry "cn=Manager,dc=my-domain,dc=com"

adding new entry "ou=People,dc=my-domain,dc=com"

adding new entry "uid=mikemac,ou=People,dc=my-domain,dc=com"

adding new entry "ou=Group,dc=my-domain,dc=com"

adding new entry "cn=mikemac,ou=Group,dc=my-domain,dc=com"
```

- Set the base distinguished name to **dc=my-domain,dc=com.** This is set in the BASE variable in the LDAP client configuration file /etc/openldap/ldap.conf:

```
# cd /etc/openldap
# cp ldap.conf ldap.conf.orig
# vi ldap.conf
#
# LDAP Defaults
#

# See ldap.conf(5) for details
# This file should be world readable but not world writable.

BASE dc=my-domain,dc=com
...
```

- Search for the new user just added with the **ldapsearch** command:

```
# ldapsearch -x uid=mikemac
...
# mikemac, People, my-domain.com
dn: uid=mikemac,ou=People,dc=my-domain,dc=com
uid: mikemac
cn: mikemac
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
loginShell: /bin/bash
uidNumber: 10000
gidNumber: 10000
homeDirectory: /home/mikemac

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

- This shows that the user exists in the LDAP database. Now you may want to set the password with the **ldappasswd** command. You will need to provide a new password for the new user and you will also need to provide the LDAP administrator password.

```
# ldappasswd -x -D "cn=Manager,dc=my-domain,dc=com" -W -S
"uid=mikemac,ou=People,dc=my-domain,dc=com"
New password:
Re-enter new password:
Enter LDAP Password:
```

You have now deleted a local user, added a new LDAP user using an LDIF file, and have set the new LDAP user's password.

You should now have an OpenLDAP server installed, configured and populated with users and groups.

11.3.5 Configure 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. In this example the Linux system running on the

LINUX153 virtual machine is used. To configure it to point to this LDAP server, perform the following steps:

- Start an SSH session as `root` to the Linux running on `LINUX153`.
- Install the `nss-pam-ldap` RPM with the **yum -y** command:

```
# yum -y install nss-pam-ldapd
...
Installed:
  nss-pam-ldapd.s390x 0:0.7.5-10.el6

Dependency Installed:
  nscd.s390x 0:2.12-1.43.el6            pam_ldap.s390x 0:185-11.el6

Complete!
```

- 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. Set the **Use LDAP** under *User Information*, and **Use LDAP Authentication** under *Authentication*. Click **Next**.

```
# 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 Fingerprint reader              ¦
      ¦                          [ ] Use Winbind Authentication          ¦
      ¦                          [*] Local authorization is sufficient   ¦
      ¦                                                                  ¦
      ¦            ----------                       --------             ¦
      ¦            ¦ Cancel ¦                       ¦ Next ¦             ¦
      ¦            ----------                       --------             ¦
      ¦                                                                  ¦
      ¦                                                                  ¦
      ¦                                                                  ¦
      -------------------------------------------------------------------¦
```

- On the next screen, set the *Server* value to point to the LDAP server. In this example, it is **ldap://*9.60.18.157*/**. Set the `Base DN` to your suffix value. In this example it is **dc=my-domain,dc=com**. "Press" **OK**.

```
             ------------------- LDAP Settings -------------------¦
             ¦                                                    ¦
             ¦          [ ] Use TLS                               ¦
             ¦  Server: ldap://9.60.18.157______________________  ¦
             ¦ Base DN: dc=my-domain,dc=com_____________________  ¦
             ¦                                                    ¦
             ¦         --------                ------             ¦
             ¦         ¦ Back ¦                ¦ Ok ¦             ¦
             ¦         --------                ------             ¦
             ¦                                                    ¦
             ¦                                                    ¦
             -----------------------------------------------------¦
```

- The LDAP client should now be pointing to the LDAP server. Test it with the **id *mikemac*** command:

```
# id mikemac
uid=10000(mikemac) gid=10000(mikemac) groups=10000(mikemac)
```

In RHEL 6.2, you can no longer authenticate over SSH without using TLS. This section has not described how to set up TLS. To do that, you would need a signed certificate that corresponds to your enterprise's DNS domain name. There is some information at the OpenLDAP Web site.

http://www.openldap.org/pub/ksoper/OpenLDAP_TLS.html

More details on the `cn=config/` directory replacing the `/etc/openldap/slapd.conf` file is on the following Red Hat Web site (you need a subscription to get to it):

https://access.redhat.com/kb/docs/DOC-3637

11.4 Create a file and print serving appliance

Samba allows Windows clients to map Linux file systems as shared drives. Samba can also act as a middle tier between Windows clients and a Linux print server. The recommended Linux print server is CUPS - the Common UNIX Printing System. This section does not describe the configuration of CUPS but it does describe how the necessary RPMs are installed.

The steps in this section are as follow:

- "Install necessary RPMs" on page 181
- "Configure Samba configuration file" on page 181
- "Adding a Samba user" on page 182
- "Start Samba at boot time" on page 183
- "Test your changes" on page 183

11.4.1 Install necessary RPMs

To install the Samba RPMs, perform the following steps:

- Start an SSH session to the Linux system onto which you will install Samba. In this example, it is `LINUX157`.
- Add the following RPM with the **`yum -y`** command:

```
# yum -y install samba
...
Installed:
  samba.s390x 0:3.5.10-104.el6

Complete!
```

Confirm that the RPMs were added:

```
# rpm -qa | grep samba
samba-winbind-clients-3.5.10-104.el6.s390x
samba-common-3.5.10-104.el6.s390x
samba-3.5.10-104.el6.s390x
samba-client-3.5.10-104.el6.s390x
```

11.4.2 Configure 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/` as it has excellent Linux documentation. The following example will create a file *share* named `sharedoc`:

To do so, perform the following steps:

Change directory to /etc/samba/ and backup the configuration file:

```
# cd /etc/samba
# cp smb.conf smb.conf.orig
```

- Add three lines at the bottom of the file defining the *share* named sharedoc:

```
# vi smb.conf    // add three lines at the bottom of the file:
...
[sharedoc]
  comment = RHEL 6.2 on System z documentation
  path = /usr/share/doc/
```

- Verify the syntax of your changes with the **testparm** command. You should see a reference to the new sharedoc section that was just added.

```
# 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.

11.4.3 Adding a Samba user

The default method that Samba uses to determines users' credentials is to look in the /etc/samba/smbpasswd file. That user must first exist in the Linux file system (/etc/passwd, /etc/shadow, etc).

This method of maintaining Samba users, groups and passwords is good 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 "Create an LDAP appliance" on page 175 because the Samba schema must first be added to LDAP. Details are outside the scope of this book.

To add a Samba user, perform the following steps:

- 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
BAD PASSWORD: it is based on a dictionary word
BAD PASSWORD: is too simple
Retype new password: lnx4vm
passwd: all authentication tokens updated successfully.
```

- Add the user sambauser1 to the smbpasswd file with 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.
```

The local and Samba user `sambauser1` should now be added to the system.

11.4.4 Start Samba at boot time

Samba can be started for the current session with the **service** command and at boot time with the **chkconfig** command. Do this for both the **smb** and **nmb** services:

```
# service smb start
Starting SMB services: [  OK  ]
# service nmb start
Starting NMB services: [  OK  ]
# chkconfig smb on
# chkconfig nmb on
```

Samba should now be running and configured to start at boot time.

11.4.5 Test your changes

Verify the configuration settings with the following steps:

- Verify that Samba, composed of the **smb** and **nmb** services, is running with the following **service** command:

```
# service smb status
smbd (pid  2073) is running...
# service nmb status
nmbd (pid  1817) is running...
```

- Verify the shares that are available with the following **smbclient** command:

```
# smbclient -U sambauser1 -L localhost
Enter sambauser1's password:
Domain=[MYGROUP] OS=[Unix] Server=[Samba 3.5.6-86.el6]

        Sharename       Type      Comment
        ---------       ----      -------
        sharedoc        Disk      RHEL 6.2 on System z documentation
        IPC$            IPC       IPC Service (Samba Server Version 3.5.6-86.el6)
        sambauser1      Disk      Home Directories
...
```

To test getting a Samba share from a Windows desktop, perform the following steps:

- 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 11-3 on page 184. In this example the UNC is **\\9.60.18.157\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.
- 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 11-3 on page 184.

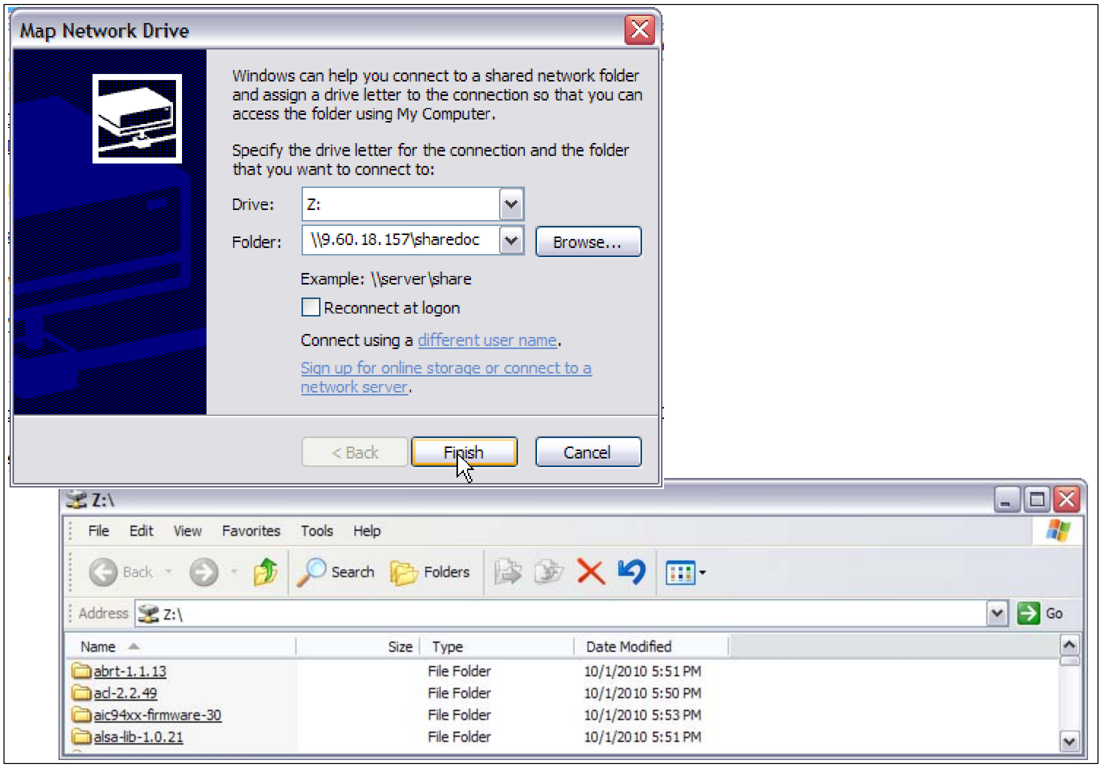


Figure 11-3 Mapping a network drive to the Samba server

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 with the following **net use** command:

```
c:\>net use y: \\9.60.18.157\sharedoc

Enter the user name for '9.60.18.157': sambauser1
Enter the password for 9.60.18.157:
The command completed successfully.
```

Detach the share with the following **net use** command:

```
c:\>net use y: /delete
y: was deleted successfully.
```

11.4.6 Configure printing

Configuring printing is beyond the scope of this section. For details see the Redpaper *Printing with Linux on zSeries Using CUPS and Samba*, REDP-3864, on the Web at:

http://www.redbooks.ibm.com/abstracts/redp3864.html

Congratulations - you should now have two virtual servers set up with Apache and application development tools on the first and LDAP and Samba on the second.

12

Service Linux with the Red Hat Network

"*The faster you go, the shorter you are.*"

— Albert Einstein

This chapter describes Red Hat Network (RHN) and its ability to manage the virtual servers. Using **yum**, 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:

- "Register your system with RHN" on page 185
- "Install and updating packages using yum" on page 186
- "Manage your systems through the RHN" on page 187

12.1 Register your system with RHN

This section assumes you have already obtained a valid entitlement for RHEL 6.2 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/z

Select the link **Free Evaluation** under the section *Try* on the left and create an account if you don't already have one. After filling out the form, you will receive an e-mail soon with activation instructions.

12.2 Install and updating packages using yum

You may choose to perform these steps first on a “clone”, such as LINUX153, then later on the golden image. In this fashion, you can test the process on an appliance that can be discarded, and later when all is tested and working, update the golden image so that all clones created thereafter are enabled for RHN.

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 /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
# rhnreg_ks --username=myuser --password=mypw --profilename=linux153.endicott.ibm.com
```

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.

12.3 Manage your systems through the 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, there are 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.

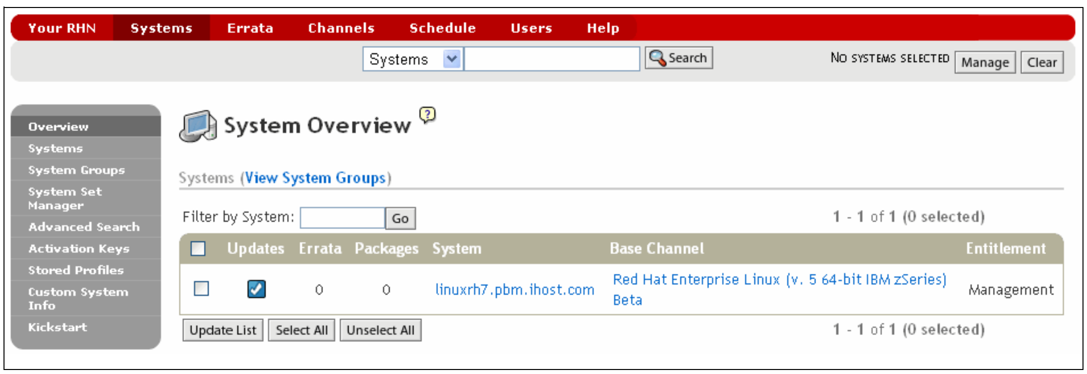


Figure 12-1 RHN system overview

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 12-2 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.

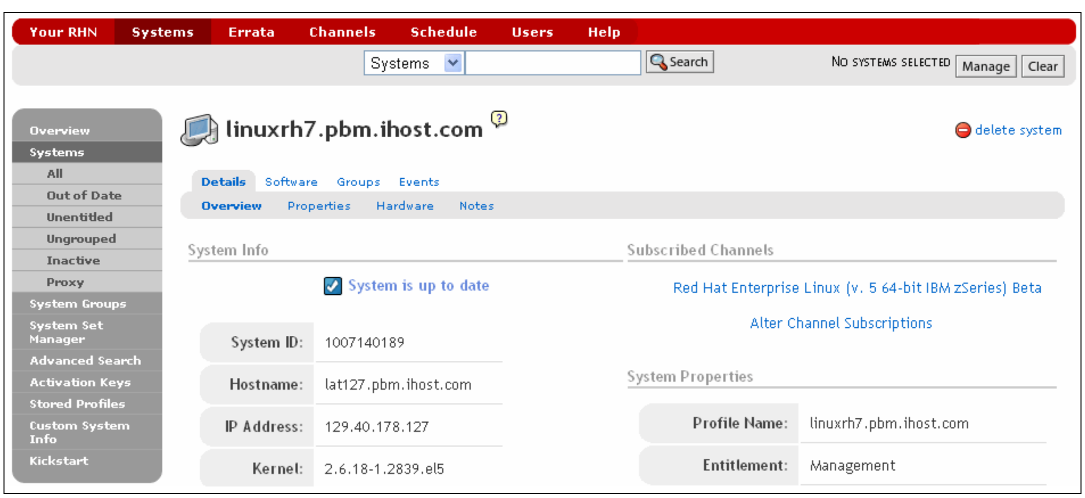


Figure 12-2 RHN system details

For more information about managing your systems through RHN, including usage guides and frequently asked questions, see:

http://rhn.redhat.com/help

Part 3

SLES 11 SP2 Linux

This part of the book focuses on SuSE Linux Enterprise Server (SLES). It consists of the following chapters:

- Chapter 13., "Install SLES 11 SP2 on LNXADMIN" on page 191 - describes how to install and configure SLES 11 SP2 onto the *Linux administration system*, which does the cloning and other tasks.
- Chapter 14., "Install the SLES 11 SP2 golden image" on page 211 - describes how to install and configure two Linux images onto the *golden image*, which is cloned from.
- Chapter 15., "Clone SLES 11 SP2" on page 237 - explains how to prepare z/VM virtual machines and clone your first virtual server, both manually and by using a shell script.

13

Install SLES 11 SP2 on LNXADMIN

"*The only thing that interferes with my learning is my education.*"

— Albert Einstein

RHEL or SLES?: If you are working only with RHEL 6.2, you can skip this chapter and follow Chapter 7, "Install RHEL 6.2 on LNXADMIN" on page 107 to install on member 2 as you did for member 1.

If you are working only with SLES 11 SP2, use this chapter to install on `LNXADMIN` on both members. However on member 1, modify the install process so there is a large logical volume mounted over `/var/`. Adding the logical volume with SLES 11 SP2 is not described in this chapter, but it is described in section 7.4 *Installing the cloner* in the book *z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES 11 SP1*, on the Web at:

`http://www.redbooks.ibm.com/abstracts/sg247931.html?Open`

In this chapter you will install SLES 11 SP2 onto the `IDENTITY LNXADMIN` on SSI member 2. This `IDENTITY` should have been defined in the z/VM user directory in section 7.1.1, "Create the identity LNXADMIN" on page 108.

To achieve this, perform the following overall steps:

- "Review the IDENTITY LNXADMIN" on page 191
- "Prepare the SLES 11 SP2 bootstrap files" on page 192
- "Install SLES 11 SP2 onto the Linux administration system" on page 195
- "Configure the SLES 11 SP2 Linux administration system" on page 203

13.1 Review the IDENTITY LNXADMIN

In this section you will review the identity that should have been defined in section 7.1.1, "Create the identity LNXADMIN" on page 108.

To accomplish this, perform the following steps:

- Logon to MAINT.
- Edit the USER DIRECT file:

```
==> x user direct c
```

- Search for the string LNXADMIN and review the SUBCONFIG LNXADM-2:

```
====> /lnxadmin
IDENTITY LNXADMIN LNX4VM 256M 1G BDEG
 INCLUDE LNXDFLT
 BUILD ON POKDEV62 USING SUBCONFIG LNXADM-1
 BUILD ON POKTST62 USING SUBCONFIG LNXADM-2
 OPTION LNKNOPAS
SUBCONFIG LNXADM-1
 MDISK 0100 3390 0001  3338 JM6290 MR LNX4VM LNX4VM LNX4VM
 MDISK 0101 3390 0521  2818 JM6289 MR LNX4VM LNX4VM LNX4VM
 MDISK 0102 3390 0001 10016 JM61A5 MR LNX4VM LNX4VM LNX4VM
 MDISK 0103 3390 0001 10016 JM61B2 MR LNX4VM LNX4VM LNX4VM
SUBCONFIG LNXADM-2
 MDISK 0100 3390 0001  3388 JM6293 MR LNX4VM LNX4VM LNX4VM
```

This will be the IDENTITY onto which SLES 11 SP2 is installed. The SUBCONFIG LNXADM-2 has only a single minidisk at virtual device 100.

13.2 Prepare the SLES 11 SP2 bootstrap files

To IPL a SLES 11 SP2 installation system, the following three bootstrap files must be copied and “punched” to the reader:

- A Linux kernel
- A parameter file
- A Linux initial RAMdisk

Think of these files as a PC Linux boot CD or DVD. A short REXX EXEC is commonly used to clean out the reader, punch the three files and IPL the reader.

To prepare the SLES 11 SP2 bootstrap files, perform the following steps:

- Start an SSH session as **root** on the PC NFS server.
- Change directory to the mounted DVD. You should see a directory boot/ where the kernel and RAMdisk are located:

```
# cd /var/nfs/sles11sp2/dvd1
# ls -F
ARCHIVES.gz   content                            gpg-pubkey-9c800aca-4be01999.asc
COPYING       content.asc                        gpg-pubkey-a1912208-446a0899.asc
COPYING.de    content.key                        gpg-pubkey-b37b98a9-4be01a1a.asc
COPYRIGHT     control.xml                        license.tar.gz
COPYRIGHT.de  directory.yast                     ls-lR.gz
ChangeLog     docu/                              media.1/
INDEX.gz      gpg-pubkey-0dfb3188-41ed929b.asc   pubring.gpg
NEWS          gpg-pubkey-1d061a62-4bd70bfa.asc   suse/
README        gpg-pubkey-307e3d54-4be01a65.asc   suse.ins
README.BETA   gpg-pubkey-3d25d3d9-36e12d04.asc
boot/         gpg-pubkey-7e2e3b05-4be037ca.asc
```

- Change directory to boot/s390x/ and invoke the **ls** command. You should see the initial RAMdisk and kernel named **initrd** and **vmrdr.ikr**:

```
# cd boot/s390x
```

```
# ls -l initrd vmrdr.ikr
-r--r--r-- 1 root root 15178300 Nov 29 03:47 initrd
-r--r--r-- 1 root root  7970048 Nov 29 03:47 vmrdr.ikr
```

- FTP to SSI member 2 (member 1 would also work as there is only one `LNXMAINT 192` disk) and login as `LNXMAINT`. Copy the SLES 11 SP2 kernel (the file `vmrdr.ikr` copied as `SLES11S1 KERNEL`) and the initial RAMdisk (the file `initrd` copied as `SLES11S1 INITRD`). These files must have a format of fixed 80-byte records and be transferred in binary. This format can be set with the **bin** and **site fix 80** FTP subcommands (if this subcommand fails, try **quote site fix 80**):

```
# ftp 9.60.18.249
Connected to 9.60.18.249.
220-FTPSERVE IBM VM Level 610 at GPOK249.ENDICOTT.IBM.COM, 08:59:44 EST THURSDAY
2009-11-26
220 Connection will close if idle for more than 5 minutes.
Name (9.60.18.249:root): lnxmaint
331 Send password please.
Password: lnx4vm
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> bin
200 Representation type is IMAGE.
ftp> site fix 80
200 Site command was accepted.
ftp> put vmrdr.ikr SLES11S2.KERNEL
local: vmrdr.ikr remote: SLES11S2.KERNEL
...
ftp> put initrd SLES11S2.INITRD
local: initrd remote: SLES11S2.INITRD
...
ftp> quit
```

- Logoff `MAINT` if you are still logged on.
- Logon to `LNXMAINT`.
- Besides the kernel and RAMdisk that you just copied, the file `SLES11S2 EXEC` should exist on the `LNXMAINT 192` disk. Use the **FILELIST** command to verify that the kernel and RAMdisk were copied in fixed-80 byte record format. You should see the following files:

```
==> filel sles11s2 * d
LNXMAINT FILELIST A0  V 169  Trunc=169 Size=3 Line=1 Col=1 Alt=0
Cmd   Filename Filetype Fm Format Lrecl    Records     Blocks   Date     Time
      SLES11S2 INITRD   D1 F         80     189938       3710 12/17/11  6:01:46
      SLES11S2 KERNEL   D1 F         80      99629       1659 12/17/11  6:01:37
      SLES11S2 EXEC     D1 V         72         10          1 11/16/11 11:16:17
```

- Quit by pressing **F3**.
- Verify that the file **SLES11S2 EXEC** has the correct information. Note the kernel and RAMdisk have hard coded file names, but the file name of the parameter file will be the user ID (**userid()** function) of the user running the EXEC:

```
==> type sles11s2 exec d

/* EXEC to punch SLES-11 SP2 install system to reader and IPL from it */
Address 'COMMAND'
'CP SPOOL PUN *'
'CP CLOSE RDR'
'CP PURGE RDR ALL'
'PUNCH SLES11S2   KERNEL * (NOHEADER'
```

```
'PUNCH' Userid() 'PARM-S11 * (NOHEADER'
'PUNCH SLES11S2   INITRD * (NOHEADER'
'CP CHANGE RDR ALL KEEP'
'CP IPL 00C CLEAR'
```

- A sample parameter file named `SAMPLE PARM-S11` is provided to save typing. View it with the **TYPE** command:

```
==> type sample parm-s11 d
ramdisk_size=65536 root=/dev/ram1 ro init=/linuxrc TERM=dumb
HostIP=n.n.n.n Hostname=yourhost.example..com
Gateway=n.n.n.n Netmask=255.255.255.0
Broadcast=n.n.n.n Layer2=1 OSAHWaddr=02:00:0B:FF:FF:FF
ReadChannel=0.0.0700  WriteChannel=0.0.0701  DataChannel=0.0.0702
Nameserver=n.n.n.n
portname=whatever
portno=0
Install=nfs://n.n.n.n/nfs/sles11sp2/SLES-11-SP2-DVD-s390x-GM-DVD1.iso
UseSSH=1  SSHPassword=12345678 UseVNC=1  VNCPassword=12345678
InstNetDev=osa OsaInterface=qdio OsaMedium=eth Manual=0
```

- Copy the sample parameter file to a new file with the file name `LNXADMIN`. Edit the file and update the networking variables with the values correct for your site. Refer to the worksheet in section 2.8.4, “Linux resources worksheet” on page 22. The fields you should change are in ***bold-italics***. The examples used in this book are as follows:

Important: If you decided to work with only layer 2 VSWITCHes instead of one layer two and one layer 3, make one additional change:

```
ReadChannel=0.0.0600  WriteChannel=0.0.0601  DataChannel=0.0.0602
```

This will specify to Linux to use the first NIC at virtual device addresses `0600-0602`, which will attach it to VSWITCH `vsw1`.

```
==> copy sample parm-s11 d lnxadmin = =
==> x lnxadmin parm-s11 d
ramdisk_size=65536 root=/dev/ram1 ro init=/linuxrc TERM=dumb
HostIP=9.60.18.152 Hostname=gpok152.endicott.ibm.com
Gateway=9.60.18.129 Netmask=255.255.255.128
Broadcast=9.60.18.255 Layer2=1
ReadChannel=0.0.0700  WriteChannel=0.0.0701  DataChannel=0.0.0702
Nameserver=9.0.2.11 portname=whatever portno=0
Install=nfs://9.60.18.240/nfs/sles11sp2/SLES-11-SP2-DVD-s390x-RC3-DVD1.iso
UseSSH=1  SSHPassword=12345678 UseVNC=1  VNCPassword=12345678
InstNetDev=osa OsaInterface=qdio OsaMedium=eth Manual=0
```

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

Note: SLES 11 SP2 documentation states VNC password should be six to eight character. However, actual installation process requires it to be at least eight characters long. If it is shorter, the installer will prompt for the password.

You are ready to start the installation.

13.3 Install SLES 11 SP2 onto the Linux administration system

In this section you will install the Linux administration system Linux image onto the new virtual machine LNXADMIN.

- Logon to LNXADMIN. The common PROFILE EXEC should run. You should see a virtual NICs being created at virtual addresses 600-602 and 700-702. You should also see two virtual disks created for swap spaces at virtual addresses 300 and 301:

```
00: z/VM Version 6 Release 2.0, Service Level 0000 (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 07:30:29 EDT MONDAY 09/05/11
00: Command complete
00: NIC 0600 is created; devices 0600-0602 defined
00: NIC 0600 is connected to VSWITCH SYSTEM VSW1
00: Command complete
00: NIC 0700 is created; devices 0700-0702 defined
00: NIC 0700 is connected to VSWITCH SYSTEM VSW2
z/VM V6.2.0    2011-11-15 11:26
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
```

- Answer no (**n**) to the question asking you to IPL Linux from 100:

```
Do you want to IPL Linux from minidisk 100? y/n
n
```

- Define the memory size (storage) to 1 GB with the **DEFINE STORAGE** command:

```
==> def stor 1g
00: STORAGE = 1G
00: Storage cleared - system reset.
```

- IPL CMS, press **Enter** at the VM READ prompt and again answer **n**o to the IPL question:

```
==> ipl cms
00: IPL CMSn
z/VM V6.2.0   2011-11-15 11:26

DMSACP723I A (191) R/O
DMSACP113S C(592) not attached or invalid device address
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
```

- Verify the increased memory size with the **QUERY VIRTUAL STORAGE** command:

```
==> q v stor
00: STORAGE = 1G
```

- Run the **SLES11S2 EXEC** to purge the reader, punch the bootstrap files and IPL from the reader. You should see the Linux RAMdisk getting loaded into memory. Look for the contents of the parameter file you created:

```
==> sles11s2
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 3.0.13-0.9-default (geeko@buildhost) (gcc version 4.3.4 Ýgcc-4_3-b
ranch revision 152973¨ (SUSE Linux) ) #1 SMP Mon Jan 16 17:33:03 UTC 2012 (54ddf
af)
setup.7055fd: Linux is running as a z/VM guest operating system in 64-bit mode
```

```
Zone PFN ranges:
  DMA      0x00000000 -> 0x00080000
  Normal   empty
Movable zone start PFN for each node
early_node_mapÝ1¨ active PFN ranges
    0: 0x00000000 -> 0x00010000
PERCPU: Embedded 10 pages/cpu @0000000001afb000 s11776 r8192 d20992 u40960
Built 1 zonelists in Zone order, mobility grouping on.  Total pages: 64640
Kernel command line: ramdisk_size=65536 root=/dev/ram1 ro init=/linuxrc TERM=dum
b                    HostIP=9.60.18.152 Hostname=gpok152.endicott.ibm.com
                     Gateway=9.60.18.129 Netmask=255.255.255.128
                     Broadcast=9.60.18.255 Layer2=1
                     ReadChannel=0.0.0700  WriteChannel=0.0.0701  DataChannel=0.
0.0702               Nameserver=9.0.2.11
                     portname=whatever
                     portno=0
                     Install=nfs://9.60.18.240/nfs/sles11sp2/SLES-11-SP2-DVD-s39
0x-RC3-DVD1.iso      UseVNC=1  VNCPassword=12345678
                     InstNetDev=osa OsaInterface=qdio OsaMedium=eth Manual=0
...
```

- You should be prompted for the MAC address. For z/VM to generate a MAC address, press **Enter** twice:

```
MAC address
>
(portname whatever)
(Layer2)
qeth.e9767c: register layer 2 discipline
(Port 0)
...
```

- You should then be prompted for a name server. Press **Enter** twice again if the default (***9.0.2.11*** in this example) is correct:

```
Enter the IP address of your name server. Leave empty or enter "+++" if you
don't need one
Ý9.0.2.11¨>
Loading Installation System (1/6) (23076 kB) -          0%      1%      2%
...
```

Important: If you see the following output on your 3270 screen, there is a problem.

```
Activating manual setup program.

>>> Linuxrc v3.3.34 (Kernel 2.6.27.19-5-default) <<<

Main Menu

1) Start Installation
2) Settings
3) Expert
4) Exit or Reboot
```

Often, this is because the install program cannot successfully mount the NFS file system, or it can mount it, but cannot find the SLES 11 SP2 install files.If you get these messages, you may want to immediately restart the install with the **#CP IPL 00C** command which IPLs from the reader. Watch the console messages carefully.

- A VNC server process will be started. You should see the messages:

```
starting VNC server...
```

```
A log file will be written to: /var/log/YaST2/vncserver.log ...

***
***          You can connect to 9.60.18.152, display :1 now with vncviewer
***          Or use a Java capable browser on  http://9.60.18.152:5801/
***

(When YaST2 is finished, close your VNC viewer and return to this window.)

*** Starting YaST2 ***
```

- Start a VNC session to the install process. In this example, a RealVNC client connects to would be ***`9.60.18.152`*****`:1`** as shown on the left side of Figure 13-1. Enter the password specified in the parameter file (**`12345678`** in this example) as shown on the right:

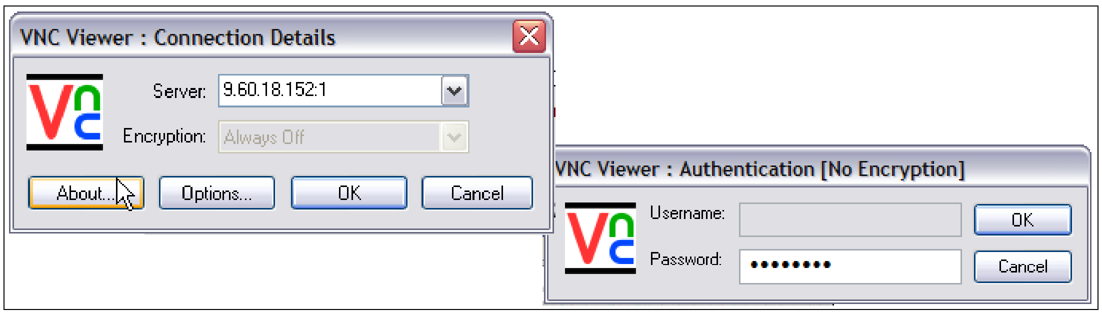


Figure 13-1 Using the VNC viewer

- You should see a window entitled *Welcome*. Select your Language and Keyboard. After reading Licence Agreement, click the check-box **I Agree to the License Terms** then click **Next**.
- At the *Disk Activation* window click the **Configure DASD Disks** button.
- At the *DASD Disk Management* window:
  a. Click **Select or Deselect** for the three read-write disks (**100, 300** and **301**).
  b. Click **Activate** on the *Perform Action* pop-up menu. The DASD will be activated quickly as shown on Figure 13-2:

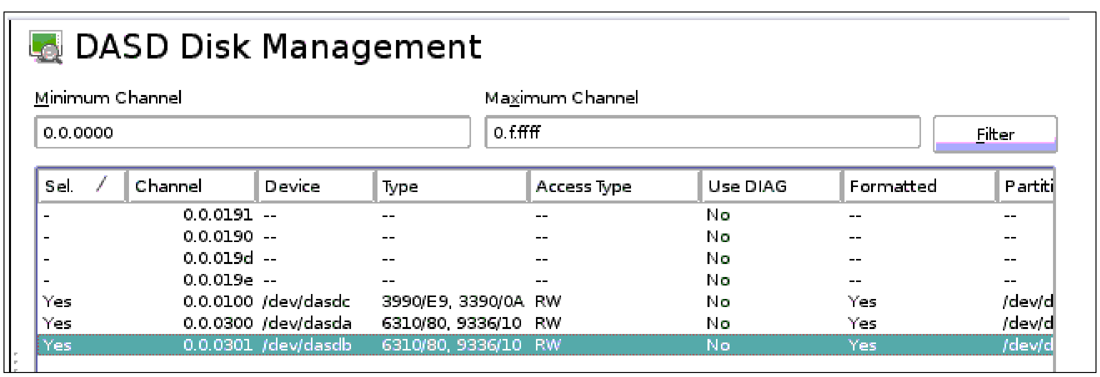


Figure 13-2 Activating DASD on the Linux administration system

  c. Click **Select or Deselect** to deselect minidisks **300** and **301** so that just minidisk `100` is selected.
  d. Click **Format** on the *Perform Action* pop-up menu.
  e. Click **OK** to the query to format the disk.

f. Click **Yes** to confirm. The minidisk will be formatted. This will take a few minutes.

g. Click **Next** when the formatting is complete.

h. In the *Disk Activation* window click **Next**.

- In the *Installation Mode* window, accept the default of **New installation** and Click **Next**.
- In the *Clock and Time Zone* window, select your time settings and click **Next**.
- In the *Installation Settings* window click **Partitioning**.
- In the *Preparing Hard Disk* window, accept the default of **Customer Partitioning (for experts)** and click **Next**.
- In the *Expert Paritioner* window and the *System View* column on the left, click on a plus sign next to **Hard disks** to expand a list of all available disks as shown in Figure 13-3 on page 198.

RHEL or SLES?: If you are working only with SLES 11 SP2, you should apply the steps in this chapter to the `LNXADMIN` virtual machine on each member. When installing on member 1, add a logical volume mounted over `/var/`. Doing so is not described in this chapter, however, it is described in chapter 7 of *z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES 11 SP1*, on the Web at:

http://www.redbooks.ibm.com/abstracts/sg247931.html?Open

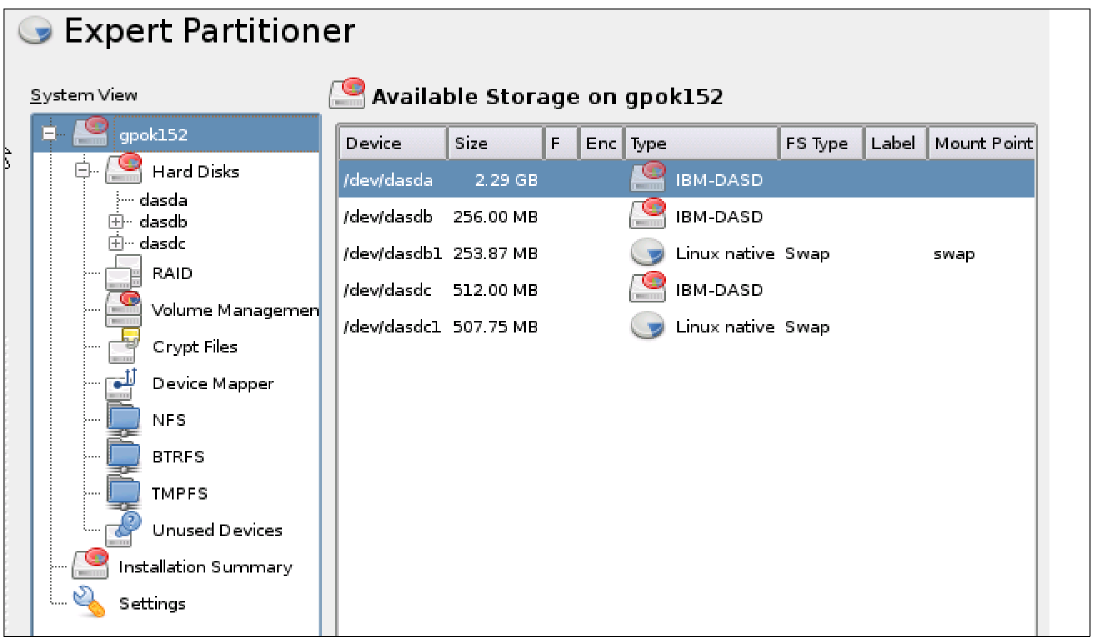


Figure 13-3 Expert partitioner - Hard Disks

- Double-click the **IBM-DASD** that corresponds to the `100` minidisk in the *Available Storage* section. In this example it is `/dev/dasda`.
- In the *Hard Disk: /dev/dasda* section, click on **Add** button to add a partition.
- In the window *Add Partition on /dev/dasda*, set partition size to **Maximum Size** and click **Next**.
- On the next screen, accept the defaults (Format partition, Ext3 file system, Mount partition and a *Mount Point* of / as shown in Figure 13-4 on page 199. Click **Finish**.

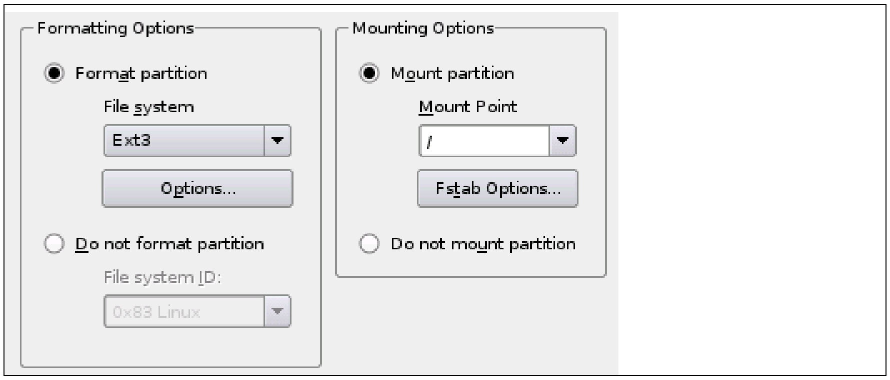


Figure 13-4 Add partition on /dev/dasda

- The two virtual disks, `/dev/dasdb1` and `/dev/dasdc1`, should be recognized as swap devices. The *Expert Partitioner* panel should now look as it does in Figure 13-5 on page 199. When it is correct, click **Accept**.

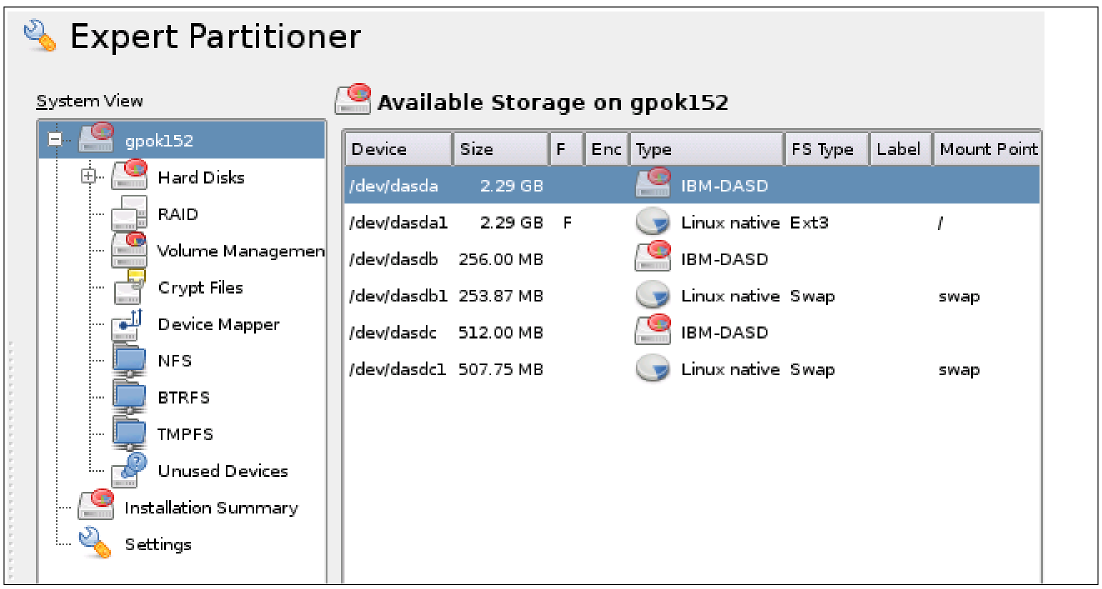


Figure 13-5 Disk allocations in the Expert Partitioner

- Back on the *Installation Settings* window, click **Software**. You may get a warning window about low disk space - this is OK. Deselect all items except **Base Server, 32bit Runtime Environment, Help and Support Documentation** and **Minimal System** as shown in Figure 13-6 on page 200. When finished, click **OK**.

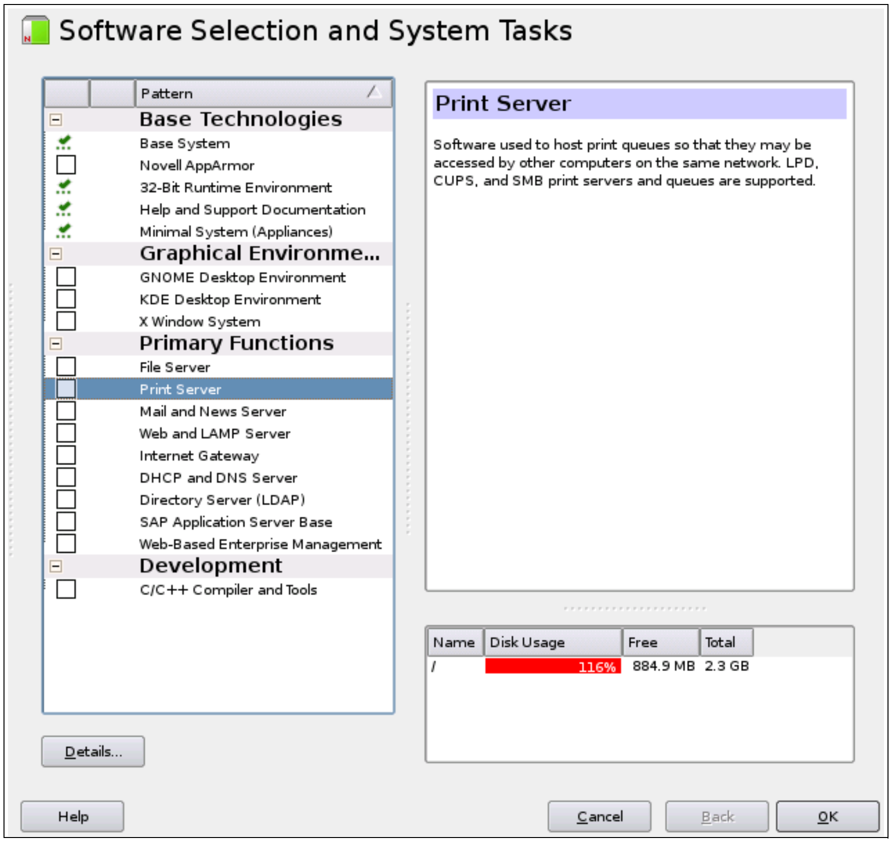


Figure 13-6 Choosing software groups

- In *Expert* tab click on **Default Runlevel** near the bottom. Choose **3: Full multiuser with network** as shown in Figure 13-7 on page 200. Click **OK**. If you receive a VNC warning, click **Yes**.

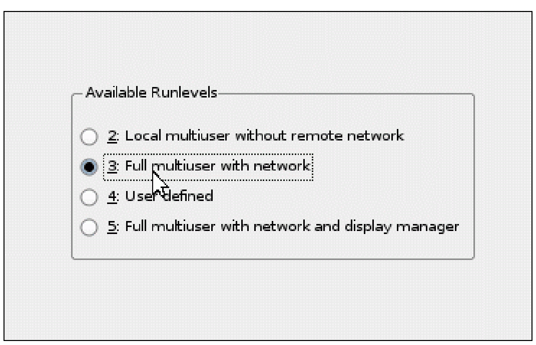


Figure 13-7 Setting default runlevel

- Click the *Overview* tab again. You should see a summary of the proposed installation:

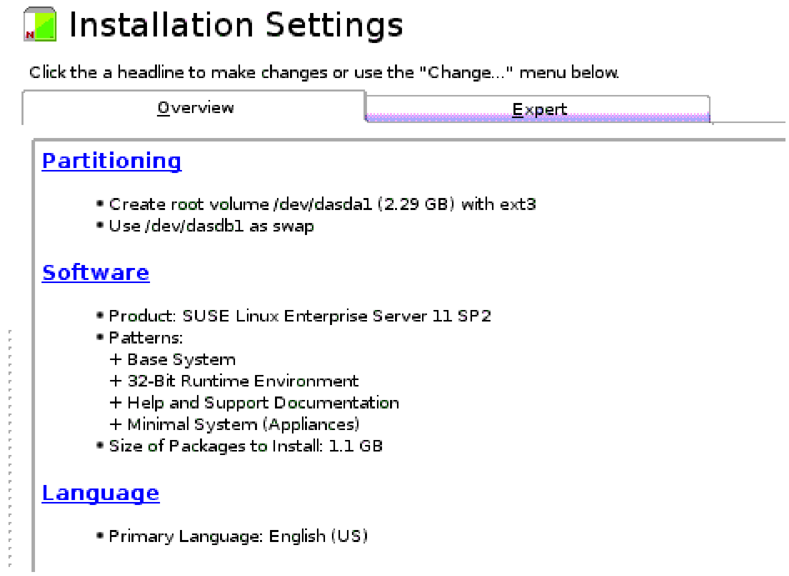


Figure 13-8 Installation Settings summary

- In the *Installation Settings* window, click **Install**.
- On the *Confirm Installation* window, click **Install**. This will begin the process of laying down RPMs onto disk. Copying the RPMs should take about 5-15 minutes. When copying of the RPMs is done, a few more windows will pass by and then your VNC viewer session will close.

13.3.1 Complete the Linux administration system installation

- Go back to your 3270 session. You will see messages indicating the Linux image is being restarted. You may need to clear the screen a number of times. At the end of the re-IPL, the VNC server is started again.

```
...
starting VNC server...
A log file will be written to: /var/log/YaST2/vncserver.log ...

***
***           You can connect to <host>, display :1 now with vncviewer
***           Or use a Java capable browser on http://<host>:5801/
***

(When YaST2 is finished, close your VNC viewer and return to this window.)

Active interfaces:

eth0      Link encap:Ethernet  HWaddr 02:00:0C:00:00:1D
          inet addr:9.60.18.152  Bcast:9.60.18.255  Mask:255.255.255.128
--
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0

NET: Registered protocol family 10
YaST2.call (2642): /proc/3208/oom_adj is deprecated, please use /proc/3208/oom_s
core_adj instead.
*** Starting YaST2 ***
Dec 11 10:58:10 gpok152 kernel: YaST2.call (2642): /proc/3208/oom_adj is depreca
```

```
ted, please use /proc/3208/oom_score_adj instead.
```

- Start the VNC viewer session again using the same credentials.
- On the *Password for the System Administrator "root"* window, set the root password twice and click **Next**. You may get a warning about the strength of the password.
- On the *Hostname and Domain Name* window, the *Hostname* and *Domain Name* fields should be filled in by values from parameter file (`LNXADMIN PARM-S11`). **Uncheck** the box *Change Hostname via DHCP*. Click **Next**.
- On the *Network Configuration* window, in the *Firewall* section, click **disable** which will disable the firewall. Click **Next**.
- On the *Test Internet Connectio*n window, if you do not have an Internet connection, select **No, Skip This Test**. However, Note that SuSE recommends you perform this test if possible. Click **Next**.
- On the *Network Services Configuration* window, accept the defaults and click **Next**. A certificate will be created.
- On the *User Authentication Method* window, select **Local (/etc/passwd)** and click **Next**.
- On the *Add a new local user* window, add a non-root userid for the primary system administrator(s) for this system and click **Next**.
- At the *Release Notes* window it is recommended that you read the information. Click **Next**.
- At the *Hardware Configuration* window, click **Next**.
- At the *Installation Complete* window, **deselect** the check box click *Clone this system for Autoyast*, then click **Finish**. The VNC viewer session will end.
- Go back to the 3270 session and you may have to clear the screen a few times. When you see the login prompt, **`DISCONNECT`** using the **`#CP`** prefix:

```
==> #cp disc
```

You have now installed the Linux administration system. You should be able to access the new system using SSH.

13.3.2 Verify the installation

To verify the installation, perform the following steps:

- Start an SSH session to the Linux administration system as **root**.
- Show the file system sizes with the **`df -h`** command:

```
# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/dasda1     2.3G  1.4G  809M  64% /
devtmpfs        499M  128K  499M   1% /dev
tmpfs           499M     0  499M   0% /dev/shm
```

- Verify that there are two swap spaces with the **`swapon -s`** commands:

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdb1                             partition       259960  0       -1
/dev/dasdc1                             partition       519924  0       -2
```

SLES 11 SP2 is now installed on the Linux administration system. The next step is to configure it.

13.4 Configure the SLES 11 SP2 Linux administration system

Now that your Linux administration system is installed, it must be configured. The following steps are involved:

13.4.1 Copy files to the Linux administration system

In section 7.2.1, "Copy RHEL 6.2 install tree to LNXADMIN" on page 125, the RHEL 6.2 install tree and files associated with this book were copied to `/var/nfs/`. In this section, you will copy the SLES 11 SP2 install files to the Linux administration system on member 1.

To do so, perform the following steps:

- Start an SSH session to `LNXADMIN` **on member 1** That is where the large logical volume is mounted over `/var/`.
- Verify there is enough disk space left:

```
# df -h | grep var
Filesystem           Size  Used Avail Use% Mounted on
/dev/dasda1          1.8G  1.3G  391M  78% /
tmpfs                498M     0  498M   0% /dev/shm
/dev/mapper/var_vg-var_lv
                      16G  2.9G   12G  20% /var
```

  In this example there is 12 GB of disk space free.

- Change to the directory `/var/nfs/` and list the files:

```
# cd /var/nfs
# ls -F
CKB-VM62/  rhel62/
```

  This shows that there are directories for the files associated with this book and for the RHEL 6.2 installation from when the member 1 Linux administration system was configured.

- Create the directory `sles11sp2/`:

```
# mkdir sles11sp2
```

- Change into the new directory and copy the DVD ISO image(s) from the PC NFS server to this directory. This step may take some time.

```
# cd sles11sp2
# scp 9.60.18.240:/nfs/sles11sp2/*.iso .
The authenticity of host '9.60.18.240 (9.60.18.240)' can't be established.
RSA key fingerprint is f5:43:ce:f3:44:35:81:b2:f5:9a:5e:06:f6:fb:46:56.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '9.60.18.240' (RSA) to the list of known hosts.
Password:
SLES-11-SP2-DVD-s390x-RC3-DVD1.iso           100% 3024MB  11.2MB/s   04:30
SLES-11-SP2-DVD-s390x-RC3-DVD2.iso           100% 4698MB  11.2MB/s   07:00
```

- Check the size of the /var/ file system again:

```
# df -h | grep var
/dev/mapper/var_vg-var_lv
                       16G   11G  4.4G  71% /var
```

The file system has filled up significantly from copying the two DVD ISO images.

- Use the **scp** command to copy the script **clone.sh** from the member 1 Linux administration system file to /usr/local/sbin/ on this system. It will be used later for cloning:

```
# cd /usr/local/sbin
# scp 9.60.18.151:/var/nfs/CKB-VM62/sles11sp2 .
The authenticity of host 'gpok151 (9.60.18.152)' can't be established.
RSA key fingerprint is eb:ac:2a:1c:d6:2c:8f:6d:77:8c:63:54:55:1f:30:d1.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'gpok152,9.60.18.151' (RSA) to the list of known hosts.
Password:
clone.sh                                      100%   13KB  12.6KB/s   00:00
```

- Export with NFS. Edit the /etc/exports file and add the newly populated directory for export:

```
# cd /etc
# vi exports
/var/nfs/rhel61             *(ro,sync)
/var/nfs/CKB-VM62           *(ro,sync)
/var/nfs/sles11sp2          *(ro,sync)
```

- Activate the changes with the **exportfs -a** command:

```
# exportfs -a
```

- Test mounting the newly exported file system locally:

```
# ls /mnt
# mount localhost:/var/nfs/sles11sp2/ /mnt
# ls /mnt
SLES-11-SP2-DVD-s390x-RC3-DVD1.iso     SLES-11-SP2-DVD-s390x-RC3-DVD2.iso
# umount /mnt
```

- End the SSH session to the Linux administration system on member 1:

```
# exit
logout
Connection to gpok151 closed.
```

You have now copied the files you will need for SLES 11 SP2 and exported them with NFS.

13.4.2 Reset the install source location

When SLES 11 SP2 is installed, the location of the *install source* is remembered. In this case, it is the PC NFS server. Now that the DVD ISO image has been copied from the PC NFS server to the member 1 Linux administration system, you can reset the install source location to point to that system. To do this task, perform the following steps:

- Start an SSH session as root on the Linux administration system on member 2.
- Invoke yast:

```
# yast
```

- Accept the default of **Software** in the left column and using the arrow keys, select **Software Repositories** in the right column and press **Enter**.

```
 +----------------------------------------------------------------------------+
 ¦                             YaST2 Control Center                           ¦
```

```
  +----------------------------------------------------------------------------+

  +---------------------+ +----------------------------------------------------+
  ¦Software             ¦ ¦Online Update                                       ¦
  ¦Hardware             ¦ ¦Software Management                                 ¦
  ¦System               ¦ ¦Add-On Products                                     ¦
  ¦Network Devices      ¦ ¦Installation into Directory                         ¦
  ¦Network Services     ¦ ¦Media Check                                         ¦
  ¦Security and Users   ¦ ¦Online Update Configuration                         ¦
  ¦Support              ¦ ¦Patch CD Update                                     ¦
  ¦Miscellaneous        ¦ ¦Software Repositories                               ¦
...
```

- At the top of the *Configured Software Repositories* panel, you should see a single entry for SLES 11 SP2. Change the installation source by using the Tab key to move to the **Edit** button at the bottom and press **Enter**.

```
  Configured Software Repositories
                                                           View
                                                           All repositories¦â
  +-----------------------------------------------------------------------------+
  ¦Priority     ¦Enabled¦Autorefresh¦Name                                       ¦
  ¦ 99 (Default)¦   x   ¦     x     ¦SUSE-Linux-Enterprise-Server-11-SP2 11.2¦
  ¦                                                                             ¦
...
```

- Set the *Path to ISO Image* to `/var/nfs/sles11sp2/SLES-11-SP2-DVD-s390x-RC3-DVD1.iso` Tab to **OK** and press **Enter**:

```
  ¦           +--------------------------------------------------+           ¦
  ¦           ¦                    NFS Server                    ¦           ¦
  ¦           ¦Repository Name                                   ¦           ¦
  ¦           ¦SUSE-Linux-Enterprise-Server-11-SP2 11.2.2-¦¦¦¦¦¦¦¦           ¦
  ¦           ¦ (x) Edit Parts of the URL  ( ) Edit Complete URL ¦           ¦
  ¦           ¦Server Name                                       ¦           ¦
  ¦           ¦9.60.18.151¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦[Browse...]¦           ¦
  ¦           ¦Path to Directory or ISO Image                    ¦           ¦
  ¦           ¦/var/nfs/sles11sp2/SLES-11-SP2-DVD-s39>[Browse...]¦           ¦
  ¦           ¦[x] ISO Image                                     ¦           ¦
  ¦           ¦[ ] NFS v4 Protocol                               ¦           ¦
  ¦           ¦                  [OK]  [Cancel]                  ¦           ¦
  ¦           +--------------------------------------------------+           ¦
...
Properties
[x] Enabled              Priority
[x] Automatically Refresh   v 99^
[ ] Keep Downloaded Packages
[Add][Edit][Delete]                                    [GPG Keys...][Refreshâ]
[Help]                                         [Cancel]                 [ OK ]

Repository Name
  ----------------------------------------------------------------------------
  Path to ISO Image
  /nfs/sles11sp2/SLES-11-SP1-DVD-s390x-GM-DVD1.iso-----------------[Browse...]
```

- In the License Agreement panel, move to **Yes** and press **Enter**. to select. Move to **Next** and press **Enter**:

```
(x) Yes, I Agree to the License Agreement
( ) No, I Do Not Agree
```

- Confirm the edit by choosing **Yes**.

- From the *Edit* entry, tab to **OK** and press **Enter**.
- Leave YaST by moving to **Quit**.

You have now changed the Linux administration system on member 2 to point to the SLES 11 SP2 software repository on the Linux administration system on member 1. Ideally, you should be able to retire the PC NFS server because all necessary files are on System z.

13.4.3 Turn off unneeded services

There are a number of services which are started in a SLES11 SP2 minimal system. Some of these can safely be turned off. To do so, perform the following steps:

- View which services are on in run level 3 with the following **chkconfig** command:

```
# chkconfig -l | grep 3:on
auditd                    0:off  1:off  2:off  3:on   4:off  5:on   6:off
cron                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
dbus                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
earlysyslog               0:off  1:off  2:on   3:on   4:off  5:on   6:off
fbset                     0:off  1:on   2:on   3:on   4:off  5:on   6:off
haldaemon                 0:off  1:off  2:on   3:on   4:off  5:on   6:off
irq_balancer              0:off  1:on   2:on   3:on   4:off  5:on   6:off
network                   0:off  1:off  2:on   3:on   4:off  5:on   6:off
network-remotefs          0:off  1:off  2:on   3:on   4:off  5:on   6:off
nfs                       0:off  1:off  2:off  3:on   4:off  5:on   6:off
nscd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
postfix                   0:off  1:off  2:off  3:on   4:off  5:on   6:off
purge-kernels             0:off  1:off  2:off  3:on   4:off  5:on   6:off
random                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
rpcbind                   0:off  1:off  2:off  3:on   4:off  5:on   6:off
smartd                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
splash                    0:off  1:on   2:on   3:on   4:off  5:on   6:off  S:on
splash_early              0:off  1:off  2:on   3:on   4:off  5:on   6:off
sshd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
syslog                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
xinetd                    0:off  1:off  2:off  3:on   4:off  5:on   6:off
```

- Turn off the following services using the **chkconfig** command:

```
# chkconfig fbset off
# chkconfig network-remotefs off
# chkconfig postfix off
# chkconfig splash off
# chkconfig splash_early off
# chkconfig smartd off
# chkconfig xinetd off
```

- Review which services are now configured to start in run level 3 with the following command:

```
# chkconfig -l | grep 3:on
auditd                    0:off  1:off  2:off  3:on   4:off  5:on   6:off
cron                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
dbus                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
earlysyslog               0:off  1:off  2:on   3:on   4:off  5:on   6:off
haldaemon                 0:off  1:off  2:on   3:on   4:off  5:on   6:off
irq_balancer              0:off  1:on   2:on   3:on   4:off  5:on   6:off
network                   0:off  1:off  2:on   3:on   4:off  5:on   6:off
nfs                       0:off  1:off  2:off  3:on   4:off  5:on   6:off
nscd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
purge-kernels             0:off  1:off  2:off  3:on   4:off  5:on   6:off
```

```
random                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
rpcbind                   0:off  1:off  2:off  3:on   4:off  5:on   6:off
sshd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
syslog                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
```

13.4.4 Apply service if necessary - online update

You may want to apply service using YaST Online Update. Internet access was not available during the writing of this book, so step-by-step details are not available.

If you have access to the Internet, or an online update source, invoke **yast** => **Software** => **Online update**.

13.4.5 Install the cmsfs package

The **clone.sh** script requires the **cmsfs** package, written by Rick Troth, in order read CMS files. cmsfs package is part of SLES11 distribution. To do so, perform the following steps:

- Install cmsfs using the **zypper install** command:

```
# zypper install cmsfs
...
Continue? [y/n/?] (y): y
Retrieving package cmsfs-1.1.8-6.2.s390x (1/1), 33.0 KiB (148.0 KiB unpacked)
Installing: cmsfs-1.1.8-6.2 [done]
```

  You should see some screens flash by as the cmsfs RPM is installed.

- Verify that it is installed with the **rpm -q** command:

```
# rpm -q cmsfs
cmsfs-1.1.8-6.2
```

- To test that the cmsfs package is properly installed, see if you can read the LNXADMIN PARM-S11 parameter file. Bring the 191 disk online using the **chccwdev -e** command:

```
# chccwdev -e 191
Setting device 0.0.0191 online
Done
```

- View the DASD with the **lsdasd** command:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0300   active      dasdb     94:4    FBA   512    256MB     524288
0.0.0301   active      dasdc     94:8    FBA   512    512MB     1048576
0.0.0191   active      dasdd     94:12   ECKD  4096   350MB     89820
```

  This shows that the 191 disk is /dev/dasdd.

- Test the **cmsfslst** command to list the files on the 191 disk:

```
# cmsfslst -d /dev/dasdd
FILENAME FILETYPE FM FORMAT LRECL       RECS     BLOCKS      DATE     TIME
         DIRECTOR P0 F         64         36          1  9/27/2011 13:37:13
         ALLOCMAP P0 F       4096          3          3  9/27/2011 13:37:13
CHPW610  XEDIT    B1 V         72        190          3  7/28/2011 13:21:03
CPFORMAT EXEC     B1 V         79        249          3  7/28/2011 13:21:03
...
```

- Test the **cmsfscat** command to list the contents of the file:

```
# cmsfscat -d /dev/dasdd -a lnxadmin.parm-s11
```

```
ramdisk_size=65536 root=/dev/ram1 ro init=/linuxrc TERM=dumb bootp=1
HostIP=9.60.18.152 Hostname=gpok152.endicott.ibm.com
...
```

You should now have `cmsfs` installed and briefly tested.

13.4.6 Enable the vmcp and cmm modules

The **vmcp** module/command allows z/VM CP commands to be issued from Linux. It is critical to the functioning of the **clone.sh** script. The **cmm** module allows Linux to do cooperative memory management (aka CMM1) with z/VM. This feature must be enabled on both the Linux and z/VM sides for it to function.

To configure the **vmcp** and **cmm** modules to be loaded at boot time, perform the following steps:

- Change directory to `/etc/sysconfig/`:

```
# cd /etc/sysconfig
```

- Edit the file `kernel` and add the module names to the variable `MODULES_LOADED_ON_BOOT` (around line 30):

```
# vi kernel
...
## Type:               string
## ServiceRestart:     boot.loadmodules
#
# This variable contains the list of modules to be loaded
# once the main filesystem is active
# You will find a few default modules for hardware which
# can not be detected automatically
#
MODULES_LOADED_ON_BOOT="vmcp cmm"
...
```

- Save the file.

You should be able to issue CP commands using the **vmcp** command after your system is rebooted.

13.4.7 Set system to halt on SIGNAL SHUTDOWN

The `Ctrl-Alt-Del` key sequence is simulated by z/VM when it issues a **SIGNAL SHUTDOWN** command. Rather than rebooting, you want your system to halt (shutdown). 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
...
# what to do when CTRL-ALT-DEL is pressed
ca::ctrlaltdel:/sbin/shutdown -h -t 4 now
...
```

This change will be picked up when the system is rebooted.

13.4.8 Modify zipl.conf

Two changes are recommended to the default `zipl.conf` file:

1. Add the parameters `vmpoff=LOGOFF` and `vmhalt=LOGOFF`. These instruct the z/VM virtual machine to be logged off when Linux is shut down. This can be convenient for shutting the z/VM system down more efficiently and also for getting a refreshed 3270 emulator session.
2. The `timeout` value is modified to 3 seconds. When SLES 11 SP2 boots, the default time that is allowed to enter a menu command is 10 seconds. Because the default value is most commonly used, this 10 seconds delays the starting of Linux. It is backed up to three seconds.

To modify the `zipl.conf` file, perform the following steps:

- Make a copy of the original `/etc/zipl.conf` file

```
# cp zipl.conf zipl.conf.orig
```

- Add the strings **vmpoff=LOGOFF** and **vmhalt=LOGOFF** to the parameter line in the `[Linux]` section and set **timeout=3** in the *menu* section near the bottom:

```
# vi zipl.conf   // add the vmpoff and vmhalt, change timeout to 3
# Modified by YaST2. Last modification on Sun Dec 11 10:57:08 EST 2011
[defaultboot]
defaultmenu = menu

###Don't change this comment - YaST2 identifier: Original name: linux###
[SLES11_SP2]
    image = /boot/image-3.0.13-0.9-default
    target = /boot/zipl
    ramdisk = /boot/initrd-3.0.13-0.9-default,0x2000000
    parameters = "root=/dev/disk/by-path/ccw-0.0.0100-part1 vmpoff=LOGOFF vmhalt=LOGOFF
hvc_iucv=8 TERM=dumb resume=/dev/disk/by-path/ccw-0.0.0301-part1"

###Don't change this comment - YaST2 identifier: Original name: failsafe###
[FailsafeV1]
    image = /boot/image-3.0.13-0.9-default
    target = /boot/zipl
    ramdisk = /boot/initrd-3.0.13-0.9-default,0x2000000
    parameters = "root=/dev/disk/by-path/ccw-0.0.0100-part1 hvc_iucv=8 TERM=dumb
noresume x11failsafe"


:menu
    default = 1
    prompt = 1
    target = /boot/zipl
    timeout = 3
    1 = SLES11_SP2
    2 = FailsafeV1
    3 = ipl
...
```

- Write the changes to the boot record with the **zipl** command:

```
# zipl
Using config file '/etc/zipl.conf'
Building bootmap in '/boot/zipl'
Building menu 'menu'
Adding #1: IPL section 'SLES11_SP2' (default)
Adding #2: IPL section 'FailsafeV1'
Adding #3: IPL section 'ipl'
Preparing boot device: dasda (0100).
Done.
```

These changes will be utilized the next time Linux is rebooted.

13.4.9 Reboot the system

You should now reboot the system to test the changes:

```
# reboot

Broadcast message from root (pts/0) (Wed Oct 19 16:03:44 2011):

The system is going down for reboot NOW!
```

Your system should be back in a few minutes. You are now done customizing the Linux administration system Linux image.

13.4.10 Verify the changes

To verify the changes, perform the following steps:

- Start an SSH session as **root** to the Linux administration system.
- Test the **vmcp** command with the CP **QUERY NAMES** command:

```
# vmcp q n
RH62GOLD - SSI , LINUX157 - SSI , TCPIP    - DSC , DTCVSW2  - DSC
DTCVSW1  - DSC , VMSERVR  - DSC , VMSERVU  - DSC , VMSERVS  - DSC
OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
LNXADMIN -L0003
VSM     - TCPIP
```

- Confirm that both of your swap spaces are operational:

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdf1                             partition       259956  0       -1
/dev/dasdg1                             partition       519924  0       -2
```

Congratulations! You have installed and configured a SLES 11 SP2 Linux system onto the Linux administration system. The next step is to install and configure the SLES 11 SP2 golden image.

14

Install the SLES 11 SP2 golden image

"I never think of the future. It comes soon enough."

— Albert Einstein

RHEL or SLES?: If you are working only with RHEL 6.2, you can skip this chapter as you should have completed Chapter 8, "Install and configuring the RHEL 6.2 golden image" on page 135.

This chapter describes how to install SLES 11 SP2 onto the virtual machine `S112GOLD`, which is referred to as the *golden image*. The golden image is the copy of Linux that will be cloned. Normally the system is shut down and the virtual machine logged off because it is not recommended to clone a running Linux system.

In this example, the golden image is given two 3390-3s at minidisk addresses `100` and `101`. This allows for about 4.5GB of disk space for each Linux system. If you want to increase that size, larger volumes such as 3390-9s be used to give approximately 14 GB, but minidisks at addresses `100` and `101` must still be defined for the **`clone.sh`** script to function.

To install and configure the golden image, perform the following steps:

- "Create the S112GOLD virtual machine" on page 211
- "Create the S112GOLD parameter file" on page 212
- "Install the golden image" on page 213
- "Configure the golden image" on page 227

14.1 Create the S112GOLD virtual machine

The golden image has a default memory size of 256 MB, and it is given class G privilege. It is given the following minidisks:

`100` Half of the disk space for the golden image.

`101` The other half of the disk space.

To define the `S112GOLD` virtual machine, perform the following steps:

- **Logon to** `MAINT`
- Edit the `USER DIRECT` file, add 6 new lines at the bottom of the file and create the following user directory entry. Set the 3390 disk labels to those appropriate for your system. In this example, the last third of ***UM63A2*** and the first third of a new 3390-9 volume, ***UM63A9***, are used:

```
==> x user direct c
====> bot
====> a 6
*
USER S112GOLD LNX4VM 256M 1G G
 INCLUDE LNXDFLT
 OPTION LNKNOPAS APPLMON
MDISK 100 3390 0001 3338 JM628E MR LNX4VM LNX4VM LNX4VM
MDISK 101 3390 0001 3338 JM628F MR LNX4VM LNX4VM LNX4VM
```

- When the disk layout is correct run **DIRECTXA** to bring the changes online:

```
==> directxa user
z/VM USER DIRECTORY CREATION PROGRAM - VERSION 6 RELEASE 2.0
EOJ DIRECTORY UPDATED AND ON LINE
HCPDIR494I User directory occupies 107 disk pages
```

You have now defined the virtual machine that will contain the Linux golden image.

14.2 Create the S112GOLD parameter file

A SLES 11 SP2 parameter file will be needed for this new virtual machine. You will need to change the IP address (`HostIP` variable) and the host name (`Hostname` variable). In this example those are ***9.60.18.145*** and ***gpok145***.

Also, the Linux administration system is used to provide the SLES 11 SP2 install directory using NFS by setting the `Install` variable.

Perform the following steps:

- Logon to `LNXMAINT`.
- Copy the `LNXADMIN` parameter file to one with a file name of `S112GOLD` on the `LNXMAINT 192` (D) disk:

```
==> copy lnxadmin parm-s11 d s112gold = =
```

> **Important:** If you decided to work with only layer 2 VSWITCHes instead of one layer two and one layer 3, make one additional change in the parameter file that follows:
>
> ```
> Layer2=1
> ```
>
> This will specify to Linux that the VSWITCH associated with the NIC at virtual device addresses `0600-0602` is Layer 2, not layer 3.

- Edit the new file and set the networking values correctly. These changes set the golden image's IP address and host name, and also points to the new install server on the Linux administration system on member 1 (***9.60.18.151*** in this example)

```
==> x s112gold parm-s11
ramdisk_size=65536 root=/dev/ram1 ro init=/linuxrc TERM=dumb
HostIP=9.60.18.145 Hostname=gpok145.endicott.ibm.com
Gateway=9.60.18.129 Netmask=255.255.255.128
Broadcast=9.60.18.255 Layer2=0
ReadChannel=0.0.0600  WriteChannel=0.0.0601  DataChannel=0.0.0602
Nameserver=9.0.2.11 portname=whatever portno=0
Install=nfs://9.60.18.151/nfs/sles11sp2/SLES-11-SP2-DVD-s390x-RC3-DVD1.iso

UseVNC=1  VNCPassword=12345678
InstNetDev=osa OsaInterface=qdio OsaMedium=eth Manual=0
```

When S112GOLD is logged onto, the new parameter file will be accessible on the A (191) disk.

14.3 Install the golden image

You should now be ready to begin the install onto the golden image. Linux will be installed onto the 100-101 minidisks. It will use 300-301 virtual disks for swapping. Most Linux virtual machines described in this book will have two read-write minidisks and two virtual disks. Disk 300 is 256MB and will act as a primary swap space. Only after it is full, disk 301, which is 512 MB, will be used. A minidisk partition will be created for a third swap space.

To install the golden image, perform the following steps:

- Logon to S112GOLD. When you logon, you should see messages indicating that a virtual NICs have been created starting at addresses 0600 and 0700 and that virtual disks 300 and 301 have been created

```
00: z/VM Version 6 Release 2.0, Service Level 1101 (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 06:02:01 EST MONDAY 12/12/11
00: Command complete
00: NIC 0600 is created; devices 0600-0602 defined
00: NIC 0600 is connected to VSWITCH SYSTEM VSW1
00: Command complete
00: NIC 0700 is created; devices 0700-0702 defined
00: NIC 0700 is connected to VSWITCH SYSTEM VSW2
z/VM V6.2.0    2011-11-15 11:26

DMSACP723I A (191) R/O
DMSACP113S C(592) not attached or invalid device address
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)
```

- You are prompted to IPL Linux, but since you have not installed Linux yet, answer **n**:

```
Do you want to IPL Linux from minidisk 100? y/n
==> n
```

- Use the **DEFINE STORAGE** command to move the memory size up to 1 GB (256 MB is not enough to complete the installation process), then re-IPL CMS:

```
==> def stor 1g
00: STORAGE = 1G
==> ipl cms
```

```
z/VM V6.2.0    2011-07-19 16:53

DMSACP723I A (191) R/O
DMSACP113S C(592) not attached or invalid device address
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
```

This shows that you have the resources necessary to install SLES 11 SP2.

14.3.1 Begin the SLES 11 SP2 installation

Follow these steps to begin the installation of `S112GOLD`.

- Run the **SLES11S2 EXEC**. You should see many screens of questions and answers scrolling by. If you had used the default parameter file shipped with SLES 11 SP2, you would have had to answer all the networking questions manually. With the proper parameters set in the file `S112GOLD PARM-S11`, the install process should proceed to where you access the install program using a VNC client:

```
==> sles11s2
00: 0000003 FILES PURGED
00: RDR FILE 0014 SENT FROM S112GOLD PUN WAS 0014 RECS 099K CPY  001 A NOHOLD NO
KEEP
00: RDR FILE 0018 SENT FROM S112GOLD PUN WAS 0018 RECS 0009 CPY  001 A NOHOLD NO
KEEP
00: RDR FILE 0022 SENT FROM S112GOLD PUN WAS 0022 RECS 194K CPY  001 A NOHOLD NO
KEEP
00: 0000003 FILES CHANGED
00: 0000003 FILES CHANGED
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 3.0.4-0.11-default (geeko@buildhost) (gcc version 4.3.4 Ýgcc-4_3-b
ranch revision 152973¨ (SUSE Linux) ) #1 SMP Mon Sep 5 20:20:34 UTC 2011 (88ecc7
a)
setup.7055fd: Linux is running as a z/VM guest operating system in 64-bit mode
Zone PFN ranges:
  DMA      0x00000000 -> 0x00080000
  Normal   empty
Movable zone start PFN for each node
early_node_mapÝ1¨ active PFN ranges
    0: 0x00000000 -> 0x00040000
PERCPU: Embedded 10 pages/cpu @0000000002684000 s11520 r8192 d21248 u40960
Built 1 zonelists in Zone order, mobility grouping on.  Total pages: 258560


Kernel command line: ramdisk_size=65536 root=/dev/ram1 ro init=/linuxrc TERM=dum
b                    HostIP=9.60.18.145 Hostname=gpok145.endicott.ibm.com
                     Gateway=9.60.18.129 Netmask=255.255.255.128
                     Broadcast=9.60.18.255 Layer2=0
                     ReadChannel=0.0.0600  WriteChannel=0.0.0601  DataChannel=0.
0.0602               Nameserver=9.0.2.11
                     portname=whatever
                     portno=0
                     Install=nfs://9.60.18.240/nfs/sles11sp2/SLES-11-SP2-DVD-s39
0x-RC3-DVD1.iso      UseVNC=1  VNCPassword=12345678
                     InstNetDev=osa OsaInterface=qdio OsaMedium=eth Manual=0

...
```

- The install system or *starter system* should continue to boot. You should see the message:

```
starting VNC server...
A log file will be written to: /var/log/YaST2/vncserver.log ...

***
***          You can connect to <host>, display :1 now with vncviewer
***          Or use a Java capable browser on http://<host>:5801/
***

(When YaST2 is finished, close your VNC viewer and return to this window.)

Active interfaces:

eth0      Link encap:Ethernet  HWaddr 02:00:0C:00:00:20
          inet addr:9.60.18.145  Bcast:9.60.18.255  Mask:255.255.255.128
--
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0

*** Starting YaST2 ***
```

- Use a VNC viewer through a Java-enabled browser, or a standalone VNC viewer. Connect to the VNC server (***9.60.18.145:1*** in this example). Enter the password specified in the parameter file (***12345678*** in this example).
- You could disconnect from the 3270 session, however messages to the console will be lost. It is recommended that you stay connected, but you may have to clear the screen periodically (or the install process may be delayed waiting for the screen to clear itself).

Now the graphical installation process should begin.

14.3.2 Begin YaST installation

Perform the following steps to install SLES 11 SP2:

- On the *Welcome* panel, choose your language and keyboard, (**English US** in this example). Read the License Agreement, choose **I Agree to the License Terms** and Click **Next**.
- The *Disk Activation* window should appears. Choose **Configure DASD Disks**.
- The *DASD Disk Management* window should appear: you will see all the DASD available to S112GOLD.
  a. Highlight each of the minidisks and virtual disks, 100, 101, 300 and 301 and click **Select or Deselect**.
  b. You should see a **Yes** appear next to them in the *Sel.* column on the left. Activate them by clicking **Perform Action** -> **Activate**. as shown in the left side of Figure 14-1 on page 216.
  c. Disks 100 and 101 must be formatted so that Linux can use them. Deselect disks 300 and 301 using the **Select or Deselect** button, so that 100 and 101 remain selected. Now click **Perform Action** -> **Format** as shown on the right side of the figure.

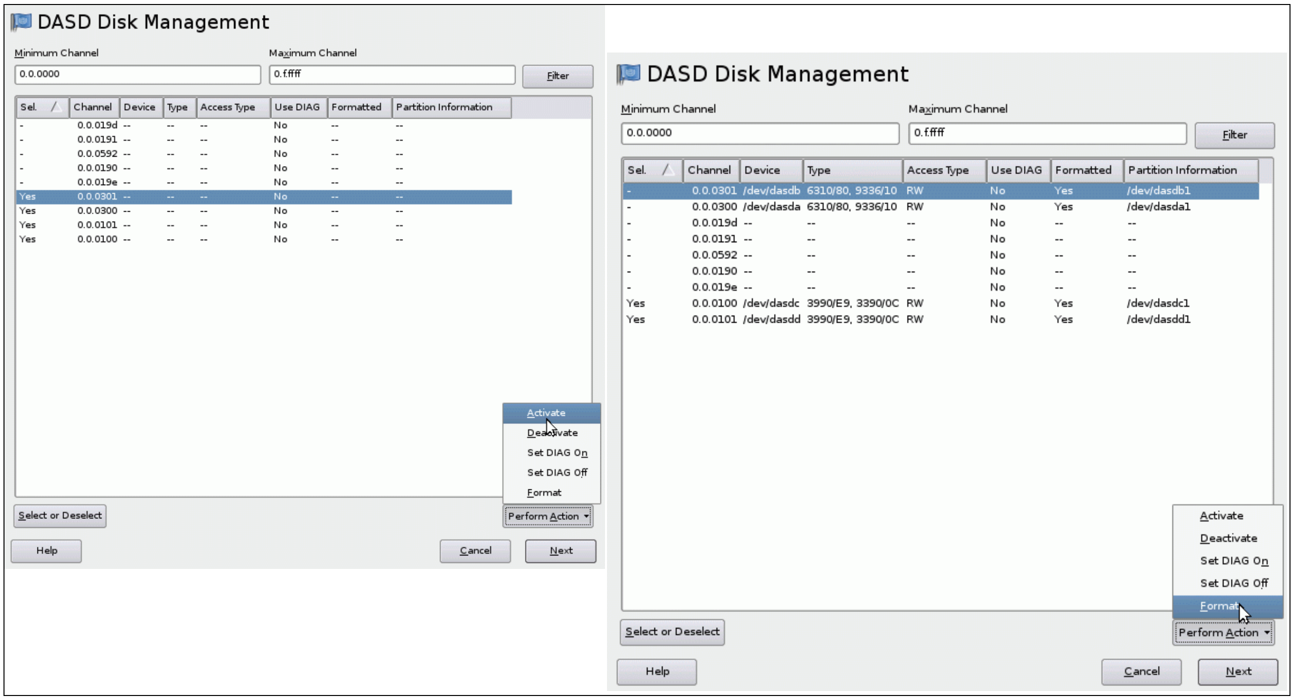


Figure 14-1 DASD available to S112GOLD

- You should see a window asking for 2 Parallel Formatted Disks. Click **OK**.
- Click **Yes** to the question *Really format the following disks?*
- A progress indicator window should appear displaying the progress of the parallel formats. This step can take 2-10 minutes depending on a number of factors.
- When the formatting is complete, click **Next** in the *DASD Disk Management* window.
- In the *Disk Activation* window click **Next** again.
- In the *Installation Mode* window accept the default of **New installation** and Click **Next**.
- The *Clock and Time Zone* window will appear. Choose your region and time zone and click **Next**.
- This will bring you to the *Installation Settings* window. Click **Partitioning** in the *Overview* tab. The *Preparing Hard Disk* window will appear.
- Accept the default of **Custom Partitioning (for experts)** and click **Next**. The *Expert Partitioner* window will appear as shown in Figure 14-2 on page 217.

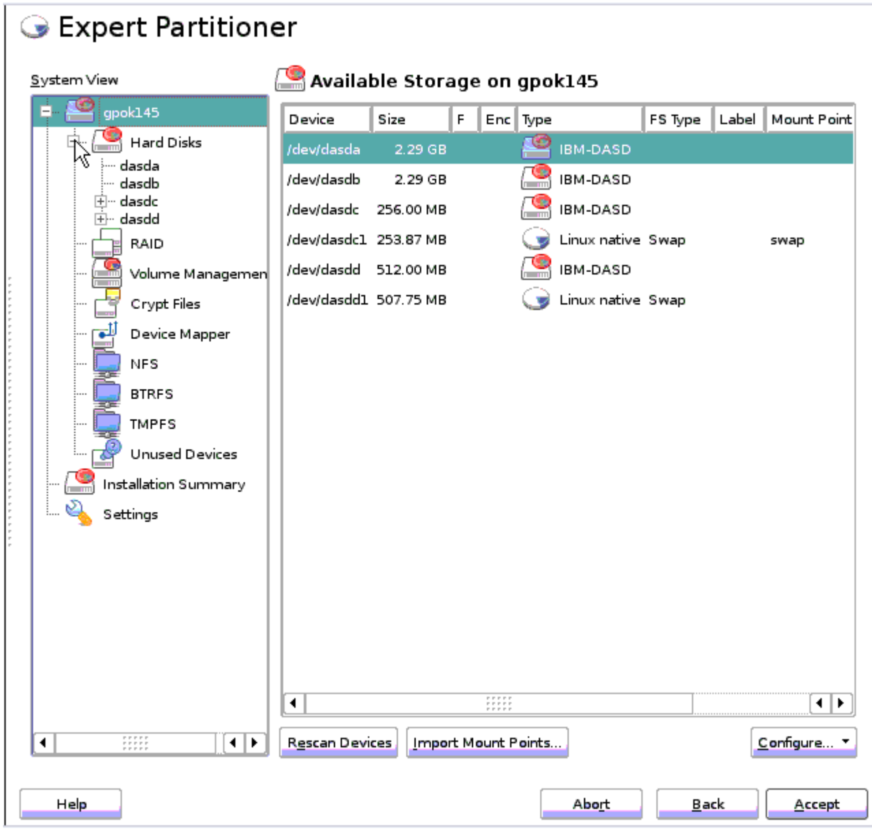


Figure 14-2 Disk partitioner - before customization

- Two partitions will be made on `/dev/dasda` and one partition will be made on `/dev/dasdb`.
  a. Click the plus sign (**+**) to the left of **Hard Disk**. It should show `dasda-dasdd` in the tree.
  b. Click on **dasda** on the left and then on the **Add** button. The window *Add Partition on /dev/dasd*a will appear.
- Accept the default of the **Custom size** radio button and set the size of **384 MB** as shown in Figure 14-3 on page 218. Click **Next**.

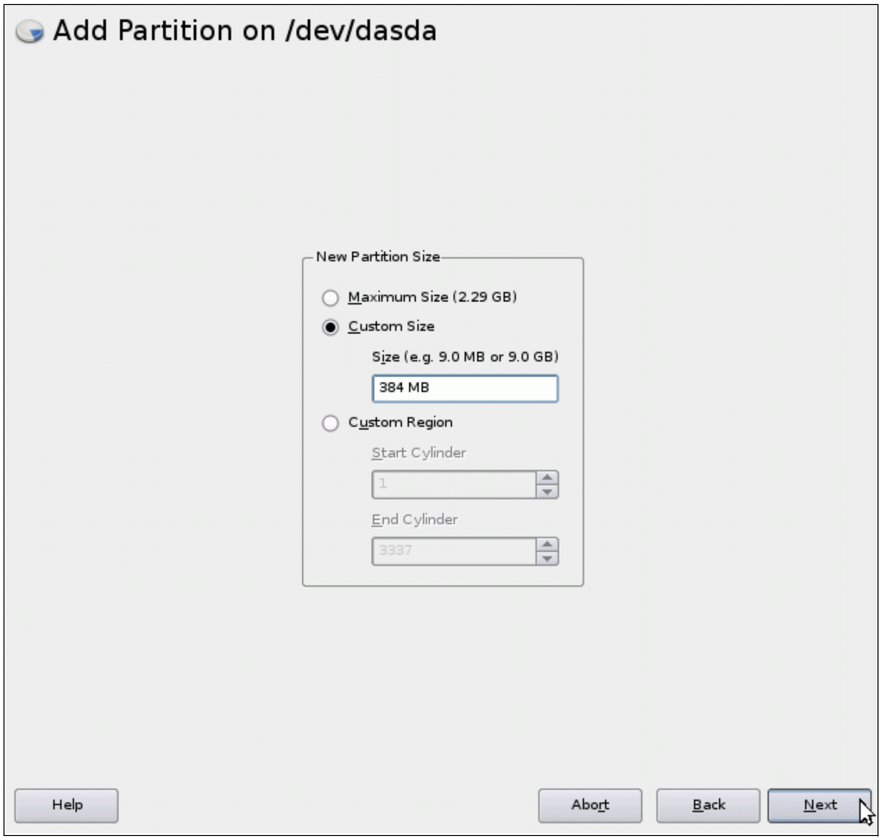


Figure 14-3 Add Partition on /dev/dasda

- You should see a new window *Add Partition on /dev/dasda*.
  - Accept the default of **Format Partition**,
  - Accept the default of a *File system* of type **ext3** in the *Formatting Options* section
  - Accept the default of a *Mount Point* of **/** (root file system) in the *Mounting Options* section.
  - Click **Finish**. This will create the partition `/dev/dasda1`.
- To create a second partition click **Add** while `dasda` is selected in the left tree. The window *Add partition on /dev/dasda* will appear.
- Click the **Maximum Size** radio button then click **Next**.
- In new *Add partition on /dev/dasda* window as shown in Figure 14-4 on page 219 perform the following steps:

  a. Click the **Do not format** radio button in the *Formatting Options* section.

  b. Click **Do not mount partition** radio button in the *Mounting options* section.

  c. Click **Finish**. This will create an empty partition `/dev/dasda2`.

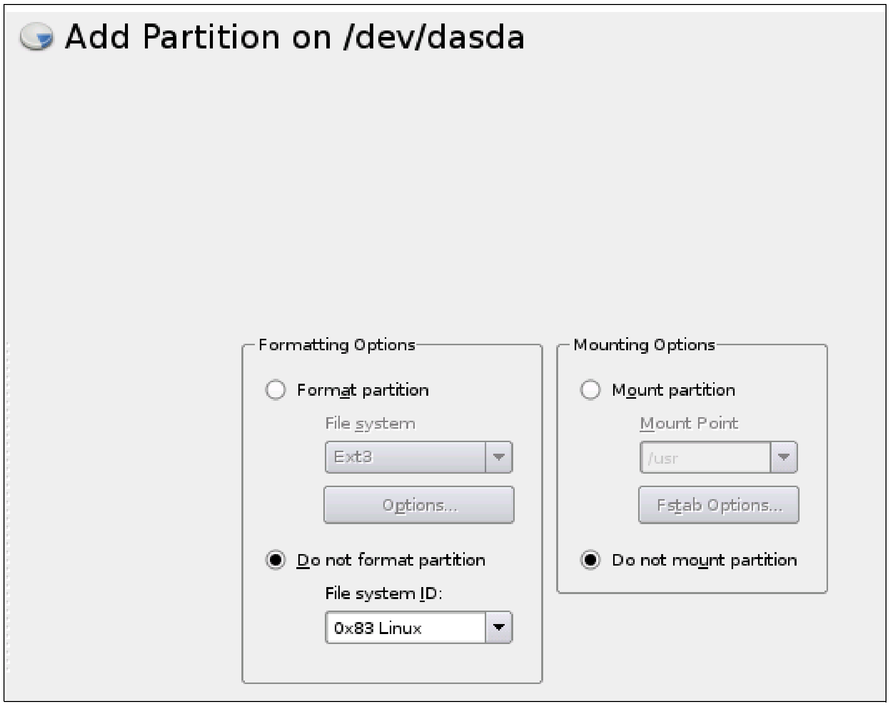


Figure 14-4 Add Partition on /dev/dasda

- Create an empty partition `/dev/dasdb1` in the same fashion as you did to create `/dev/dasda2` (Select `/dev/dasdb` => **Add** => Maximum Size => **Do not format/Do not mount partition**). Accept the default size of all the space.
- Now `/dev/dasda2` and `/dev/dasdb1` can be used to create a volume group. Click **Volume Management** as shown on Figure 14-5 on page 220.

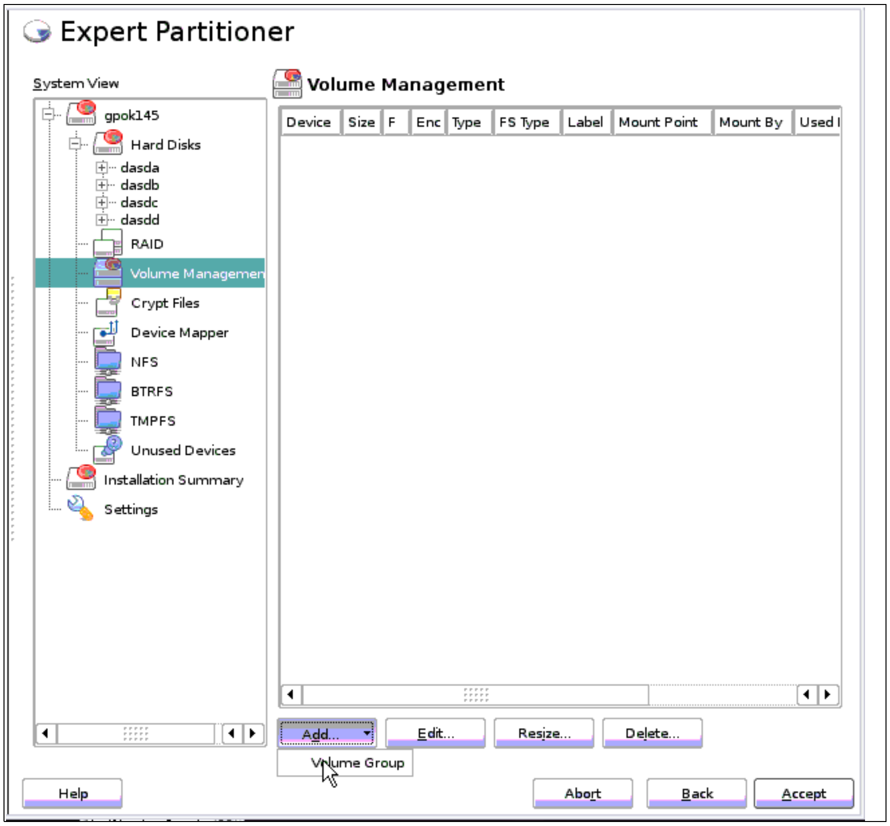


Figure 14-5 Creating logical volumes with Volume Management

- Click on **Add** => **Volume Group**.
- At the *Add Volume Group* window set the volume group name to **system-vg**.
- Click **Add** after selecting both */dev/dasda2* and */dev/dasdb1* to the *Selected Physical Volumes* as shown on Figure 14-6 on page 221. Click **Finish**.

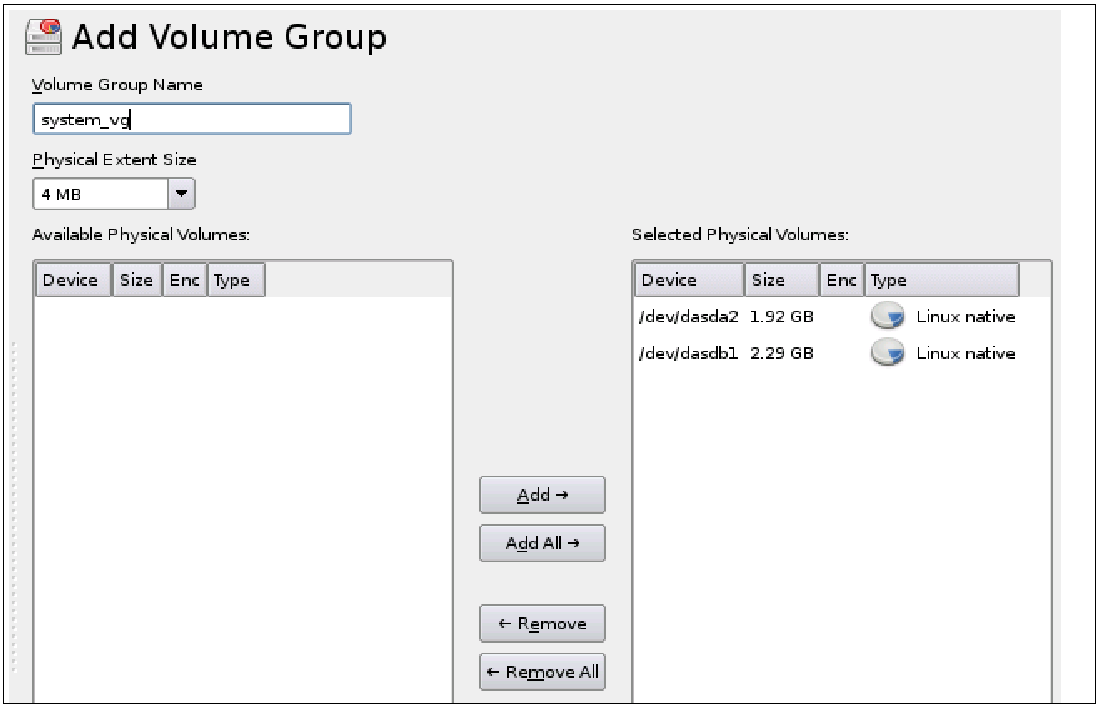


Figure 14-6 Add Volume Group

- Back in *Expert Partitioner*, click on the **plus sign** to the left of *Volume Management*. You should see the new **system-vg** volume group.
- Click on **system-vg** and the *Volume Group: /dev/system-vg* window will appear as shown on Figure 14-7 on page 222. It shows there are no logical volumes defined. Click on **Add**.

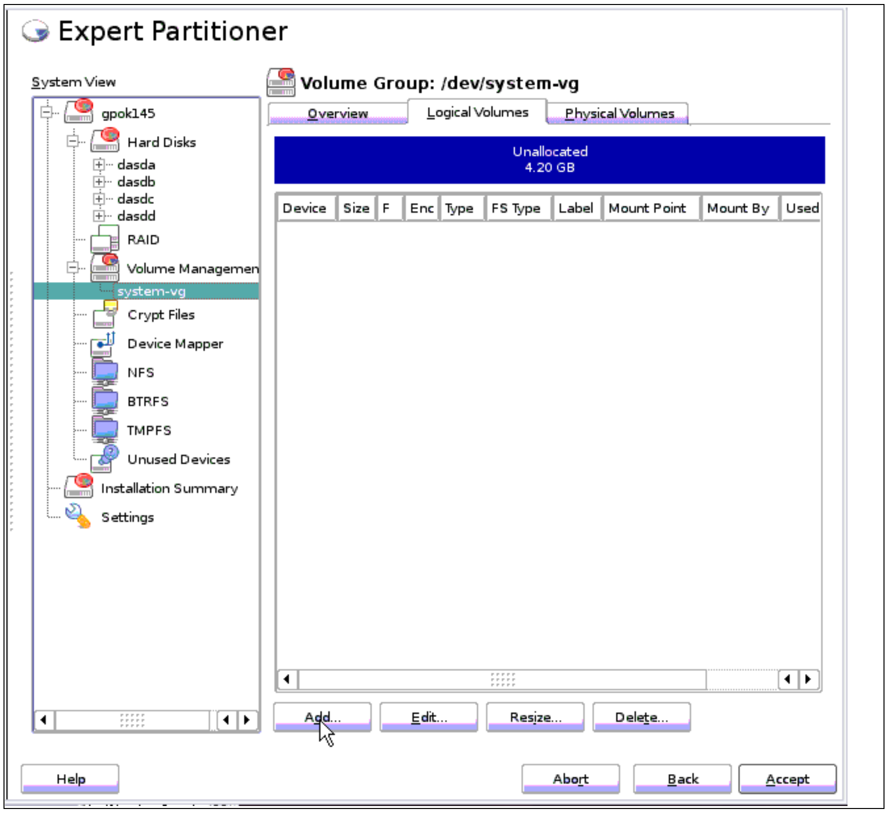


Figure 14-7 Volume Group view in Expert Partitioner.

- Enter **usr-lv** as the name for new logical volume and click **Next**.
- On the *Add Logical Volume* window, click the *Custom Size* radio button and enter a size of **2.5 GB**. Accept the default of one stripe. Click **Next.**
- Accept the defaults of **ext3** as the *File System* type and the *Mount point* of **/usr**. Click **Finish**.
- Create three more logical volumes similarly, using the following table:

Table 14-1 Logical volumes added to system-vg volume group

Mount point	Logical volume name	Size
`/usr/`	**usr-lv**	2.5 GB
`/var/`	**var-lv**	512 MB
`/opt/`	**opt-lv**	384 MB
`/tmp/`	**tmp-lv**	384 MB

You may choose other file systems and sizes. See 2.4, “Disk planning” on page 12 for more discussion.

- Back in the *Expert Partitioner* window, click on the top object in the *System View* (***gpok145*** in this example). You should see output similar to Figure 14-8 on page 223. You don’t have to format `/dev/dasdc1` (vdev `300`) and `/dev/dasdd1` (vdev `301`) because they were properly formatted as a Linux swap space by the **`SWAPGEN EXEC`**, and thus should be recognized as a swap space. Click **Accept**.

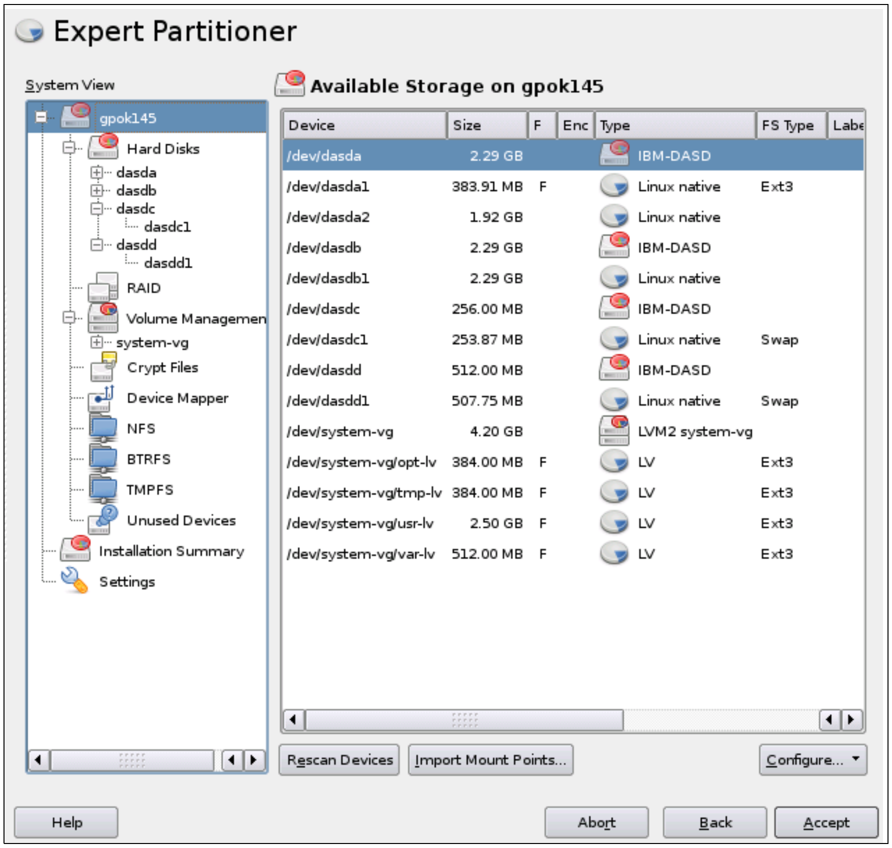


Figure 14-8 Disk partitioner - after customization

- In the *Installation Settings* window, select **Software**. You may see an *Disk Space Warning* window. If so, click **OK**.
- The *Software Selection* window opens. Leave **Base System, 32-Bit Runtime Environment**, **Help and Support Documentation** and **Minimal System** selected. Uncheck all other setting as showed in Figure 14-9 on page 224**,** click **OK**.

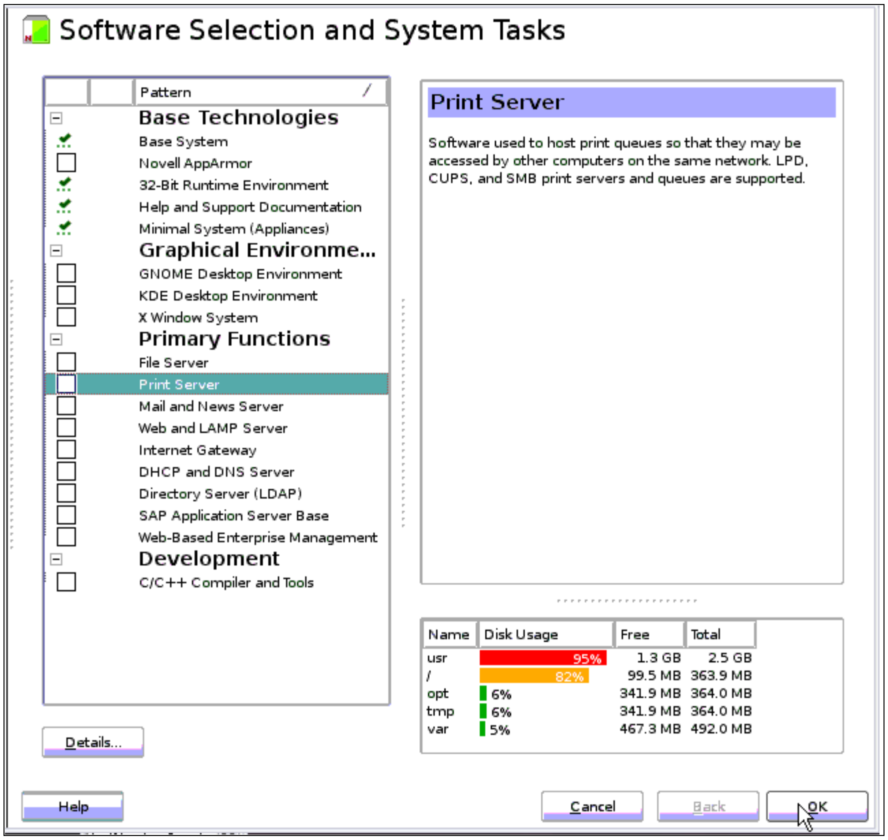


Figure 14-9 Software selection window

- In *Expert* tab move down to *Default Runlevel* and choose **3: Full multiuser with network** as shown in Figure 14-10 on page 224. Click **OK**. You will receive a VNC warning. Click **Yes**.

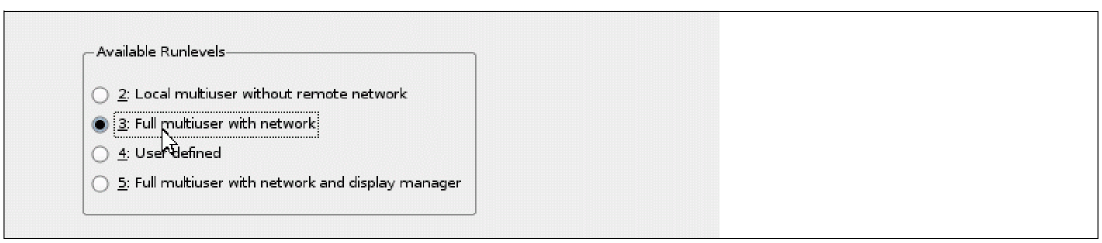


Figure 14-10 Default Runlevel

- You are now ready to begin copying the RPMs onto your root file system. In the *Installation Settings* window, check the settings and click **Install**.
- In the *Confirm Installation* window click **Install**.
- The SLES 11 SP2 system will be installed onto DASD. This should take about 5-20 minutes depending on a number of factors.
- The window Finishing Basic Installation will appear, then the VNC client will end and the system will halt. Back on the 3270 console, you should see:

```
01: HCPGSP2629I The virtual machine is placed in CP mode due to a SIGP stop from
 CPU 01.
00: HCPGSP2629I The virtual machine is placed in CP mode due to a SIGP stop from
 CPU 00.
```

14.3.3 Reboot the new Linux system from disk

Perform the following steps:

- Go back to the 3270 console and boot Linux from disk with the **IPL** command:

```
==> ipl 100
...
```

This time it boots from disk, not from the reader. You should see messages scrolling in the z/VM 3270 session. You will have to clear the screen a number of times. The install program will bring up a VNC server again to complete the installation.

```
...
starting VNC server...
A log file will be written to: /var/log/YaST2/vncserver.log ...

***
***           You can connect to 9.60.18.145, display :1 now with vncviewer
***           Or use a Java capable browser on  http://9.60.18.145:5801/
***

(When YaST2 is finished, close your VNC viewer and return to this window.)
...
```

- **Go back to the same VNC client** used for the first part of installation. If it was a browser either click **Login Again** or click the browser's **refresh** button until another VNC login screen appears. If you are using a VNC client then open that application again.
- **Log in** using the same VNC password (**12345678** in this example).
- In the Password for root user window, type the root password twice and click **Next**. *Don't forget* this password!
- In the *Hostname and Domain Name* window, both *Hostname* and *Domain Name* are entered by installer as they were specified in `S112GOLD PARM-S11` file. In this example ***gpok145*** is the host name and ***endicott.ibm.com*** is the domain name. Uncheck the **Change Hostname via DHCP** check-box. Click **Next**.
- In the *Network Configuration* window you will see *Firewall is enabled*, click on the word **disable** to disable it as shown in Figure 14-11 on page 226. All other values should be correct so just click **Next**.

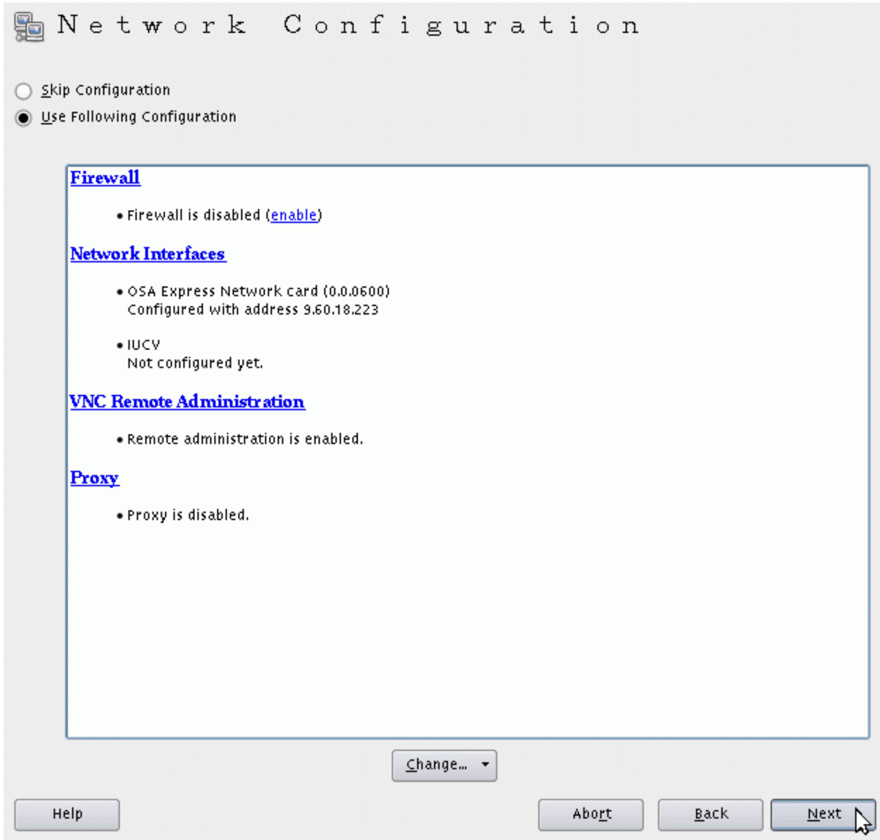


Figure 14-11 Disabling firewall in Network Configuration window

- In the *Test Internet Connection* window, if you do not have Internet access, select **No, skip this test** and click **Next**.
- In the *Network Services Configuration* window, accept the default and click **Next**. A certificate will be created.
- In the *User Authentication Method* window select **Local (/etc/passwd)** and click **Next**.
- In the *New Local User* window, add at least one user so as to have a *non-root* ID on all cloned systems. When you are done, click **Next**.
- In the *Writing the system configuration* window the **SuSEconfig** tool writes all your settings to disk.
- The next window will be *Release Notes*. After reviewing the release notes, click **Next**.
- In the *Hardware Configuration* window choose the **Skip Configuration** radio button and click **Next**.
- The last installation window is *Installation Completed*. Uncheck the box *Clone This System for Autoyast* check box and click **Finish.**

The VNC session should end. Return to the 3270 session and you may have to clear the screen a few times. Then you should see a login prompt. You are done installing Linux! You can disconnect from the 3270 session using the **DISCONNECT** command:

```
==> #cp disc
```

You can now access this Linux system with SSH. If you have a Windows desktop, but do not have an SSH client configured, see 3.1.1, “PuTTY: a free SSH client for Windows” on page 25.

14.4 Configure the golden image

Customize the golden image before cloning. The following high level steps are recommended.

- "Configure the VNC server"
- "Prepare for Online Update" on page 229
- "Turn off unneeded services" on page 230
- "Apply service - online update" on page 231
- "Configure /etc/inittab" on page 231
- "Configure SSH keys" on page 231
- "Modify zipl.conf" on page 233
- "Clean up temporary files" on page 234
- "Reboot the system and verifying changes" on page 234

14.4.1 Configure the VNC server

Often applications require a graphical environment. The `tightvnc` package is a Virtual Network Computing (VNC) server. It allows for a graphical environment to be set up easily using the **vncserver** command.

SLES 11 SP2 configures a VNC connection which starts by default. It is configured through the **xinetd** daemon. It is recommended that you disable it. VNC will be configured to run only when needed.

Perform the following steps:

- Start an SSH session as **root** to the golden image.
- VNC is the only service that **xinetd** starts by default (this can be verified with the **chkconfig --list** command). You can turn off the **xinetd** service with the **service xinetd stop** command for this session and the **chkconfig** command at boot time:

```
# service xinetd stop
Shutting down xinetd:                                              done
# chkconfig xinetd off
```

Start the VNC server manually

When you first start the VNC server, you are prompted to set a password. After it is set, this will be the password that you will need to connect to it from a VNC client. Perform the following steps:

- Start the VNC server:

```
# vncserver
You will require a password to access your desktops.

Password:
Verify:
Would you like to enter a view-only password (y/n)? n
xauth:  creating new authority file /root/.Xauthority

New 'X' desktop is gpok145:1

Creating default startup script /root/.vnc/xstartup
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/gpok145:1.log
```

- Stop the VNC server using the **-kill :1** argument:

```
# vncserver -kill :1
Killing Xvnc process ID 22541
```

- The `icewm` package allows for the `IceWM`, a window manager that is more usable than the Tiny Window Manager (**twm**) that VNC uses by default. It is therefore recommended that you change to **icewm**. The package should be already installed on your system. Verify the RPM was added:

```
# rpm -q icewm icewm-lite
icewm-1.2.36-1.35
icewm-lite-1.2.36-1.18
```

- Change the window manger from **twm** to **icewm** in the file `/root/.vnc/xstartup`:

```
# cd /root/.vnc/
# vi xstartup
#!/bin/sh

xrdb $HOME/.Xresources
xsetroot -solid grey
xterm -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
icewm &
```

- You may want to remove the `passwd` file so the cloned system does not have the same password as you just entered. If so, use the command **rm passwd**. If you remove the password file, when a system is cloned, the password will be prompted for the first time that the VNC server is initialized.

Verify the VNC server configuration

Perform the following steps:

- Start the VNC server again:

```
# vncserver

New 'X' desktop is gpok145:1

Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/gpok145:1.log
```

- Start a VNC client as shown on the left side of Figure 14-12. Enter the password and the resulting VNC session is shown on the right side of Figure 14-12 on page 229.

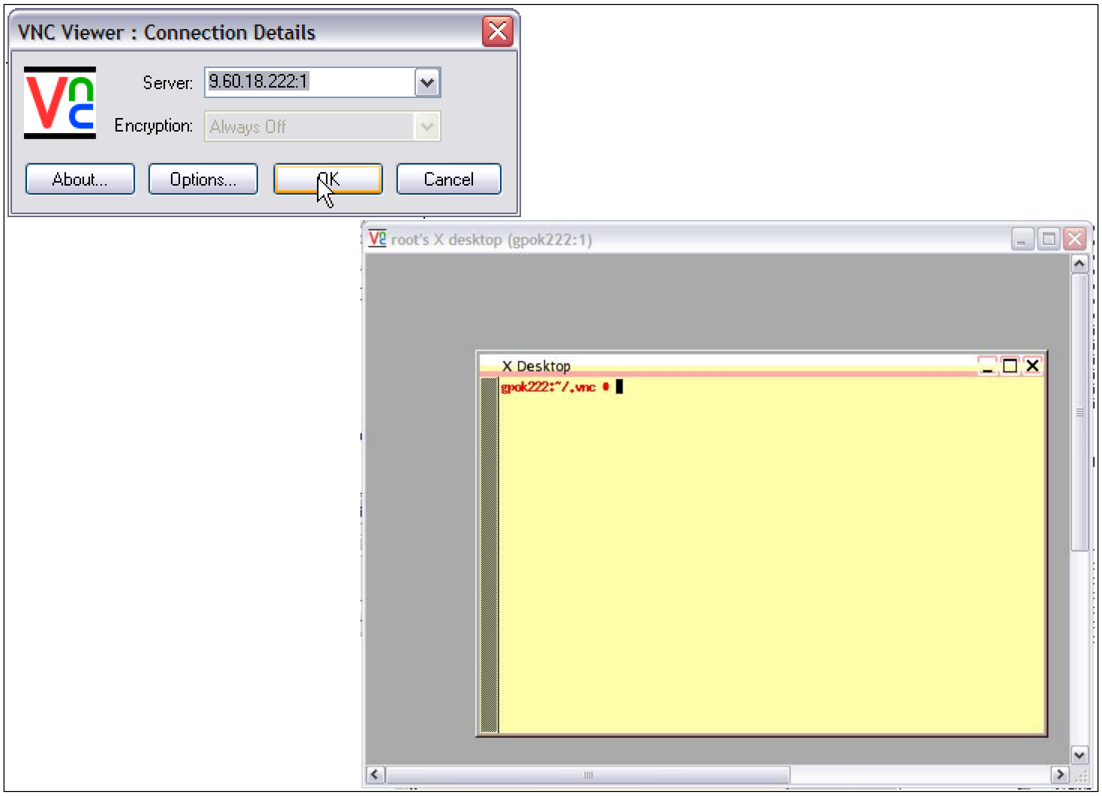


Figure 14-12 VNC session

Note that the VNC server will not be started across reboots. When you need a graphical environment, just start the **vncserver** manually.

- Stop the VNC server using the **-kill:1** argument:

```
# vncserver -kill :1
Killing Xvnc process ID 22601
```

The VNC server is now configured.

14.4.2 Prepare for Online Update

This step is recommended if you have Internet access.

Note: The system used for the writing of this book did not have Internet access, so the following steps have not been tested for SLES 11 SP2.

Before you can use *Online Update* (recommended), online sources have to be configured. This is done through the Novell Customer Center Configuration. To configure the Customer Center, a Web browser is needed. For this reason, a VNC server session must be started.

A graphical environment is recommended for this step. Start a VNC viewer. In a terminal session, start YaST with command **yast2** and choose the **Software => Online Update Configuration** selection.

14.4.3 Turn off unneeded services

There are a number of services which are started in a SLES 11 SP2 minimal system. Perform the following steps:

- View the services that are on in run level 3 with the following **chkconfig** command:

```
# chkconfig -l | grep 3:on
cron                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
dbus                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
earlysyslog               0:off  1:off  2:on   3:on   4:off  5:on   6:off
fbset                     0:off  1:on   2:on   3:on   4:off  5:on   6:off
haldaemon                 0:off  1:off  2:on   3:on   4:off  5:on   6:off
irq_balancer              0:off  1:on   2:on   3:on   4:off  5:on   6:off
network                   0:off  1:off  2:on   3:on   4:off  5:on   6:off
network-remotefs          0:off  1:off  2:on   3:on   4:off  5:on   6:off
nfs                       0:off  1:off  2:off  3:on   4:off  5:on   6:off
nscd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
postfix                   0:off  1:off  2:off  3:on   4:off  5:on   6:off
random                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
rpcbind                   0:off  1:off  2:off  3:on   4:off  5:on   6:off
smartd                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
splash                    0:off  1:on   2:on   3:on   4:off  5:on   6:off  S:on
splash_early              0:off  1:off  2:on   3:on   4:off  5:on   6:off
sshd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
syslog                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
```

- In order to keep the golden image as lean as possible in terms of CPU usage, some of these services can be turned off. Turn off the following services using the **chkconfig** command:

```
# chkconfig auditd off
# chkconfig fbset off
# chkconfig network-remotefs off
# chkconfig postfix off
# chkconfig splash off
# chkconfig splash_early off
# chkconfig smartd off
```

- 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 -l | grep 3:on
cron                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
dbus                      0:off  1:off  2:on   3:on   4:off  5:on   6:off
earlysyslog               0:off  1:off  2:on   3:on   4:off  5:on   6:off
haldaemon                 0:off  1:off  2:on   3:on   4:off  5:on   6:off
irq_balancer              0:off  1:on   2:on   3:on   4:off  5:on   6:off
network                   0:off  1:off  2:on   3:on   4:off  5:on   6:off
nfs                       0:off  1:off  2:off  3:on   4:off  5:on   6:off
nscd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
purge-kernels             0:off  1:off  2:off  3:on   4:off  5:on   6:off
random                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
rpcbind                   0:off  1:off  2:off  3:on   4:off  5:on   6:off
sshd                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
syslog                    0:off  1:off  2:on   3:on   4:off  5:on   6:off
```

This shows the services that will run in the default run level of 3.

14.4.4 Apply service - online update

If you have a SuSE Maintenance Web account then you can use it to retrieve the latest patches for SLES 11 SP2. Because many of these patches contain security and bug fixes, it is recommended that you apply the patches for the golden image so that it is up to date. Subsequently all the servers you clone after the golden image will also be up to date.

Section 14.4.2, "Prepare for Online Update" on page 229 must be completed before proceeding.

The system used for the writing of this book did not have Internet access, so this part was not tested at all. You may choose which packages to install and which not. It is highly recommended to download security patches.

When you finish here, don't reboot yet, you will reboot shortly after some boot configuration changes are made.

14.4.5 Configure /etc/inittab

If you did not change the default runlevel from 5 to 3 during the install process, this would be a good time to do so. By default, SLES 11 boots into run level 5, which is designed for a graphical environment. To keep the golden image as lean as possible, it is recommended that this be reset to run level 3. This is set in the `/etc/inittab` file with the variable `initdefault`.

Linux 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, it might be better that your system halts (shuts down).

Change **`shutdown -r`** to **`shutdown -h`**:

```
# cd /etc
# vi inittab
...
# The default runlevel is defined here
id:3:initdefault:
...
# what to do when CTRL-ALT-DEL is pressed
ca::ctrlaltdel:/sbin/shutdown -h -t 4 now
...
```

This change will be picked up when the system is rebooted.

14.4.6 Configure SSH keys

SSH sessions are typically authenticated using passwords typed in from the keyboard. With SSH *key-based authentication* 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 system must have the client's public key.
- The SSH client must send its private key.
- The keys must match cryptographically.

SSH key-based authentication can be set up from the Linux administration system (client) to the virtual servers. If the preceding requirements are met, then key-based authentication will work to the cloned virtual servers.

To set up key-based authentication from the Linux administration system to the golden image (and consequently from Linux administration system to all cloned images), perform steps that follow. Note that you will need SSH sessions on both the SLES 11 SP2 golden image (***gpok145***) and the SLES 11 SP2 Linux administration system (***gpok152***), so prefixes are used in the following examples.

- On the golden image, see if the directory `/root/.ssh/` exists:

```
gpok145: # cd /root
gpok145: # ls -d .ssh
ls: cannot access .ssh: No such file or directory
```

- In this example, it does not exist. Create the directory and set the permission bits to octal `700` with the **chmod** command:

```
gpok145: # mkdir .ssh
gpok145: # chmod 700 .ssh
```

- **Start an SSH session as root** to the Linux administration system. In this example the IP address is ***9.60.18.152***.
- Again, see if the directory `/root/.ssh/` exists:

```
gpok152: # cd /root
gpok152: # ls -d .ssh
ls: cannot access .ssh: No such file or directory
```

- Again, it does not exist. Create the directory and set the permission bits to octal `700` with the **chmod** command:

```
gpok152:~ # mkdir .ssh
gpok152:~ # chmod 700 .ssh
```

- Create a new key of type DSA in root's `.ssh/` directory:

```
gpok152:~ # cd .ssh
gpok152:~/.ssh # ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/root/.ssh/id_dsa): Enter
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
The key fingerprint is:
05:c0:f4:47:28:43:23:ba:d9:70:40:12:f4:13:06:15 root@gpok152
The key's randomart image is:
+--[ DSA 1024]----+
|+==Eo+=....      |
| .oo..+o.o       |
|  oo.  o. o      |
|   *.    o       |
|  o .   S        |
|                 |
|                 |
|                 |
|                 |
+-----------------+
```

- Copy the new public key to the `/root/.ssh/` directory that you just created to the golden image (***9.60.18.145*** in this example) as the file `/root/.ssh/authorized_keys`:

```
gpok152:~/.ssh # scp id_dsa.pub 9.60.18.145:/root/.ssh/authorized_keys
The authenticity of host '9.60.18.145 (9.60.18.145)' can't be established.
RSA key fingerprint is f1:6e:3e:70:f4:e3:c4:58:a5:e8:4a:0d:f6:b4:44:c8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '9.60.18.145' (RSA) to the list of known hosts.
Password: lnx4vm
id_dsa.pub                                    100%  602     0.6KB/s   00:00
```

- You should now be able to get an SSH session to the golden image without needing a password, because the Linux administration system is recognized as an authorized host:

```
gpok152: # ssh 9.60.18.145
The authenticity of host 'gpok145 (9.60.18.145)' can't be established.
RSA key fingerprint is bf:14:4f:27:da:44:59:1c:17:79:ac:81:31:be:86:24.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'gpok145' (RSA) to the list of known hosts.
Last login: Fri Nov 11 10:05:16 2011 from sig-9-76-44-138.mts.ibm.com
gpok145:~ # exit
```

  If you get an SSH session without having to supply a password, as with the above example, it shows that key-based authentication is working.

- Close the SSH session to the Linux administration machine:

```
# exit
```

14.4.7 Modify zipl.conf

Two changes are recommended to the default `zipl.conf` file:

1. The parameters `vmpoff=LOGOFF` and `vmhalt=LOGOFF` have the effect of logging the virtual machine off when Linux is shut down. This can be convenient for shutting the z/VM system down more efficiently and also for getting a refreshed 3270 emulator session.
2. The `timeout` value is modified to 3 seconds. When SLES 11 SP2 boots, the default time that is allowed to enter a menu command is 10 seconds. Because the default value is most commonly used, this 10 seconds delays the starting of Linux. It is backed up to three seconds.

To make these changes, perform the following steps:

- Make a backup copy of the file

```
# cd /etc
# cp zipl.conf zipl.conf.orig
```

- Add the strings **`vmpoff=LOGOFF`** and **`vmhalt=LOGOFF`** to the parameter line in the `[Linux]` section and set **`timeout = 3`** in the *menu* section:

```
# vi zipl.conf # Modified by YaST2. Last modification on Fri Nov 11 09:10:36 EST 2011
# Modified by YaST2. Last modification on Mon Dec 12 07:29:30 EST 2011
[defaultboot]
defaultmenu = menu

###Don't change this comment - YaST2 identifier: Original name: linux###
[SLES11_SP2]
    image = /boot/image-3.0.10-0.7-default
    target = /boot/zipl
    ramdisk = /boot/initrd-3.0.10-0.7-default,0x2000000
    parameters = "root=/dev/disk/by-path/ccw-0.0.0100-part1 vmpoff=LOGOFF vmhalt=LOGOFF 
hvc_iucv=8 TERM=dumb resume=/dev/disk/by-path/ccw-0.0.0300-part1"

###Don't change this comment - YaST2 identifier: Original name: failsafe###
[FailsafeV1]
    image = /boot/image-3.0.10-0.7-default
    target = /boot/zipl
    ramdisk = /boot/initrd-3.0.10-0.7-default,0x2000000
    parameters = "root=/dev/disk/by-path/ccw-0.0.0100-part1 hvc_iucv=8 TERM=dumb 
noresume x11failsafe"
```

```
:menu
    default = 1
    prompt = 1
    target = /boot/zipl
    timeout = 3
...
```

- Run the **zipl** command so the changes are written to the boot record:

```
# zipl
Using config file '/etc/zipl.conf'
Building bootmap in '/boot/zipl'
Building menu 'menu'
Adding #1: IPL section 'SLES11_SP1' (default)
Adding #2: IPL section 'FailsafeV1'
Adding #3: IPL section 'ipl'
Preparing boot device: dasda (0100).
Done.
```

These changes will be utilized the next time Linux is rebooted.

14.4.8 Clean up temporary files

Now is the best time to clean temporary files from golden image. If they stay there they will be copied to each clone and occupy space there. There are a number of directories which can be cleaned up and the history of the commands executed can be cleared.

- Create a script **cloneprep.sh** that can be run to prepare for cloning:

```
# cd /usr/local/sbin
# vi cloneprep.sh
#!/bin/bash
rm -fr /tmp/*
rm -fr /var/log/YaST2/*
history -c
```

  You may choose different steps to prepare for cloning.

- Make the script executable with the **chmod +x** command:

```
# chmod +x cloneprep.sh
```

- Call the script interactively:

```
# cloneprep.sh
```

The system should now be cleaned up for cloning.

14.4.9 Reboot the system and verifying changes

You are now done customizing the golden Linux image. Now **reboot** to test your changes:

```
# reboot
Broadcast message from root (pts/0) (Mon Nov 30 08:51:49 2009):

The system is going down for reboot NOW!
```

When the system comes back up you should verify the changes that you made.

- **SSH back into the golden image** and check a few settings.
- Use the **df -h** command to display your file systems:

```
# df -h
```

```
Filesystem                         Size  Used Avail Use% Mounted on
/dev/dasda1                        372M  149M  205M  43% /
devtmpfs                           121M  156K  121M   1% /dev
tmpfs                              121M     0  121M   0% /dev/shm
/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-user--lv    2.5G  1.2G  1.2G  51% /usr
/dev/mapper/system--vg-var--lv     504M   64M  416M  14% /var
```

- Confirm that both of your swap spaces are operational:

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdc1                             partition       259956  0       -1
/dev/dasdd1                             partition       519924  0       -2
```

  Device with higher priority will be used first.

- Shutdown your golden image from the SSH session (to clone Linux it should be shut down).

```
# shutdown -h now
```

Congratulations! You have now successfully installed the golden image. This image will normally be shut down.

15

Clone SLES 11 SP2

"The whole of science is nothing more than a refinement of everyday thinking."

--Albert Einstein

This chapter focuses on cloning the SLES 11 SP2 golden image. Examples are given for cloning manually and through the use of a script. The following steps are described:

- “Clone a virtual server manually” on page 237
- “Clone a virtual server automatically” on page 241

This sections assumes you have created two virtual machines with the same disk layout as the golden image. If you have not done that yet, see section 9.1, “Define two new virtual machines” on page 151.

15.1 Clone a virtual server manually

Before using the shell script **`clone.sh`** to clone a server, you may want to 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 (golden) Linux image is on the virtual machine `S112GOLD` on minidisks `100` and `101`.
- The target virtual machine, `LINUX153` in this example, has identically sized minidisks.
- The **`vmcp`** command is available to the Linux administration system, `LNXADMIN`, to issue z/VM CP commands.
- The z/VM **`FLASHCOPY`** command can be used but if you don’t have that support, the Linux **`dasdfmt`** and **`dd`** commands can also be used.

Given these assumptions, one set of steps that can be used to clone a system is as follows:

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 **dasdfmt** and **dd** commands.
4. Mount the newly copied root file system.
5. Modify the networking information on the target system.
6. Detach the target disks.
7. IPL the target system.
8. Modify the SSH keys on the target system.

The following sections describe these steps in detail.

Link the source disks read-only

Start an SSH session as root to the Linux administration system, LNXADMIN.

The source minidisks at virtual addresses 100 and 101 are on the golden image, S112GOLD. They are linked read-only as virtual addresses 1100 and 1101 using the RR parameter to the **CP LINK** command:

```
# vmcp link s112gold 100 1100 rr
# vmcp link s112gold 101 1101 rr
```

Link the target disk read-write

The target minidisks, also at addresses 100 and 101 are on the Linux administration system, LINUX153. They are linked multi-read (read-write if no other virtual machine has write access) using the MR parameter as virtual device 2100 and 2101 using the **CP LINK** command:

```
# vmcp link linux153 100 2100 mr
# vmcp link linux153 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 3337 TO 2100 0 3337
# vmcp flashcopy 1101 0 end to 2101 0 end
Command complete: FLASHCOPY 1101 0 3337 TO 2101 0 3337
```

Note: If you do not have **FLASHCOPY** support, you can use the Linux **dasdfmt** and **dd** commands. You must first enable the source and target disks using the **chccwdev -e** command, then determine the device name using the **lsdasd** command:

```
# chccwdev -e 1100
Setting device 0.0.1100 online
Done
# chccwdev -e 1101
Setting device 0.0.1101 online
Done
# chccwdev -e 2100
Setting device 0.0.2100 online
Done
# chccwdev -e 2101
Setting device 0.0.2101 online
Done
# lsdasd
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0300   active      dasdb     94:4    FBA   512    256MB     524288
0.0.0301   active      dasdc     94:8    FBA   512    512MB     1048576
0.0.1100   active      dasdd     94:12   ECKD  4096   2347MB    600840
0.0.1101   active      dasde     94:16   ECKD  4096   2347MB    600840
0.0.2100   active      dasdf     94:20   ECKD  4096   2347MB    600840
0.0.2101   active      dasdg     94:24   ECKD  4096   2347MB    600840
```

In this example the source devices are /dev/dasdd and /dev/dasde, and the target devices are /dev/dasdf and /dev/dasdg. Format the target disks using the **dasdfmt** command then copy it using the **dd** command using a block size of 4K (4096) bytes:

```
# dasdfmt -b 4096 -y -f /dev/dasdf
...
# dasdfmt -b 4096 -y -f /dev/dasdg
...
# dd if=/dev/dasdd of=/dev/dasdf bs=4096
600840+0 records in
600840+0 records out
# dd if=/dev/dasde of=/dev/dasdg bs=4096
600840+0 records in
600840+0 records out
```

The golden image should now be copied to the target disks. Disable the disks:

```
# chccwdev -d 1100
# chccwdev -d 1101
# chccwdev -d 2100
# chccwdev -d 2101
```

Mount the newly copied root file system

First enable target 100 (local 2100) disk for Linux usage with **chccwdev -e** command:

```
# chccwdev -e 2100
Setting device 0.0.2100 online
Done
```

Determine which device name is the target 100, or local 2100 disk:

```
# lsdasd | grep 2100
0.0.2100   active      dasdd     94:12   ECKD  4096   2347MB    600840
```

The first target disk in this example is the device node is `/dev/dasdd`. The first partition is the root file system, it will be `/dev/dasdd1`. Mount it over a the mount point `/mnt` and **cd** into it:

```
# mount /dev/dasdd1 /mnt
# cd /mnt
```

Observe that this appears to be a root file system:

```
# ls
bin   dev  home  lib64       media  opt   root  selinux  success  tmp  var
boot  etc  lib   lost+found  mnt    proc  sbin  srv      sys      usr
```

Modify networking information on the target system

In this example, the only two pieces of networking information that are modified are the IP address (from ***9.60.18.145*** to ***9.60.18.153***) and the host name (from ***gpok145*** to ***gpok153***).

The host name is changed in the file `/mnt/etc/HOSTNAME`:

```
# cd etc
# vi HOSTNAME
gpok153.endicott.ibm.com
```

The IP address is changed in the file `/mnt/etc/sysconfig/network/ifcfg-eth0`:

```
# cd sysconfig/network
# vi ifcfg-eth0
BOOTPROTO='static'
IPADDR='9.60.18.153/25'
BROADCAST='9.60.18.255'
STARTMODE='onboot'
NAME='OSA Express Network card (0.0.0600)'
```

Detach the target disks

Change directory to /, use the **sync** command to flush the disks and the **umount** command to unmount the modified root file system:

```
# cd /
# sync
# umount /mnt
```

Remove the target disk from Linux administration system:

```
# chccwdev -d 2100
Setting device 0.0.2100 offline
Done
```

Detach the source and target minidisks using the **CP DETACH** command:

```
# vmcp det 1100
DASD 1100 DETACHED
# vmcp det 1101
DASD 1101 DETACHED
# vmcp det 2100
DASD 2100 DETACHED
# vmcp det 2101
DASD 2101 DETACHED
```

The newly copied and modified system disks have now been detached.

IPL the target system

You should now be ready to IPL the manually cloned system.

Start a 3270 session to *LINUX153* and IPL from minidisk 100:

```
00: z/VM Version 6 Release 2.0, Service Level 0000 (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 12:34:43 EST FRIDAY 11/11/11
00: Command complete
00: NIC 0600 is created; devices 0600-0602 defined
00: NIC 0600 is connected to VSWITCH SYSTEM VSW1
00: Command complete
00: NIC 0700 is created; devices 0700-0702 defined
00: NIC 0700 is connected to VSWITCH SYSTEM VSW2
00: STORAGE = 256M MAX = 1G INC = 2M STANDBY = 768M RESERVED = 0
00: Storage cleared - system reset.
z/VM V6.2.0    2011-07-19 16:53

DMSACP723I A (191) R/O
DMSACP113S C(592) not attached or invalid device address
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
y
00: zIPL v1.15.0-0.79.3 interactive boot menu
00:
00:  0. default (SLES11_SP2)
00:
00:  1. SLES11_SP2
00:  2. FailsafeV1
00:  3. ipl
00:
00: Note: VM users please use '#cp vi vmsg <number> <kernel-parameters>'
00:
00: Please choose (default will boot in 3 seconds):
00: Booting default (SLES11_SP2)...
...
```

Watch for error messages on the console. Your new system should come up cleanly using the modified IP address and host name.

Congratulations! 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 can learn how to do it automatically. You will use the `LINUX153` virtual machine again. To clone, the target virtual machine 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 you can crash the `LINUX153` virtual machine using the **`LOGOFF`** command:

```
==> #cp log
```

15.2 Clone a virtual server automatically

Now that you have cloned a server manually and better understand the steps, you can use the **`clone.sh`** script to clone automatically. To do so, perform the following steps:

- Create a SLES 11 parameter file, `LINUX153 PARM-S11` on the `LNXMAINT 192` disk. See section 14.2, “Create the S112GOLD parameter file” on page 212 for a reference.

- Start an SSH session as root to the Linux administration system. The **clone.sh** script should be in your PATH in the directory /usr/local/sbin/. You can verify this with the **which** command:

```
# which clone.sh
/usr/local/sbin/clone.sh
```

- Clone the golden image. Both the source and target virtual machines must be logged off. The script reads the parameter file on the LNXMAINT 192 disk (the Linux administration system's 191 disk) to obtain information necessary to give the new Linux virtual server an identity. It should take less than a minute to clone with **FLASHCOPY** support and 10-20 minutes without it. Following is an example of cloning to the LINUX153 virtual machine with **FLASHCOPY** support. The output is divided into sections.

```
# clone.sh from s112gold to linux153
Checking that S112GOLD exists and is not logged on ...
Invoking CP command: QUERY S112GOLD
HCPCQU045E S112GOLD not logged on
Error: non-zero CP response for command 'QUERY S112GOLD': #45
Checking that LINUX153 exists and is not logged on ...
Invoking CP command: QUERY LINUX153
HCPCQU045E LINUX153 not logged on
Error: non-zero CP response for command 'QUERY LINUX153': #45
Setting device 0.0.0191 offline
Done
Setting device 0.0.0191 online
Done
S112GOLD PARM-S11 D1 F        80          9          1 11/11/2011  8:25:23
LINUX153 PARM-S11 D1 F        80          9          1 11/11/2011 12:53:23

WARNING!!: Mindisks 100 and 101 will be copied to LINUX153
New host name will be:  gpok153.endicott.ibm.com
New IP address will be: 9.60.18.153
Other network data is retrieved from LINUX153 PARM-S11 on 191 disk
Are you sure you want to overwrite these disks (y/n): y
```

The script verifies that the virtual machines S112GOLD and LINUX153 exist and are logged off. The 191 disk is deactivated and reactivated to pick up any changes that may have occurred. It is then verified that the two parameter files, S112GOLD PARM-S11 and LINUX153 PARM-S11 exist. You are then asked to confirm that the disks on the target system can be written to. Answer yes with **y**.

The script continues:

```
Linking source and target 100 disks ...
Invoking CP command: detach 1100
HCPDTV040E Device 1100 does not exist
Error: non-zero CP response for command 'DETACH 1100': #40
Invoking CP command: link s112gold 100 1100 rr

Invoking CP command: detach 2100
HCPDTV040E Device 2100 does not exist
Error: non-zero CP response for command 'DETACH 2100': #40
Invoking CP command: link LINUX153 100 2100 mr

Copying 100 disks ...

FLASHCOPYing 1100 to 2100 ...
Invoking CP command: FLASHCOPY 1100 0 end to 2100 0 end
Command complete: FLASHCOPY 1100 0 3337 TO 2100 0 3337
Take 1100 Offline....
Device is already offline
```

```
Done
Invoking CP command: det 1100
DASD 1100 DETACHED

---------------------------------------
Linking source and target 101 disks ...
Invoking CP command: detach 1101
HCPDTV040E Device 1101 does not exist
Error: non-zero CP response for command 'DETACH 1101': #40
Invoking CP command: link s112gold 101 1101 rr

Invoking CP command: detach 2101
HCPDTV040E Device 2101 does not exist
Error: non-zero CP response for command 'DETACH 2101': #40
Invoking CP command: link LINUX153 101 2101 mr

Copying 101 disks ...

FLASHCOPYing 1101 to 2101 ...
Invoking CP command: FLASHCOPY 1101 0 end to 2101 0 end
Command complete: FLASHCOPY 1101 0 3337 TO 2101 0 3337
Taking 1101 Offline...
Device is already offline
Done
Invoking CP command: det 1101
DASD 1101 DETACHED
Taking 2101 Offline...
Device is already offline
Done
Invoking CP command: det 2101
DASD 2101 DETACHED
Mounting newly cloned image over /mnt/targetLinux ...
Setting device 0.0.2100 online
Done
```

In the section above, the script copies the source 100 and 101 disks to the target virtual machine using the **FLASHCOPY** command. If **FLASHCOPY** is not supported or fails for some other reason the script falls back to the **dasdfmt** and **dd** commands to copy the disks.

```
Mounting /dev/dasdi1 over /mnt/targetLinux ...
Modifying cloned image under /mnt/targetLinux ...
Removing SSH keys
Removing 9.60.18.224 from known_hosts file
Setting device 0.0.2100 offline
Done
Invoking CP command: DETACH 2100
DASD 2100 DETACHED
```

In the section above the script mounts the newly copied root file system over a mount point, /mnt/targetLinux/. The networking information is modified and the SSH keys are regenerated. Then the disk is detached.

```
Invoking CP command: XAUTOLOG LINUX153
Command accepted
Successfully cloned s112gold to LINUX153
You should be able to ping 9.60.18.224 within one minute
```

In the final section, the target virtual machine is logged on using **XAUTOLOG**. Because the PROFILE EXEC detects that the ID is logged on in a disconnected mode, Linux is IPLed from minidisk 100. The new system should be on the network in about 30-45 seconds.

Note: If the `clone.sh` script fails you can add the `-v` flag for some more diagnostics. Also, check that:

- The target virtual machine has been granted access to the virtual switch.
- The parameter file is copied and set correctly on the `LNXMAINT 192` disk.
- The target `100` and `101` minidisks are sized identically.

Congratulations - you have successfully cloned a SLES 11 SP2 system both manually and automatically. The next section describes how to create appliances from these *clones*.

16

Create SLES 11 SP2 appliances

"Common sense is the collection of prejudices acquired by age eighteen."

--Albert Einstein

This chapter describes how to clone and customize the following Linux virtual servers:

These Linux virtual servers can be thought of as *virtual appliances* once they have been cloned and *personalities* have been added to them.

16.1 Creating a virtual Web server

The example in this section uses the `LINUX153` virtual machine to create a virtual Web server. You should have a vanilla virtual server cloned to the virtual machine `LINUX153` as described in Chapter 15, “Clone SLES 11 SP2” on page 237.

16.1.1 Installing Apache RPMs

This section describes how to install the Apache Web Server RPMs. To do so, perform the following steps:

- **SSH** as root to the Linux system running on `LINUX153`.
- Install the following Apache RPMs using the **`zypper install`** command:

```
# zypper install apache2-prefork apache2 apache2-doc apache2-example-pages
zypper install apache2-prefork apache2 apache2-doc apache2-example-pages
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW packages are going to be installed:
```



```
  apache2 apache2-doc apache2-example-pages apache2-prefork apache2-utils
  libapr-util1 libapr1

The following recommended package was automatically selected:
  apache2-prefork

7 new packages to install.
Overall download size: 3.1 MiB. After the operation, additional 14.1 MiB will
be used.
Continue? [y/n/?] (y): y
...
```

- Confirm the RPMs have been added using the **rpm -qa** command:

```
# rpm -qa | grep apache
apache2-prefork-2.2.12-1.24.1
apache2-example-pages-2.2.12-1.24.1
apache2-2.2.12-1.24.1
apache2-doc-2.2.12-1.24.1
apache2-utils-2.2.12-1.24.1
```

You now have the Apache RPMs installed.

16.1.2 Testing Apache

Start the Apache Web server to verify it is installed successfully.

- Start the apache server using the **service** command and set it to start at boot time using the **chkconfig** command:

```
# service apache2 start
Starting httpd2 (prefork)                                              done
# chkconfig apache2 on
```

- To verify that Apache is installed correctly, after it's been started, point a Web browser to the server and see the Apache test page. In your Web browser, put in the host name or IP address of your Web server as the URL. In this example the virtual server running on `LINUX153` has a DNS name of `gpok153.endicott.ibm.com`:

```
http://gpok153.endicott.ibm.com
```

- You should see a test page with two words **It works!**

If you get an error in starting Apache, look in the log file `/var/log/apache2/error-log` for clues. If Apache started successfully but you can't reach the test page from a browser, try accessing it using the IP address rather than the DNS name.

16.1.3 Populating your Web site

You can begin to put your Web pages in the directory `/srv/www/htdocs/`, which is the default Web root. For security and customization purposes, you might want to change the default Web root to point to another directory. The easiest way to do this is to copy `/etc/apache2/default-server.conf` to your own configuration file, i.e. `/etc/apache2/my-server.conf`. Make the changes in `/etc/apache2/my-server.conf`, and then edit `/etc/apache2/httpd.conf` to use `my-server.conf`.

16.1.4 Apache resources

The following Web sites contain additional information on 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

16.2 Create 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 clone Linux and configure 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.

Then you may want to configure the golden image so that it is pointing to this virtual server. If you do so, all Linux images that are cloned will be able to use this virtual LDAP server.

The steps in this section are as follow:

- “Clone a Linux” on page 247
- “Configure the LDAP server” on page 248
- “Add an LDAP user” on page 250
- “Set another virtual server to use the LDAP server” on page 251

16.2.1 Clone a Linux

From a root session on the Linux administration system, clone from the golden image (S112GOLD virtual machine) to LINUX157 using the **clone.sh** script:

```
# clone.sh from s112gold to linux157
Checking that S112GOLD exists and is not logged on ...
Invoking CP command: QUERY S112GOLD
HCPCQU045E S112GOLD not logged on
Error: non-zero CP response for command 'QUERY S112GOLD': #45
Checking that LINUX157 exists and is not logged on ...
Invoking CP command: QUERY LINUX157
HCPCQU045E LINUX157 not logged on
Error: non-zero CP response for command 'QUERY LINUX157': #45
Setting device 0.0.0191 offline
Done
Setting device 0.0.0191 online
Done
S112GOLD PARM-S11 D1 V        74         11          1 12/12/2011  6:24:17
LINUX157 PARM-S11 D1 V        74         11          1 12/14/2011 12:07:29

WARNING!!: Mindisks 100 and 101 will be copied to LINUX157
New host name will be:  gpok157.endicott.ibm.com
New IP address will be: 9.60.18.157
Other network data is retrieved from LINUX157 PARM-S11 on 191 disk
Are you sure you want to overwrite these disks (y/n): y
...
```

The **clone.sh** script should create a new virtual server.

16.2.2 Configure the LDAP server

To configure the OpenLDAP server, the `yast` tool is recommended.

- **Start an SSH session as root** to the new server.
- Invoke the `yast` command. The `YaST Control Center` should appear:
- Use the down arrow key to move to *Network Services* on the left side. Use the Tab or right arrow key to move to the right side and select *LDAP Server*. Press **Enter**.

```
# yast

 +--------------------------------------------------------------------------+
 |                          YaST2 Control Center                            |
 +--------------------------------------------------------------------------+

 +--------------------+ +---------------------------------------------------+
 |Software            | |Hostnames                                          |
 |Hardware            | |Kerberos Client                                    |
 |System              | |Kerberos Server                                    +
 |Network Devices     | |LDAP Browser                                       |
 |Network Services    | |LDAP Client                                        |
 |Security and Users  | |LDAP Server                                        +
 |Support             | |Mail Transfer Agent                                |
 |Miscellaneous       | |NFS Client                                         |
 |                    | |NFS Server                                         |
 |                    | |NIS Client                                         |
 |                    | |NIS Server                                         |
 +--------------------+ +---------------------------------------------------+
```

- You should see a pop-up window with the following message. Press **Enter** to accept the default of **Install** and some RPMs will be installed:

```
These packages need to be installed: openldap2
```

- The *LDAP Server Configuration* panel will appear. Accept the default of **Yes** by using the tab key to **Next** and pressing **Enter**.

```
                    General Settings
             +Start LDAP Server-------------------------------+
             |(x) Yes                                         |
             |( ) No                                          |
             |[ ] Register at an SLP Daemon                   |
             +------------------------------------------------+

             +Firewall Settings-------------------------------+
             |[ ] Open Port in Firewall  [Firewall Details...]|
             |Firewall is disabled                            |
             +------------------------------------------------+
```

- A second *LDAP Server Configuration* panel will appear. Accept the default of **Standalone server** by using the tab key to **Next** and pressing **Enter**.

```
                          Server type
 +---------------------------------------------------------------------------+
 |                                                                           |
 |(x) Stand-alone server                                                     |
 |                                                                           |
 |( ) Master server in a replication setup                                   |
 |                                                                           |
```

```
¦( ) Replica (slave) server.                                             ¦
¦    All data, including configuration, is replicated from a remote server.¦
+------------------------------------------------------------------------+
```

- In the *TLS settings* panel, accept all defaults of using TLS and press **Next**. This will ensure that LDAP communications are encrypted.
- The *Basic Database Settings* panel will appear. The *Base DN* (in this example `dc=endicott,dc=ibm,dc=com`) should be correct for your DNS domain. Set the LDAP administrator password (twice) and press **Next**.

```
New Database
                          Basic Database Settings
         Database Type
         hdb#########â
         Base DN
         dc=endicott,dc=ibm,dc=com################################

         Administrator DN
         cn=Administrator###################### [x] Append Base DN

         LDAP Administrator Password
         ******###################################################
         Validate Password
         ******###################################################

         Database Directory
         /var/lib/ldap################################ [Browse...]
         [x] Use this database as the default for OpenLDAP clients
```

- The *LDAP Server Configuration Summary* panel should appear now., this time with one database listed identified by the Base DN (`dc=endicott,dc=ibm,dc=com` in this example). Select **Finish** by pressing **Enter**.

```
LDAP Server Configuration Summary
+------------------------------------------------------------------------+
|Startup Configuration                                                   |
|                                                                        |
|Start LDAP Server: Yes                                                  |
|                                                                        |
|Register at SLP Service: No                                             |
|                                                                        |
|Create initial Database with the following Parameters                   |
|                                                                        |
|Database Suffix: dc=endicott,dc=ibm,dc=com                              |
|                                                                        |
|Administrator DN: cn=Administrator,dc=endicott,dc=ibm,dc=com            |
|                                                                        |
+------------------------------------------------------------------------+
```

- The panel *Saving LDAP Server Configuration* should appear. The database will be created and the LDAP server configured.
- You should be returned to the *YaST Control Center*. Move the cursor to **Quit** and press **Enter**.
- Verify that the LDAP server is running with the **`service`** command and that it is set to start in run levels 3 and 5 with the **`chkconfig`** command

```
# service ldap status
```

```
Checking for service ldap:                                              running
# chkconfig --list ldap
ldap                      0:off  1:off  2:off  3:on   4:off  5:on   6:off
```

You have now cloned a new virtual server and configured it to run OpenLDAP.

16.2.3 Add an LDAP user

When the golden image was installed, it was recommended that a non-root user be added. In this example, it was named `mikemac`.

- Verify that this user exists with the **id** command and see that there is an entry in the `/etc/passwd` file with the **grep** command:

```
# id mikemac
uid=1000(mikemac) gid=100(users) groups=16(dialout),33(video),100(users)
# grep mikemac /etc/passwd
mikemac:x:1000:100::/home/mikemac:/bin/bash
```

- Delete this local user using the **userdel** command so it can be added to LDAP later.

```
# userdel mikemac
no crontab for mikemac
# id mikemac
id: mikemac: No such user
```

- An LDIF (LDAP Interchange Format) file is created to add an organizational unit named **People** and a user named **mikemac**. Create a similar file for your system's values.

```
# cd /var/lib/ldap
# vi initial.ldif // create the input file ...
dn: ou=People,dc=endicott,dc=ibm,dc=com
ou: People
objectClass: top
objectClass: organizationalUnit

dn: uid=mikemac,ou=People,dc=endicott,dc=ibm,dc=com
uid: mikemac
cn: mikemac
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
loginShell: /bin/bash
uidNumber: 501
gidNumber: 100
homeDirectory: /home/mikemac
```

- Add the contents of the LDIF file to the LDAP server with the **ldapadd** command (the line wraps, but it is one command):

```
# ldapadd -x -h localhost -D "cn=Administrator,dc=endicott,dc=ibm,dc=com" -f
initial.ldif -W
Enter LDAP Password:
adding new entry "ou=People,dc=endicott,dc=ibm,dc=com"

adding new entry "uid=mikemac,ou=People,dc=endicott,dc=ibm,dc=com"
```

- Search for the new user just added with the **ldapsearch** command:

```
# ldapsearch -x uid=mikemac
# extended LDIF
#
# LDAPv3
```

```
# base <> with scope subtree
# filter: uid=mikemac
# requesting: ALL
#

# mikemac, People, endicott.ibm.com
dn: uid=mikemac,ou=People,dc=endicott,dc=ibm,dc=com
uid: mikemac
cn: mikemac
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
loginShell: /bin/bash
uidNumber: 501
gidNumber: 100
homeDirectory: /home/mikemac

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

This shows that the user exists in the LDAP database.

- Set the password with the **ldappasswd** command. You will need to provide a new password for the new user and you will also need to provide the LDAP administrator password:

```
# ldappasswd -x -D "cn=Administrator,dc=endicott,dc=ibm,dc=com" -W -S
"uid=mikemac,ou=People,dc=endicott,dc=ibm,dc=com"
New password:
Re-enter new password:
Enter LDAP Password:
Result: Success (0)
```

You have now deleted a local user, added a new LDAP user using an LDIF file, and have set the new LDAP user's password.

16.2.4 Set another virtual server to use the LDAP server

Now that you have a virtual LDAP server, you may want to point another virtual server to it so you will have a centralized user database. If you have been following along in this book you should have created a Web server running on the `LINUX153` virtual machine. To point it to an LDAP server is fairly easy. In this section you will perform the following steps:

- "Test that the LDAP client is not working" on page 251
- "Use YaST to modify the LDAP authentication client" on page 252
- "Test the LDAP client" on page 253

Test that the LDAP client is not working

Before you start, try a couple of commands to show that LDAP is *not* working. **Get an SSH session to the virtual Web server** running on the virtual machine `LINUX153`.

Search for the LDAP user that you added earlier to the virtual LDAP server. In this example it is `mikemac`.

```
# ldapsearch -x uid=mikemac
```

```
ldap_sasl_bind(SIMPLE): Can't contact LDAP server (-1)
```

The **ldapsearch** command cannot resolve the LDAP user because it cannot contact the LDAP server.

Delete the non-root user (`mikemac` in this example) from the local file system with the **userdel** command:

```
# userdel mikemac
no crontab for mikemac
```

Use YaST to modify the LDAP authentication client

The **yast** system administration interface can be used to configure the LDAP authentication client.

- Invoke the yast command. The `YaST Control Center` should appear.
- Select **Network Services** on the left side and **LDAP Client** on the right. Press **Enter**:

```
# yast
  +---------------------------------------------------------------------------+
  ¦                          YaST Control Center                              ¦
  +---------------------------------------------------------------------------+

  +--------------------+ +--------------------------------------------------+
  ¦Software            ¦ ¦DNS and Hostname                                  ¦
  ¦Hardware            ¦ ¦Hostnames                                         ¦
  ¦System              ¦ ¦Kerberos Client                                   ¦
  ¦Network Devices     ¦ ¦LDAP Browser                                      ¦
  ¦Network Services    ¦ ¦LDAP Client                                       ¦
  ¦Security and Users  ¦ ¦LDAP Server                                       ¦
  ¦Miscellaneous       ¦ ¦Mail Transfer Agent                               ¦
  ¦                    ¦ ¦NFS Client                                        ¦
  ...
```

- On the *LDAP Client Configuration panel*, perform the following steps:
  a. Use the Tab key to move to **Use LDAP** and press the **space bar** to select that choice.
  b. Move to the *Addresses of LDAP Servers* field and enter the IP address (or DNS name) of your LDAP server. You can either enter LDAP base DN manually or press **Fetch DN** and then **OK** in result window. This way you can make sure LDAP server is accessible.
  c. Deselect LDAP TLS/SSL:

```
+User Authentication----------------------------------------------------+
¦ ( ) Do Not Use LDAP                                                   ¦
¦ (x) Use LDAP                                                          ¦
¦ ( ) Use LDAP but Disable Logins                                       ¦
¦ [ ] Use System Security Services Daemon (SSSD)                        ¦
+-----------------------------------------------------------------------+
+LDAP Client------------------------------------------------------------+
¦ Addresses of LDAP Servers                                             ¦
¦ 9.60.18.157¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦[Find] ¦
¦ LDAP Base DN                                                          ¦
¦ dc=endicott,dc=ibm,dc=com¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦¦[Fetch DN] ¦
+-----------------------------------------------------------------------+
+Secure Connection------------------------------------------------------+
¦ [ ] LDAP TLS/SSL                         [Download CA Certificate] ¦
+-----------------------------------------------------------------------+

[ ] Start Automounter
[ ] Create Home Directory on Login
```

```
[Advanced Configuration...]
```

- Use the Tab key to move to **OK** and press **Enter**. You should get the following prompt. Press **Enter** to continue:

```
These packages need to be installed:
 pam_ldap
 nss_ldap
 pam_ldap-32bit
 nss_ldap-32bit
```

- Accept **OK** if you get a warning window. Your changes will be saved.
- At the main *YaST2 Control Center*, press the **Quit** button on the main window to quit YaST.

Your Web server virtual Linux should now also be using OpenLDAP for user and group ID resolution and authentication.

Test the LDAP client

To test the LDAP client, perform the following steps:

- Try the **id** command against the new LDAP user:

```
# id mikemac
uid=501(mikemac) gid=100(users) groups=100(users)
```

Note that the UID is `501` in this example (from the LDIF file), not the value `1000` (from the Linux installation).

- Try the **ldapsearch** command again:

```
# ldapsearch -x uid=mikemac
# extended LDIF
#
# LDAPv3
# base <dc=endicott,dc=ibm,dc=com> (default) with scope subtree
# filter: uid=mikemac
# requesting: ALL
#

# mikemac, People, endicott.ibm.com
dn: uid=mikemac,ou=People,dc=endicott,dc=ibm,dc=com
uid: mikemac
cn: mikemac
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
loginShell: /bin/bash
uidNumber: 501
gidNumber: 100
homeDirectory: /home/mikemac

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

- Start an SSH session to the virtual Web server using the LDAP user. You should be able to successfully start a session.

You may also want to set the golden image to authenticate with the LDAP server. In this fashion, all virtual servers cloned after that will be able to utilize a centralized authentication server.

16.3 Create a virtual file and print server

Samba allows Windows clients to map Linux file systems as shared drives. Samba can also act as a middle-man between Windows clients and a Linux print server. The recommended Linux print server is CUPS - the Common UNIX Printing System. This section does not describe the configuration of CUPS but it does describe how the necessary RPMs are installed.

The steps in this section are as follow:

- “Install Samba and CUPS”
- “Configure Samba configuration file” on page 255
- “Add a Samba user” on page 255
- “Set Samba to start at boot time” on page 256
- “Test the configuration” on page 256

16.3.1 Install Samba and CUPS

To install Samba and CUPS, perform the following steps:

- Start an SSH session to a virtual server as root. In this example the virtual machine `LINUX153` is used again
- Install necessary RPMs with the following **`zypper install`** command:

```
# zypper install samba yast2-samba-server samba-doc samba-winbind cups cups-drivers
Loading repository data...
Reading installed packages...
'yast2-samba-server' is already installed.
No update candidate for 'yast2-samba-server-2.17.13-0.4.72.noarch'. The highest
available version is already installed.
Resolving package dependencies...

The following NEW packages are going to be installed:
  a2ps cups cups-drivers fam foomatic-filters ghostscript-fonts-other
  ghostscript-fonts-std ghostscript-library ghostscript-omni libavahi-client3
  libavahi-common3 libdns_sd libgimpprint libgmodule-2_0-0 libiniparser0
  libiniparser0-32bit libldb1 libpoppler5 libpython2_6-1_0-32bit
  libtalloc2-32bit libtdb1 libtdb1-32bit libtevent0 libwbclient0
  libwbclient0-32bit poppler-data poppler-tools samba samba-32bit samba-client
  samba-client-32bit samba-doc samba-krb-printing samba-winbind
  samba-winbind-32bit wdiff

The following recommended packages were automatically selected:
  ghostscript-fonts-other ghostscript-omni poppler-data samba-krb-printing

36 new packages to install.
Overall download size: 68.8 MiB. After the operation, additional 299.6 MiB will
be used.
Continue? [y/n/?] (y): y
...
```

- Confirm that the RPMs were added:

```
# rpm -qa | egrep "samba|cups"
```

```
yast2-samba-client-2.17.21-0.5.130
yast2-samba-server-2.17.13-0.4.72
samba-client-32bit-3.6.1-0.14.1
samba-winbind-3.6.1-0.14.1
cups-1.3.9-8.44.1
cups-client-1.3.9-8.44.1
samba-winbind-32bit-3.6.1-0.14.1
samba-32bit-3.6.1-0.14.1
samba-krb-printing-3.6.1-0.14.1
cups-drivers-1.3.9-2.31
cups-libs-32bit-1.3.9-8.44.1
cups-libs-1.3.9-8.44.1
samba-doc-3.6.1-0.14.1
samba-client-3.6.1-0.14.1
samba-3.6.1-0.14.1
```

The Samba and CUPS RPMs are now installed.

16.3.2 Configure 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/` as it has excellent 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 = SLES 11 SP2 on System z documentation
        path = /usr/share/doc/
```

This will cause an SMB share named **sharedoc** consisting of the contents of `/usr/share/doc` to be created when Samba is started.

16.3.3 Add a Samba user

The default method that Samba uses to determines users' credentials is to look in the `/etc/samba/smbpasswd` file. That user must first exist in the Linux file system (`/etc/passwd`, `/etc/shadow`, etc).

The following example shows adding the user `mikemac` to the `smbpasswd` file. Create a new Samba user with the **smbpasswd -a** command:.

```
# smbpasswd -a mikemac
New SMB password:
Retype new SMB password:
Added user mikemac.
```

This method of maintaining Samba users, groups and passwords is good 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 16.2.4, "Set another virtual server to use the LDAP server" on page 251, because the Samba schema must first be added to LDAP.

16.3.4 Set Samba to start at boot time

Samba consists of two daemons **nmbd** and **smbd**. To set Samba to start at Linux boot time, perform the following steps:

- Start Samba in the current session with the **service** command for both the **nmb** and **smb** daemons:

```
# service nmb start
Starting Samba NMB daemon                                        done
# service smb start
Starting Samba SMB daemon                                        done
```

- Use the **chkconfig** command to so these daemons will start at boot time:

```
# chkconfig nmb on
# chkconfig smb on
```

Samba should now be running and configured to start at boot time.

16.3.5 Test the configuration

To test the Samba configuration, perform the following steps:

- Verify that the Samba daemons are running with the **service** command:

```
# service nmb status
Checking for Samba NMB daemon                                    running
# service smb status
Checking for Samba SMB daemon                                    running
```

- 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 16-1 on page 257. In this example the UNC is ***9.60.18.153*\sharedoc**.
  - You may need to click **Connect using different user name**, if the sample virtual machine and password are different on your desktop computer from the values you set on the Samba server.
  - 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 the figure.

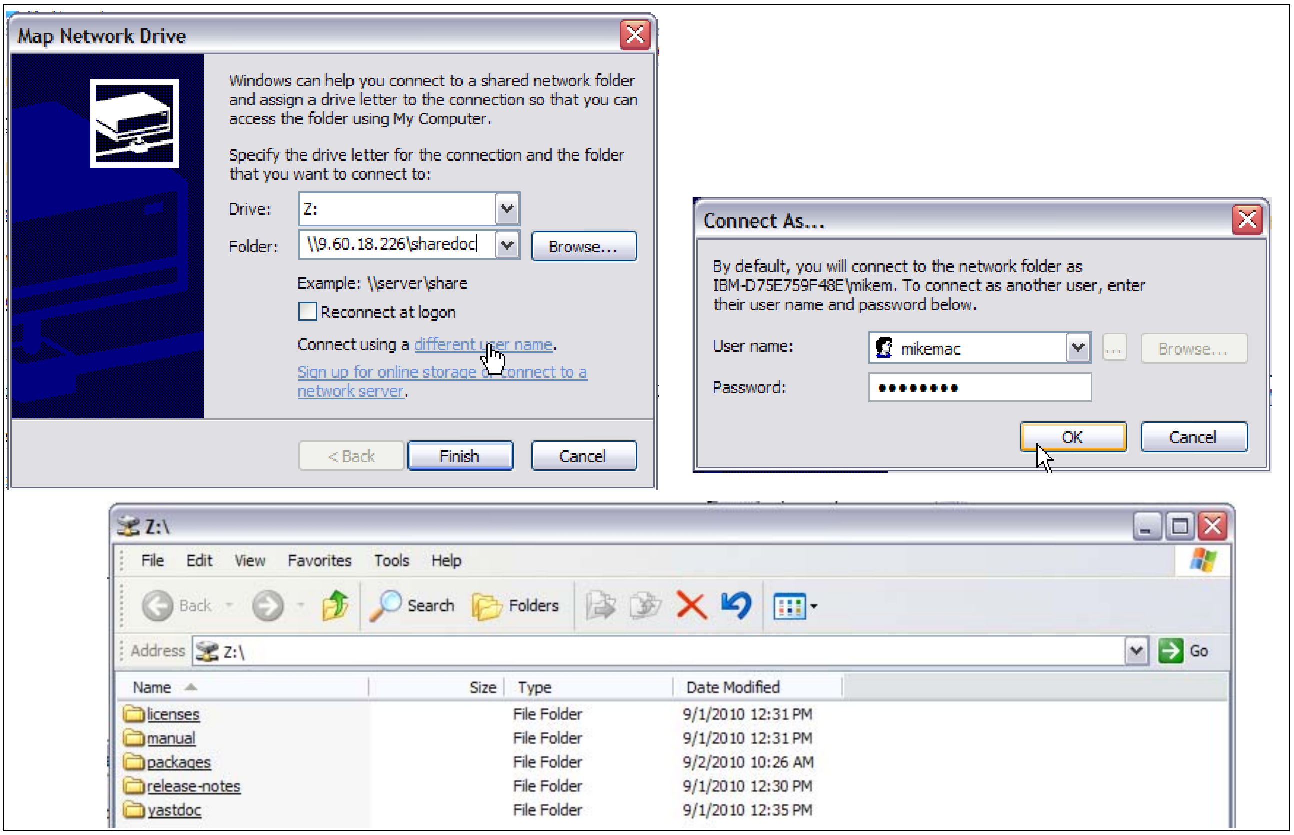


Figure 16-1 Mapping a network drive to a Samba share

You should now have Samba configured and running with one new share available.

- Test getting a Samba share from a Windows DOS session:
  - Use the **NET USE** command to access a Samba share:

```
c:\> net use * \\9.60.18.153\sharedoc
The password is invalid for \\9.60.18.153\sharedoc.

Enter the user name for '9.60.18.153': mikemac
Enter the password for 9.60.18.153:
Drive Z: is now connected to \\9.60.18.153\sharedoc.

The command completed successfully.
```

  - Query the Samba server with the **NET VIEW** command:

```
c:\> net view \\9.60.18.153
Shared resources at \\9.60.18.226

Samba 3.4.3-1.17.2-2359-SUSE-CODE11

Share name   Type   Used as   Comment

-------------------------------------------------------------------------------
groups       Disk             All groups
mikemac      Disk             Home Directories
profiles     Disk             Network Profiles Service
sharedoc     Disk   Z:        SLES 11 SP2 on System z documentation
users        Disk             All users
The command completed successfully.
```

16.4 Create an application development appliance

Most Linux distributions come with a robust set of application development tools, making Linux one of the most versatile development systems. These basic tools are ideal for projects of any size.

There are three main areas of development in Linux:

- Linux kernel development (C) for the Linux operating system itself, such as subsystems, device drivers, memory management.
- Application development (C/C++ and Java) for software to be used on Linux.
- Web development for applications to be run on the Web, such as stock trade applications or E-mail applications.

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 are software available on Linux to help form a development system for developers to create integrated applications. 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. Other times, it is just as useful to know about Linux development tools when you want to build an application from source code downloaded from `www.sourceforge.net`.

To create an application development server, perform the following steps:

- **Start an SSH session as root** to the new virtual server.
- You can install *all* the application development tools described in this section with the following command:

```
# zypper install python perl tcl php gcc gdb make java-1_6_0-ibm
...
18 new packages to install.
Overall download size: 66.9 MiB. After the operation, additional 156.6 MiB will
be used.
Continue? [y/n/?] (y): y
...
```

If you want to install only certain tools, each specific RPM or group of RPMs is described in the sections that follow.

16.4.1 Scripting Languages

Scripts are good for quickly automating a process or writing your own commands. They are also used for being the backbone of robust applications. There are numerous scripting languages used in Linux application development, here are overviews of the most popular and general ones, obtained from their package descriptions.

- **Python**: Python is an interpreted, object-oriented programming language, and is often compared to Tcl, Perl, Scheme, or Java. You can find an overview of Python in the documentation and tutorials included in the python-doc (HTML) or python-doc-pdf (PDF) packages. To install the python interpreter, execute the command:

```
# zypper install python
...
```

- **Perl**: Practical Extraction and Report Language. Perl is optimized for scanning arbitrary text files, extracting information from those text files, and printing reports based on that information. It is also good for many system management tasks. Perl is intended to be practical (easy to use, efficient, and complete) rather than beautiful (tiny, elegant, and minimal). To install perl, execute the command:

```
# zypper install perl
...
```

- **tcl**: the “Tool Command Language”, is a very simple programming language. Tcl provides basic language features such as variables, procedures, and control. Another very popular extension is Expect which can be used to automate console-based interactive applications. To install tcl, execute the command

```
# zypper install tcl
...
```

- **PHP**: PHP (recursive acronym for “PHP: Hypertext Preprocessor”) is a widely-used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. PHP development is focused on server-side scripting, but you can do much more with it. To install PHP, execute the command:

```
# zypper install php
...
```

16.4.2 C/C++ development tools

Most Linux distributions come with the C/C++ compiler, `gcc`. This is also known as the *GNU compiler collection* because it can compile other languages such as Fortran but it’s most frequently used to compile C and C++ code. In the minimal SLES10 installation, none of the development packages are installed. In order to use `gcc`, you must install it using yast:

```
# zypper install gcc
...
# rpm -qa | grep gcc
libgcc43-4.3.4_20091019-0.7.35
gcc43-4.3.4_20091019-0.7.35
gcc-4.3-62.198
libgcc43-32bit-4.3.4_20091019-0.7.35
```

The GNU debugger, or **gdb**, is a very popular and robust debugger for C/C++ programs. You can step through your program (that has been successfully compiled) to see where it is failing. Install it using yast:

```
# zypper install gdb
```

There is a good tutorial on getting started with `gdb`:

http://www.unknownroad.com/rtfm/gdbtut/gdbuse.html

Keep in mind that you can also set breakpoints at functions in the code. Refer to the manual page of **gdb** for more information: **man gdb**.

To make a large program more manageable, developers usually create a makefile that specifies instructions on how to compile a program. Then use the GNU **make** tool to use the makefile to make a working program. To install `make`, issue the command:

```
# zypper install make
```

16.4.3 Java development tools

SLES 11 comes with IBM Java Standard Development Kit (SDK) which is needed if you want to develop Java applications. You need a Java Runtime Environment (JRE) if you only want to run Java applications. The following command installs both:

```
# zypper install java-1_6_0-ibm
```

A good Java debugger is **jdb**, it comes with IBMJava2-SDK and can be run similar to **gdb**. A good tutorial is on the Web at:

http://java.sun.com/j2se/1.3/docs/tooldocs/solaris/jdb.html

You can use the GNU **make** to build from Java makefiles or the more recent and popular Ant. Ant uses XML technology. Here's a great guide to get you started with either tool:

http://www.onlamp.com/pub/a/onlamp/2004/11/18/gnumake_3e.html

16.4.4 Setting up subversion

If you need to set up a source code control system, subversion, or *svn*, is a good choice. To set up svn, perform the following steps.

- Install the subversion RPM on RHEL 6.2, perform the following step:

```
# yum install subversion
...
```

- To install the subversion RPM on SLES 11 SP2, mount the first Software Development Kit (SDK) DVD, then install `libapr` and `libneon` through yast, then install the `subversion` RPM

```
# mount 129.40.45.3:/mnt/sles11sp2sdk /mnt
# yast # ... => install libapr* and libneon* RPMs
# rpm -ivf subversion-1.6.17-1.11.1.s390x.rpm
Preparing packages for installation...
subversion-1.6.17-1.11.1
Updating etc/sysconfig/svnserve...
```

- Set up an `svn` group and user with both the *gid* and *uid* being `500`:

```
# groupadd -g 500 svn
# useradd -c "SVN" -d /var/svnrepos -g 500 -u 500 -s /sbin/nologin -M svn
```

- Install xinetd:

```
# yum install xinetd
...
```

- On RHEL 6.2, set up an **xinetd** configuration file for subversion, and start **xinetd**:

```
# cd /etc/xinetd.d/
# vi svn                # create the following file:
# description: svnserve allows access to Subversion repositories using
#              the svn network protocol.
service svn
{
        disable                 = no
        port                    = 3690
        socket_type             = stream
        protocol                = tcp
        wait                    = no
        user                    = svn
        server                  = /usr/bin/svnserve
        server_args             = -i -r /var/svnrepos
}
```

The `-r /var/svnrepos` parameter means this is the starting directory when a client starts a session with the server.

```
# service xinetd start
Starting xinetd: [  OK  ]
```

- On SLES 11 SP2, modify the `/etc/sysconfig/svnserve` configuration file to point to `/var/svnrepos/` directory and remove the **-R** (read-only) flag. Then turn the **svnserve** service on:

```
# cd /etc/sysconfig
# vi svnserve
## Path:        Network/Subversion/svnserve
## Description: Basic configuration for svnserve

## Type:        string
## Default      "-d r /srv/svn/repos"
#
# Default options for the svnserve process.
# The -R option enforces read-only access, i.e. write operations to the
# repository (such as commits) will not be allowed.
# Authentication should be configured before allowing write access.
# See
http://svnbook.red-bean.com/en/1.5/svn.serverconfig.svnserve.html#svn.serverconfig.svnse
rve.auth
#
SVNSERVE_OPTIONS="-d -R -r /var/svnrepos"
...
# service svnserve start
Starting svnserve                                                    done
# chkconfig svnserve on
```

- Check that the server is now listening:

```
# netstat -l | grep svn
tcp        0      0 *:svn                       *:*                          LISTEN
```

- Made a directory for the subversion repositories and change into that directory:

```
# mkdir /var/svnrepos
# cd /var/svnrepos
```

- Create a new repository (**foo** in this example) with the **svnadmin create** command:

```
# svnadmin create foo
# cd foo
```

- Make the `svn` user and group owner of all new directories:

```
# chown -R svn.svn /var/svnrepos
```

- Set up the configuration file for the new project for local access. In the file `conf/svnserve.conf` uncomment and set **anon-access** to none, and ucomment the **auth-access** and **passwd-db** lines:

```
# cd conf
# vi svnserve.conf
### This file controls the configuration of the svnserve daemon, if you
### use it to allow access to this repository.  (If you only allow
### access through http: and/or file: URLs, then this file is
### irrelevant.)

### Visit http://subversion.tigris.org/ for more information.

[general]
### These options control access to the repository for unauthenticated
### and authenticated users.  Valid values are "write", "read",
### and "none".  The sample settings below are the defaults.
anon-access = none
auth-access = write
### The password-db option controls the location of the password
```

```
### database file.  Unless you specify a path starting with a /,
### the file's location is relative to the directory containing
### this configuration file.
### If SASL is enabled (see below), this file will NOT be used.
### Uncomment the line below to use the default password file.
password-db = passwd
...
```

- Add a user and password. In this example it is **mikemac** and **passwd**:

```
# vi passwd
### This file is an example password file for svnserve.
### Its format is similar to that of svnserve.conf. As shown in the
### example below it contains one section labelled [users].
### The name and password for each user follow, one account per line.

[users]
mikemac = passwd
...
```

- Set key-based or passwordless authentication for the local host. First note that a password is need to SSH to the local host:

```
# ssh localhost hostname
The authenticity of host 'localhost (::1)' can't be established.
RSA key fingerprint is 2b:6c:1a:0a:9f:d5:8f:22:0a:63:15:bd:60:c7:d6:99.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (RSA) to the list of known hosts.
Password:
zntc75
```

- Copy the SSH DSA public key to the file `authorized_keys` and verify that key-based authentication is working:

```
# cd ~/.ssh
# ls
id_dsa  id_dsa.pub  known_hosts
# cat id_dsa.pub > authorized_keys
# ssh localhost hostname
zntc75
```

The subversion server should now be configured with one empty project - `foo` in this example.

Populate the project

The project is now ready to be populated with files. To do so, perform the following steps:

- Create directories for development. In this example - in this case under `/root/foo`. The example package has five main directories where the code will reside when installed on a target Linux system:

```
# cd /root
# mkdir -p foo/etc
# mkdir -p foo/usr/local/sbin
# mkdir -p foo/usr/local/share/foo
# mkdir -p foo/srv/www/cgi-bin
# mkdir -p foo/usr/src/packages/SPECS
# cd foo/etc
# scp <hostWithCode>:/etc/foo.conf .
...     # and populate the other directories
# cd
# ls -R foo
foo:
etc  srv  usr
```

```
foo/etc:
foo.conf

foo/srv:
www
...
```

The initial source code is now populated in the project.

- Import the source code into subversion under the directory `/root/foo/`:

```
# svn import /root/foo file:///var/svnrepos/foo/ -m "Initial import of files"
Adding         /root/src/foo/usr
Adding         /root/src/foo/usr/local
Adding         /root/src/foo/usr/local/share
Adding         /root/src/foo/usr/local/share/foo
...
Committed revision 1.
```

- The files have been copied to this server, then pulled in to subversion. However, subversion does not know about the `/root/foo/` directory evidenced by the fact that there are no `.svn/` subdirectories. Remove the unversioned files from the directory `/root/foo/` and check the files back out:

```
# rm -fr /root/foo/*
# svn co svn://localhost/foo /root/foo
A    /root/foo/usr
A    /root/foo/usr/src
A    /root/foo/usr/src/packages
A    /root/foo/usr/src/packages/SPECS
A    /root/foo/usr/src/packages/SPECS/foo-server.spec
A    /root/foo/usr/local
...
```

- Note that `.svn/` subdirectories have now been created in each directory:

```
# find /root/foo -name .svn
/root/foo/srv/.svn
/root/foo/srv/www/cgi-bin/.svn
/root/foo/srv/www/.svn
/root/foo/.svn
...
```

This shows that a subversion repository has been set up and populated.

16.4.5 Create an RPM

When it comes time to distribute your code, you may want to create and RPM, as this is a common Linux package format. An RPM for a package can be created through the combination of checking code out of SVN, the **rpm-build** package and a *spec file* for the package.

- Install the `rpm-build` RPM on RHEL 6.2:

```
# yum install rpm-build
...
```

- Install the `rpm` RPM on SLES 11 SP2:

```
# zypper install rpm
...
```

- Write a script to build your RPM as you will probably have to build it many, many, many times. Following is an example:

```
# cat mkrpm
#!/bin/bash
#+--------------------------------------------------------------------------+
# mkrpm
# Build the foo server RPM
#+--------------------------------------------------------------------------+
function error()
 {
  echo "ERROR!!!: $@"
  exit 1
 }

echo "Removing old files ..."
rm -rf foo   # remove the old files
if [ $? != 0 ]; then
  error 'Failed to remove checkout directory prior to attempting checkout'
fi
echo "Checking out current files ..."
svn co --non-interactive --username mikemac --password passwd svn://zntc75/foo
if [ $? != 0 ]; then
  error 'Failed to check out foo code from SVN repository.'
fi
echo "Deleting old in-place files ..."
rm -rf /usr/src/packages/SPECS/foo-server.spec \
       /usr/local/sbin/foo* \
       /usr/local/sbin/sm* \
       /usr/local/share/foo \
       /etc/foo.conf
if [ $? != 0 ]; then
  error 'Failed to remove old RPM build files'
fi
echo "Copying files to in-place"
cp -a foo/* /
if [ $? != 0 ]; then
  error 'Could not copy foo source files to build locations'
fi
echo "Building RPM ..."
cd /usr/src/packages/SPECS
if [ $? != 0 ]; then
  error 'Could not change working dir to /usr/src/packages/SPECS'
fi
rpmbuild -bb foo-server.spec
if [ $? != 0 ]; then
  error 'Failed building foo RPM'
fi
cd -
if [ $? != 0 ]; then
  error 'Could not go back to original working directory'
fi
rm -rf foo \
       /usr/src/packages/SPECS/foo-server.spec \
       /usr/local/sbin/foo* \
       /usr/local/share/foo \
       /usr/share/doc/packages/foo/* \
       /etc/foo.conf
if [ $? != 0 ]; then
  error 'Failed to clean up RPM build files'
fi
```

```
mv /usr/src/packages/RPMS/s390x/foo*.rpm .
if [ $? != 0 ]; then
  error 'Could not retrieve foo RPM(s) to local directory'
fi
```

This section has quickly shown one way to build an RPM.

16.4.6 Additional resources

The following Web sites are resources for additional application development information:

Scripting languages

http://www.perl.com/
http://www.python.org/
http://www.freeos.com/guides/lsst/

C/C++

http://gcc.gnu.org/onlinedocs/gcc/
http://en.wikipedia.org/wiki/GNU_Compiler_Collection#External_links

Java

http://www.oracle.com/technetwork/java/index.html
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/

Help with vi

http://www.freeos.com/guides/lsst/misc.htm#commonvi

Part 4

Other topics

This part of the book includes the following chapters:

17

z/VM Live Guest Relocation

"If the facts don't fit the theory, change the facts."

--Albert Einstein

z/VM 6.2 now has the ability to relocate Linux guests between members in an SSI cluster. This is known as Live Guest Relocation (LGR). While continuing to run, Linux systems can be moved cross-LPAR on the same CEC, or cross-CEC, if the SSI is set up that way. This new function will allow for very few or even no planned outages.

17.1 LGR considerations

An SSI cluster has two types of virtual machines:

Single-configuration virtual machine

A virtual machine defined by the `USER` statement can be logged on to any member of the SSI cluster, but on only one member at a time. Single-configuration virtual machines are eligible for guest relocation.

Multi-configuration virtual machine

A virtual machine defined by the `IDENTITY` and `SUBCONFIG` statements can be logged on concurrently to multiple members of the SSI cluster. The virtual machines have common attributes but can also be configured to access different resources. Multi-configuration virtual machines are not eligible for guest relocation.

There are many items that must be considered with regard to relocating running Linux systems:

17.1.1 General considerations prior to relocation

When determining the size of a guest being relocated you must take into consideration the following:

- The private virtual disks the virtual machine may have.

- The potential size to which the guest could grow, including standby and reserved memory (storage) settings.
- The level of memory over-commitment currently on the destination system. Relocation may increase paging demands. Therefore, be sure there is at least two times more paging space than the total virtual memory across all guests.
- A guideline is to never allow DASD paging space to go above 50% full. This will give CP space to react to sudden increases in central memory demand. Check on this value with the **CP QUERY ALLOC PAGE** command. If you add in the size of the virtual machine(s) being relocated to the pages in use, and that brings the "in use" percentage above 50%, the relocation may have an negative impact on system performance.
- Use the **VMRELOCATE TEST** command before **VMRELOCATE MOVE**.
- The **SET RESERVED** setting for the guest (if any) on the source system is not carried over to the destination system. This setting for the guest on the destination should be established after the relocation completes based on the available resources and workload on the destination system.

Mandatory memory checking performed during relocation

As part of eligibility checking and in-between memory move passes, relocation ensures the current memory size of Linux fits in available space on the destination system.

- For purposes of the calculation, relocation assumes the Linux's memory is fully populated (including the guest's private virtual disks), and includes an estimate of the size of the supporting CP structures.
- Available space includes the sum of available central, expanded, and auxiliary memory.

There is no way for this check to be bypassed. If it fails, the relocation is terminated. The error message displayed indicates the size of the guest along with the available capacity on the destination system.

Optional memory checking performed during relocation

In addition to the mandatory test described above, by default the following three checks are also performed during eligibility checking and in-between memory passes:

- Will the guest's current memory size (including CP supporting structures) exceed auxiliary paging capacity on the destination?
- Will the guest's maximum memory size (including CP supporting structures) exceed available space (central storage, expanded storage, and auxiliary storage) on the destination?
- Will the guest's maximum memory size (including CP supporting structures) exceed auxiliary paging capacity on the destination?

Note: the maximum memory size includes any standby and reserved memory the guest may have.

If any of these tests fail, the relocation is terminated. The error message(s) displayed indicates the size of the guest along with the available capacity on the destination system.

If you are certain the above three checks are not-applicable to your installation (for instance, because you have an over abundance of central memory and a less than recommended amount of paging space), you can choose to have CP skip these three checks by specifying **FORCE STORAGE** on the **VMRELOCATE** command.

Minimizing link and resource contention:

- Link and resource contention may negatively affect performance and thus increase quiesce time during relocation.
- It is recommended only one relocation be performed at a time. If a set of relocations is to be initiated from a single script or EXEC, this can be accomplished by using the **SYNC** option (the default) on the **VMRELOCATE** command.

17.2 Relocate a Linux system

You can use the **VMRELOCATE** command to move a Linux system from the SSI member on which it is running to another member in the cluster. To accomplish this task, perform the following steps:

- Logon as MAINT on the member where the Linux system is running. In this example the Linux system LINUX157 is running on member 1, POKDEV62.
- Choose a sample Linux system to relocate and verify that it is running on the member. In this example, the target is LINUX157.

```
==> q linux157
LINUX157 - DSC
```

- Issue the **VMRELOCATE TEST** command with a target of the second SSI member. This will test to see if the system is eligible for relocation:

```
==> vmrelo test linux157 poktst62
User LINUX157 is eligible for relocation to POKTST62
```

- You may choose to start a **ping** from another session. For example to ping continuously from a DOS session, issue the following command:

```
c:\>ping /t gpok157.endicott.ibm.com

Pinging gpok157.endicott.ibm.com [9.60.18.157] with 32 bytes of data:

Reply from 9.60.18.157: bytes=32 time=214ms TTL=51
Reply from 9.60.18.157: bytes=32 time=63ms TTL=51
Reply from 9.60.18.157: bytes=32 time=63ms TTL=51
...
```

- Issue the **VMRELOCATE MOVE** command to migrate the running Linux system:

```
==> vmrelo move linux157 poktst62
Relocation of LINUX157 from POKDEV62 to POKTST62 started
User LINUX157 has been relocated from POKDEV62 to POKTST62
```

- Monitor the **ping** session to see if packets are delayed or dropped.
- Verify the Linux system is now running somewhere in the SSI:

```
==> q linux157
LINUX157 - SSI
```

This has shown how to migrate a running Linux system using the **VMRELOCATE** command.

18

Configure DirMaint, SMAPI and RACF

“Science is a wonderful thing if one does not have to earn one's living at it.”

--Albert Einstein

This chapter describes how to enable and configure DirMaint, SMAPI and RACF. If you plan to use xCAT as described in Chapter 21, “xCAT” on page 357, DirMaint and SMAPI are required, so this section must be completed. You may or may not choose to also include RACF. If you turn DirMaint on, you can no longer edit the `USER DIRECT` file and use the **`DIRECTXA`** command, rather, a different DirMaint interface is used.

This chapter consists of the following sections:

18.1 Configure DirMaint

To set up DirMaint, perform the following tasks:

18.1.1 Enable DirMaint

In order to use DirMaint, you need a valid license for it. Please verify that you are licensed before proceeding.

To install and configure DirMaint, perform the following steps:

- Log on to MAINT620 on member 1 of the SSI cluster.
- Enable DirMaint via the following **SERVICE** command:

```
==> service dirm enable
...                                                        // a few screens go by
VMFSRV1233I The following products have been serviced.
VMFSRV1233I DIRM
VMFSRV2760I SERVICE processing completed successfully
```

- Put DirMaint into production with the **PUT2PROD** command:

```
==> put2prod dirm
VMFP2P2760I PUT2PROD processing started
VMFP2P2760I PUT2PROD processing started for DIRM
VMFP2P1233I The following products have been put into production. Recycle the
            appropriate servers.
VMFP2P1233I DIRM
VMFP2P2760I PUT2PROD processing completed successfully
```

- This process appends to the end of your SYSTEM CONFIG file. To see the changes, link to the PMAINT CF0 disk and type the SYSTEM CONFIG file to observe these lines at the end of the file:

```
==> vmlink pmaint cf0
DMSVML2060I PMAINT CF0 linked as 0120 file mode Z
==> type system config z
...    // many screens cleared
PRODUCT PRODID 6VMDIR20 STATE ENABLED DESCRIPTION '06/14/12.10:57:20.MAINT620 In
stall/service DirMaint using minidisk'
```

- Log off of MAINT620.
- **Repeat** the **PUT2PROD DIRM** command from MAINT620 on every other member in the SSI cluster.

DirMaint should now be enabled on the SSI.

18.1.2 Configure DirMaint

To configure DirMaint, perform the following steps:

- Log on to 6VMDIR20 on the first member of the SSI cluster.
- Access the 492 disk as E to get access to the **DIR2PROD EXEC**:

```
==> acc 492 e
```

- Use the **DIR2PROD EXEC** to access the necessary minidisks:

```
==> dir2prod access_new 6vmdir20 dirm
DMSACP726I 492 E released
DIR2PROD: Normal Termination.
```

- Note that you have three new mindisks accessed as J, K and L:

```
==> q disk
LABEL  VDEV M  STAT   CYL TYPE BLKSZ   FILES  BLKS USED-(%) BLKS LEFT  BLK TOTAL
DRM191 191  A   R/W     9 3390 4096        2          12-01      1608       1620
MNT5E5 5E5  B   R/O    18 3390 4096      131        1714-53      1526       3240
MNT51D 51D  D   R/W    26 3390 4096      308        1769-38      2911       4680
DIR1DF 1DF  J   R/W     9 3390 4096       13          20-01      1600       1620
DRM492 492  K   R/W    15 3390 4096      281        1619-60      1081       2700
DRM41F 41F  L   R/W    16 3390 4096       52         661-23      2219       2880
MNT190 190  S   R/O   207 3390 4096      703      19275-52      17985      37260
```

```
MNT19E 19E  Y/S R/O   500 3390 4096     1093      30310-34      59690      90000
```

- Access the 11F disk as file mode Z:

```
==> acc 11f z
DMSACC724I 11F replaces Z (120)
```

- A final disk is needed to access the user directory, USER DIRECT. Link to the MAINT 2CC disk read-only with the **VMLINK** command. The read password will either be what you set all passwords to using the **CHPW620 XEDIT** macro, or **READ**.

```
==> vmlink maint 2cc
ENTER READ PASSWORD:
DMSVML2060I MAINT 2CC linked as 0120 file mode X
```

- Copy the USER DIRECT file from MAINT 2CC (file mode X) to DIRMAINT 1DF (file mode J) as the file USER INPUT. This will cause the current user directory to be loaded into DirMaint when it starts for the first time:

```
==> copy user direct x = input j
```

- Create the file CONFIGAA DATADVH Z. This is the main DirMaint configuration file. Add the following lines:

```
==> x configaa datadvh z
====> a 9
ALLOW_ASUSER_NOPASS_FROM= VSMGUARD *
ALLOW_ASUSER_NOPASS_FROM= VSMWORK1 *
ALLOW_ASUSER_NOPASS_FROM= VSMWORK2 *
ALLOW_ASUSER_NOPASS_FROM= VSMWORK3 *
ASYNCHRONOUS_UPDATE_NOTIFICATION_EXIT.TCP= DVHXNE EXEC
ASYNCHRONOUS_UPDATE_NOTIFICATION_EXIT.UDP= DVHXNE EXEC
DISK_CLEANUP= YES
ONLINE= IMMED
RUNMODE= OPERATIONAL
```

Notes:

– The ALLOW_ASUSER_NOPASS_FROM lines allow SMAPI users to issue commands to the Directory Manager using the ASUSER modifier and the password of that user.

– The ASYNCHRONOUS_UPDATE_NOTIFICATION_EXIT lines activate an exit that will notify SMAPI of changes made to the user directory.

– If privacy of residual data is a concern on your system, use DISK_CLEANUP= YES.

– The ONLINE= IMMED line sets your changes to be made immediately.

– The RUNMODE= OPERATIONAL line sets directory changes to be made. This can be set to **TESTING** and the changes will not be made.

- Release the z disk:

```
==> rel z
```

- Create the file AUTHFOR CONTROL on the J disk (DIRMAINT 1DF). Add 12 lines so that MAINT and LNXADMIN are authorized to perform DirMaint tasks, as well as the SMAPI virtual machines VSMGUARD, VSMWORK1, VSMWORK2 and VSMWORK3.

```
==> x authfor control j
====> a 12
ALL LNXADMIN * 140A ADGHOPS
ALL LNXADMIN * 150A ADGHOPS
ALL MAINT * 140A ADGHOPS
ALL MAINT * 150A ADGHOPS
ALL VSMGUARD * 140A ADGHOPS
ALL VSMGUARD * 150A ADGHOPS
ALL VSMWORK1 * 140A ADGHOPS
```

```
ALL VSMWORK1 * 150A ADGHOPS
ALL VSMWORK2 * 140A ADGHOPS
ALL VSMWORK2 * 150A ADGHOPS
ALL VSMWORK3 * 140A ADGHOPS
ALL VSMWORK3 * 150A ADGHOPS
====> file
```

You have now created the many of the DirMaint configuration files. The next important file is the EXTENT CONTROL file.

18.1.3 Customize the EXTENT CONTROL file

The EXTENT CONTROL file defines disks (volumes) to DirMaint for minidisk allocation. It also contains system and device default values used during allocation operations. There are two main sections that should be populated:

Regions
: Defines disks and their sizes to DirMaint. The AUTOR keyword can be used in user directory entries to take space from the regions.

Groups
: Defines pools of disks so the AUTOG keyword can be used to take space from the pools, not from specific disks.

To configure the EXTENT CONTROL file, perform the following steps:

- Make a copy of the original file:

```
==> copy extent control j = contorig =
```

- From a different 3270 emulator session to MAINT, you may want to use the **QUERY DASD** command to see which disks are attached to SYSTEM. Disregard the CP-owned DASD and the common volumes.

```
==> q da
DASD 61A5 CP SYSTEM JM61A5   1
DASD 61B2 CP SYSTEM JM61B2   1
DASD 6232 CP OWNED  JP6232   0
DASD 6233 CP OWNED  JP6233   0
DASD 6280 CP OWNED  JV6280   102
DASD 6281 CP OWNED  JS6281   1
DASD 6282 CP OWNED  JP6282   0
DASD 6283 CP SYSTEM JV6283   19
DASD 6284 CP OWNED  CV6284   13
DASD 6285 CP SYSTEM CV6285   12
DASD 6286 CP SYSTEM CV6286   2
DASD 6287 CP SYSTEM CV6287   8
DASD 6288 CP OWNED  JP6288   0
DASD 6289 CP SYSTEM JM6289   2
DASD 628A CP OWNED  JP628A   0
DASD 628C CP SYSTEM JM628C   0
DASD 628D CP SYSTEM JM628D   0
DASD 628E CP SYSTEM JM628E   0
DASD 628F CP SYSTEM JM628F   0
DASD 6290 CP SYSTEM JM6290   1
DASD 6293 CP SYSTEM JM6293   0
DASD 6294 CP SYSTEM JM6294   0
DASD 6327 CP SYSTEM JM6327   0
DASD 6328 CP SYSTEM JM6328   0
DASD 6339 CP SYSTEM JM6339   0
DASD 633A CP SYSTEM JM633A   0
DASD 639C CP OWNED  WS639C   0
```

- Add the DASD attached to `SYSTEM` to the `:REGIONS.` section (assuming these volumes will be available for minidisk creation). The convention used in this example is that the `RegionID`, field 1, is set to the `VolSer`, field 2. Fields 3 and 4 set the cylinder range to all cylinders except cylinder 0, and the `Dev-Type`, the last field, informs DirMaint as to the size of the disk. If you are not sure of the device type, use the **QUERY DASD DETAILS <rdev>** command from `MAINT`. Each region name is also added to a group named `POOL1`.

```
==> x extent control j
* *********************************************************************
...

 Purpose:   Default Extent Control file.
...
* *********************************************************************
:REGIONS.
*RegionId  VolSer    RegStart      RegEnd  Dev-Type  Comments
JM61A5     JM61A5    0001          END     3390-09
JM61B2     JM61B2    0001          END     3390-09
JM6289     JM6289    0001          END     3390-03
JM628C     JM628C    0001          END     3390-03
JM628D     JM628D    0001          END     3390-03
JM628E     JM628E    0001          END     3390-03
JM628F     JM628F    0001          END     3390-03
JM6290     JM6290    0001          END     3390-03
JM6293     JM6293    0001          END     3390-03
JM6294     JM6294    0001          END     3390-03
JM6327     JM6327    0001          END     3390-03
JM6328     JM6328    0001          END     3390-03
JM6339     JM6339    0001          END     3390-03
JM633A     JM633A    0001          END     3390-03
:END.
:GROUPS.
*GroupName RegionList
  POOL1 JM61A5 JM61B2 JM6289 JM628C JM628D JM628E JM628F JM6290
  POOL1 JM6293 JM6294 JM6327 JM6328 JM6339 JM633A

:END.
:SSI_VOLUMES.
* Added during Installation, Do not remove.
*VolumeFamily     Member   VolSer
IBM_RES           POKDEV62 JV6280
IBM_WORK1         POKDEV62 JV6283
IBM_RES           POKTST62 WV639B
IBM_WORK1         POKTST62 WV639E
:END.
:DEFAULT_GROUPS.
*GroupName Member
:END.
:EXCLUDE.
* entry_name Address
MAINT*       012*
MAINT*       013*
PMAINT       013*
SYSDUMP1     012*
SYSDMP*      012*
:END.
:AUTOBLOCK.
* IBM supplied defaults are contained in the AUTOBLK DATADVH file.
* The following are customer overrides and supplements.
*
```

```
*DASDType BlockSize Blocks/Unit Alloc_Unit Architecture
:END.
:DEFAULTS.
* IBM supplied defaults are contained in the DEFAULTS DATADVH file.
* The following are customer overrides and supplements.
*
*DASDType Max-Size
:END.
====> file
```

- Log off of 6VMDIR20.

You should now have the EXTENT CONTROL file configured, which is read when DirMaint starts.

18.1.4 Start DirMaint

To start DirMaint, perform the following steps:

- Log on to MAINT on the first SSI member.
- Issue the following command which is really two separate commands. The command on the left half of the "#" (the line-end character) starts DIRMAINT with the **XAUTOLOG** command and the **SYNC** option that returns control to MAINT. The second command, on the right side of the "#", sets MAINT to be the secondary user of DIRMAINT. This way, DIRMAINT does not have to be logged on to, but MAINT can see its console output.

  You may want to issue a **#CP TERM MORE 0 0** as there will be many, many screens of output:

```
==> xautolog dirmaint sync # set secuser dirmaint *
AUTO LOGON  ***       DIRMAINT USERS = 13
Ready; T=0.01/0.01 12:17:51
HCPCFX6768I SECUSER of DIRMAINT initiated.
Ready; T=0.01/0.01 12:17:51
DIRMAINT: z/VM V6.2.0    2012-06-07 17:58
DIRMAINT: ................................................................
..
DIRMAINT:
DIRMAINT: PRODUCT:
DIRMAINT: IBM Directory Maintenance Facility for z/VM (DirMaint)
DIRMAINT: 5741-A07 (C) Copyright IBM Corporation 1979, 2011.
DIRMAINT: Function Level 620 Service Level 0000.
DIRMAINT: DMSACC724I 155 replaces A (191)
DIRMAINT: DMSACP723I F (551) R/O
DIRMAINT: DMSACC723I X (01DE) R/W - OS
DIRMAINT:
 DVHPRO2008I ROLE = DIRMAINT
DIRMAINT: HCPMFS057I OPERATOR not receiving; disconnected
DIRMAINT: DVHPRO2008I ROLE = DIRMAINT
DIRMAINT:
 DIRMAINT: DVHPRO2010I TESTING USE OF MSGNOH ...
DIRMAINT: DASD 0192 DETACHED
DIRMAINT: DASD 021F DETACHED
DIRMAINT: ................................................................
..
DIRMAINT: DIRMAINT POKDEV62. - 2012/06/14; T=0.02/0.02 12:17:51
 DIRMAINT: DVHILZ3510I Starting DVHINITL with directory: USER INPUT E
 DIRMAINT: DVHILZ3510I DVHINITL Parms: BLDMONO NOCRCWARN
...
DIRMAINT: DVHWAI2140I Waiting for work on 12/06/19 at 12:00:57.
```

Watch for errors. Note the message suggesting that the DirMaint directory is being initialized using the file USER INPUT which was copied from USER DIRECT earlier.

- Turn the secondary user setting off so MAINT will no longer see the DIRMAINT console messages:

```
==> set secuser dirmaint off
DIRMAINT: HCPCFX6769I Your SECUSER terminated by MAINT.
HCPCFX6769I SECUSER of DIRMAINT terminated.
```

DirMaint should now be running and should have read the USER INPUT, CONFIGAA DATADVH, AUTHFOR CONTROL and EXTENT CONTROL configuration files.

18.1.5 Test DirMaint

To test DirMaint, perform the following steps:

- Send the **needpass no** command and type in the MAINT password so that a password is not needed for every DirMaint command. Watch for a 0 return code.

```
==> dirm needpass no
File DIRMAINT NEWMAIL E2 sent to * at POKDEV62 on 08/11/11 11:27:36
DVHXMT1181R Enter the current logon password of MAINT at POKDEV62 for
DVHXMT1181R authentication.  It will not be displayed on the
DVHXMT1181R terminal.  To exit without processing the command, just
DVHXMT1181R press ENTER.

DVHXMT1191I Your NEEDPASS request has been sent for processing to
DVHXMT1191I DIRMAINT at POKDEV62.
Ready; T=0.02/0.02 11:27:38
 DVHREQ2288I Your USEROPTN request for MAINT at * has been accepted.
 DVHBIU3450I The source for directory entry MAINT has been updated.
 DVHBIU3456I Object directory update is not required for this source
 DVHBIU3456I update.
 DVHREQ2289I Your USEROPTN request for MAINT at * has completed; with
 DVHREQ2289I RC = 0.
```

If you do not get a 0 return code, go back and review your configuration settings.

- Issue **DIRMAINT REVIEW** command. You should no longer need to supply the MAINT password. This will send a file to MAINT's reader containing an overview of the directory entry:

```
==> dirm rev
DVHXMT1191I Your REVIEW request has been sent for processing to DIRMAINT
DVHXMT1191I at POKDEV62.
Ready; T=0.01/0.01 11:28:26
 DVHREQ2288I Your REVIEW request for MAINT at * has been accepted.
RDR FILE 0012 SENT FROM DIRMAINT PUN WAS 0013 RECS 0412 CPY  001 A NOHOLD NOKEEP
 DVHREQ2289I Your REVIEW request for MAINT at * has completed; with RC
 DVHREQ2289I = 0.
```

- The file number of the file sent to the reader can be used, which in this example is ***12***. Use the **PEEK** command to view the file. The **for *** parameter specifies to view all records.

```
==> peek 12 (for *
IDENTITY MAINT XXXXXXXX 128M 1000M ABCDEFG

DVHRXV3366I The following configurations will be used on SSI nodes.
DVHRXV3366I The following configuration MAINT-1 will be used on SSI node
DVHRXV3366I POKDEV62.
SUBCONFIG MAINT-1
...
```

This shows that DirMaint is configured and functioning.

18.1.6 Test DirMaint at IPL time

It is recommended that you shut down and re-IPL the system. With z/VM 6.2, there is new code in AUTOLOG1's **PROFILE EXEC** to start DirMaint. To do so, perform the following steps:

- **If you are sure you are in a position to do so**, issue the **SSISHUTD** command now with the **REIPL** parameter:

```
==> ssishutd reipl
Are you sure you want to SHUTDOWN REIPL the SSI cluster? (y/n)
y
SYSTEM SHUTDOWN STARTED
HCPSHU960I System shutdown may be delayed for up to 630 seconds
VMSERVP : DMS5BC3108I Shutdown Signal received. STOP processing started
...
```

  You will lose your 3270 emulator session(s). If you watch the HMC, the SSI member LPARs should immediately turn from white to green, then return to white after a minute or so.

- After the system comes back, log on **as MAINT**.
- Run the **QUERY NAMES** command on all SSI members with the **SSICMD** command:

```
==> ssicmd q n
POKDEV62:
DIRMSAT2 - SSI
FTPSERVE - DSC , TCPIP    - DSC , DIRMAINT - DSC , DTCVSW2  - DSC
DTCVSW1  - DSC , VMSERVP  - DSC , VMSERVR  - DSC , VMSERVU  - DSC
VMSERVS  - DSC , OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC
OPERATOR - DSC , MAINT    -L0003
VSM     - TCPIP

POKTST62:
DIRMAINT - SSI , VMSERVP  - SSI
FTPSERVE - DSC , TCPIP    - DSC , DIRMSAT2 - DSC , DTCVSW2  - DSC
DTCVSW1  - DSC , VMSERVR  - DSC , VMSERVU  - DSC , VMSERVS  - DSC
OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
VSM     - TCPIP
```

The DIRMAINT and DIRMSAT2 virtual machines are the primary and satellite DirMaint workers.

This shows that DirMaint is started on both SSI members after a z/VM IPL.

18.2 Configure SMAPI

Once DirMaint (or another directory maintenance product) is configured, SMAPI can be enabled and configured. To set up SMAPI, perform the following tasks:

- "Set up basic SMAPI configuration"
- "Disable support for ensembles" on page 281
- "Start SMAPI at IPL time" on page 282
- "Test SMAPI" on page 283

18.2.1 Set up basic SMAPI configuration

The following steps need to be performed on only one SSI member:

- Log on to MAINT on SSI member 1.
- Grant authority to the VSMGUARD virtual machine to use certain Shared File System (SFS) directories with the following three **GRANT** commands:

```
==> grant authority vmsys:vsmwork1. to vsmguard (write newwrite
==> grant authority vmsys:vsmwork1.data to vsmguard (write newwrite
==> grant authority * * vmsys:vsmwork1. to vsmguard (read
```

- Access the shared file system VMSYS:VSMWORK1 as your F disk in read-write mode:

```
==> access vmsys:vsmwork1. f (forcerw
```

- Edit the file VSMWORK1 AUTHLIST on that disk:

```
==> x vsmwork1 authlist f
```

- Duplicate the last line by putting a double-quote in the prefix area (it is important to use this to duplicate the line because the lines are 195 characters wide):

```
00001 DO.NOT.REMOVE                                                    DO.NOT.RE
MOVE
00002 MAINT                                                            ALL
"     VSMPROXY                                                         ALL
```

- Press **Enter** and the line will be duplicated. Replace the user ID with LNXADMIN and save the file:

```
00001 DO.NOT.REMOVE                                                    DO.NOT.RE
MOVE
00002 MAINT                                                            ALL
00003 VSMPROXY                                                         ALL
00004 LNXADMIN                                                         ALL
====> file
```

This change will allow the LNXADMIN virtual machine to invoke SMAPI calls.

18.2.2 Disable support for ensembles

Assuming your system will not be ensemble-managed, virtual machines related to ensembles need to be commented out in a certain configuration file. To turn off ensembles, perform the following steps:

- As MAINT, access the 193 disk as file mode G:

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

- Access the shared file system vmsys:vsmwork1.data disk read-write as file mode H:

```
==> acc vmsys:vsmwork1.data h (forcerw
```

- Copy the DMSSISVR NAMES file from MAINT 193 to the SFS disk:

```
==> copy dmssisvr names g = = h
```

- Edit the DMSSISVR NAMES file and comment out the last four servers in the file by putting asterisks in the first column of each line:

```
==> x dmssisvr names h
...
**************************************************************
*** the following machines are only available in ensembles ***
**************************************************************
```

```
* Default Management Network Server
*:server.VSMREQIM
*:type.REQUEST
*:protocol.AF_MGMT
*:address.INADDR_ANY
*:port.44446

* Primary Vswitch Controller
*:server.DTCENS1
*:type.VCTRL

* Backup Vswitch Controller
*:server.DTCENS2
*:type.VCTRL

* Management Guest
*:server.ZVMLXAPP
*:type.MG
```

These settings will ensure that virtual machines related to ensembles (especially DTCENS1 and DTCENS2) do not start automatically when SMAPI is started.

18.2.3 Start SMAPI at IPL time

To have SMAPI start at IPL time, add one line to the **PROFILE EXEC** on the AUTOLOG1 191 disk. To accomplish this task, perform the following steps:

- Link the LNXMAINT 192 disk read-write and access it as file mode I:

```
==> link autolog1 191 1191 mr
DASD 1192 LINKED R/W;
==> acc 1191 i
```

- Edit the PROFILE EXEC and add one line to start SMAPI:

```
==> x profile exec i
...
/*******************************************************************/
/* Customer processing can be added here                          */
/*******************************************************************/
"CP XAUTOLOG TCPIP"             /* Start TCPIP                  */
"CP SET MDC STOR OM 256M"       /* Limit minidisk cache in CSTOR    */
"CP SET MDC XSTORE OM OM"       /* Disable minidisk cache in XSTOR  */
"CP SET SIGNAL SHUTDOWN 600"    /* Allow guests 10 min to shut down */
"CP XAUTOLOG LNXADMIN"          /* Start the Linux admin machine     */
"CP XAUTOLOG VSMGUARD"          /* Start SMAPI                      */
...
```

- Repeat this for all other members in the SSI cluster.

Verify SMAPI comes up at IPL time

- Query the virtual machines running with the **SSICMD EXEC** and the **QUERY NAMES** command to query all active virtual machines on all members:

```
==> ssicmd q n
POKDEV62:
DIRMSAT2 - SSI
FTPSERVE - DSC , TCPIP    - DSC , DIRMAINT - DSC , DTCVSW2  - DSC
DTCVSW1  - DSC , VMSERVP  - DSC , VMSERVR  - DSC , VMSERVU  - DSC
VMSERVS  - DSC , OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC
OPERATOR - DSC
```

```
VSM     - TCPIP

POKTST62:
DIRMAINT - SSI , VMSERVP  - SSI
FTPSERVE - DSC , TCPIP    - DSC , DIRMSAT2 - DSC , DTCVSW2  - DSC
DTCVSW1  - DSC , VMSERVR  - DSC , VMSERVU  - DSC , VMSERVS  - DSC
OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
MAINT    -L0003
VSM     - TCPIP
```

- **If you are sure you're in a position to do so**, shut down and re-IPL the SSI cluster with the supplied **SSISHUTD EXEC**:

```
==> ssishutd reipl
Are you sure you want to SHUTDOWN REIPL the SSI cluster? (y/n)
y
SYSTEM SHUTDOWN STARTED
HCPSHU960I System shutdown may be delayed for up to 630 seconds
VMSERVP : DMS5BC3108I Shutdown Signal received. STOP processing started
...
```

- When the system comes back up, log on to the first SSI member as MAINT.
- Query the virtual machines running with the **SSICMD EXEC** as a reference:

```
==> ssicmd q n
POKDEV62:
DIRMSAT2 - SSI
VSMEVSRV - DSC , VSMPROXY - DSC , VSMREQIU - DSC , VSMREQI6 - DSC
VSMREQIN - DSC , DTCSMAPI - DSC , PERSMAPI - DSC , VSMWORK3 - DSC
VSMWORK2 - DSC , VSMWORK1 - DSC , FTPSERVE - DSC , VSMGUARD - DSC
TCPIP    - DSC , DIRMAINT - DSC , DTCVSW2  - DSC , DTCVSW1  - DSC
VMSERVP  - DSC , VMSERVR  - DSC , VMSERVU  - DSC , VMSERVS  - DSC
OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
MAINT    -L0003
VSM     - TCPIP

POKTST62:
DIRMAINT - SSI , VMSERVP  - SSI
VSMEVSRV - DSC , VSMPROXY - DSC , VSMREQIU - DSC , VSMREQI6 - DSC
VSMREQIN - DSC , DTCSMAPI - DSC , PERSMAPI - DSC , VSMWORK3 - DSC
VSMWORK2 - DSC , VSMWORK1 - DSC , FTPSERVE - DSC , VSMGUARD - DSC
TCPIP    - DSC , DIRMSAT2 - DSC , DTCVSW2  - DSC , DTCVSW1  - DSC
VMSERVR  - DSC , VMSERVU  - DSC , VMSERVS  - DSC , OPERSYMP - DSC
DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
VSM     - TCPIP
```

The SMAPI virtual machines are shown in bold. SMAPI should now be running and configured.

18.2.4 Test SMAPI

To test SMAPI, a REXX EXEC named **CALLSM1** has been included with the files associated with this book. It should have been copied to the MAINT 191 (A) disk. To use it, perform the following steps:

- Log on to MAINT on member 1.
- Verify that the **CALLSM1 EXEC** has been copied to the MAINT 191 disk:

```
==> listfile callsm1 *
CALLSM1  EXEC     A1
```

- Link to the `TCPMAINT 592` disk:

```
==> vmlink tcpmaint 592
DMSVML2060I TCPMAINT 592 linked as 0120 file mode Z
```

- Run the **CALLSM1 EXEC**:

```
==> callsm1

                     buffLen = 57
0000 00000035 00000019 496D6167 655F4465 *    5    Image_De *
0016 66696E69 74696F6E 5F517565 72795F44 * finition_Query_D *
0032 4D000000 00000000 00000000 054D4149 * M            MAI *
0048 4E540000 00032A20 00                * NT    *          *

calling send()
receiving requestId,  buffLen = 4
returned from recv() rc,retvalue =0,4
Request id:= 3756453462

receiving length, buffLen = 4
returned from recv() rc,retvalue =0,4
receiving data, buffLen = 2808
returned from recv() rc,retvalue =0,2808

Request id: 3756453462 Return code:0 Reason code:0 possible outdata len:2792

<COMMAND_DEFINE_CPU=>
<COMMAND_SET_CPUAFFINITY=>
<COMMAND_SET_SHARE=>
<COMMAND_SET_VCONFIG=>
<CONSOLE=VDEV=0009 DEVTYPE=3215 CLASS=T>
...
<VMRELOCATE=>
```

This output shows that SMAPI is working.

18.3 Enable and configure RACF

Important: If you plan to enable RACF, consider Alan's words of wisdom:

1. You must decide on the set of activities you wish to audit, and whether audit is always on for those activities or only on demand. It will be necessary to **LINK** and **ACCESS** the active SMF disk to see how fast it is filling. In a Linux farm, most of the activity will be the system programmers and system administrators doing what they do.
2. If both the primary and secondary SMF minidisks unexpectedly become full, then no more audit records can be recorded, even though security-relevant events can continue to occur. Naturally, any such loss of audit records is unacceptable in a secure system. The `SEVER YES` setting in the `SMF CONTROL` file instructs RACF to *sever* when this happens. It is there to ensure "If it didn't get written down, it didn't happen." An excellent policy to have if you are being cross-examined on the witness stand (possibly as the Accused) in a data theft case.
3. The SMF log disks need to be sized to hold an audit log that has all of the data for a single archive interval. That is, if `RACFSMF` is logged on once a day, then the SMF disks need to be large enough to hold one day's worth of data. (Since there are two disks, it can actually hold double that amount per day.)
4. The `RACFSMF 192` archive disk needs to be large enough to hold 'n' archives, where 'n' is your defined value. This is a safety mechanism. The oldest files need to be erased as required to make room for the latest archive. Warning: As shipped, RACFSMF is not this smart. It simply sends a message to `OPERATOR` when the disk is 80% full. How quaint.
5. You must modify RACFSMF to send the newly-archived file to a more permanent location. It can FTP it, put it in SFS, SENDFILE to MVS, dump to tape, FLASHCOPY the 192 to the next in a series of disks,... whatever. It would be useful to have some pre-packaged skeleton activities in SMFPROF.

This section assumes a new RACF database is being created. For migrating an existing RACF database, see the *RACF Program Directory*, on the Web at:

http://www.vm.ibm.com/progdir/6vmrac20.pdf

This section also assumes that DirMaint and SMAPI have been configured as per the previous two sections in this chapter. To configure RACF on a new z/VM 6.2 system, perform the following steps. The first five steps are done before RACF is started. Steps six and seven put RACF into production. The last step is performed after RACF is in production.

1. "Create the RACF command file" on page 285
2. "Customize SMF" on page 287
3. "Delete the ICHRCX02 exit" on page 288
4. "Copy the RACF databases" on page 290
5. "Set up the AUTOLOG1 and AUTOLOG2 virtual machines" on page 295
6. "Enable RACF" on page 295
7. "Put RACF into production on all members" on page 296
8. "Configure SMAPI to work with RACF" on page 302

18.3.1 Create the RACF command file

To set up the initial RACF database, a set of RACF commands is constructed from the user directory source file, then modified later. The **RPIDIRCT EXEC** helps you migrate the user directory data to a RACF database. It translates directory statements into RACF commands and puts them in an output file named `RPIDIRCT SYSUT1`.

To create `RPIDIRCT SYSUT1` for later use with **RPIDIRCT**, perform the following steps:

- Log on to `MAINT` on the first SSI member.
- Link the `6VMRAC20 191` disk read-write and access it as file mode `F`:

```
==> link 6vmrac20 191 1191 mr
==> acc 1191 f
```

- Link the `6VMRAC20 505` disk read-write and access it as file mode `G`:

```
==> link 6vmrac20 505 1505 mr
==> acc 1505 g
```

- Get the current user directory with passwords with the **DIRMAINT USER WITHPASS** command:

```
==> dirm user withpass
DVHXMT1191I Your USER request has been sent for processing to DIRMAINT
DVHXMT1191I at POKDEV62.
 DVHREQ2288I Your USER request for MAINT at * has been accepted.
RDR FILE 0004 SENT FROM DIRMAINT PUN WAS 0005 RECS 4539 CPY  001 A NOHOLD NOKEEP
 DVHREQ2289I Your USER request for MAINT at * has completed; with RC = 0.
```

- Receive the file onto the `6VMRAC20 191` disk (`F`). In this example the reader file was number 4 noted from the previous command output:

```
==> receive 4 = = f
File USER WITHPASS F0 created from USER WITHPASS A0 received from DIRMAINT at PO
KDEV62
```

- Create the `RPIDIRCT SYSUT1` file from the user directory with the **RPIDIRCT** command. Enter "**n**" to the question of changing the default group ID. This will allow RACF to give all existing virtual machines access to the resources they currently have.

  You may want to issue a **#CP TERM MORE 0 0** as there will be many, many screens of output:

```
==> rpidirct user withpass f
Output defaulted to "A" disk.
  Default group ID = SYS1.
  Would you like to change this default?
  Enter Y/N
n
  Default group ID = SYS1.

PROFILE IBMDFLT

PROFILE TCPCMSU

PROFILE TCPGCSU

PROFILE TCPSSLU

SUBCONFIG MAINT-1
...
**********  4424 Directory records processed  *********

***************  RPIDIRCT SYSUT1 CREATED  **************
```

- Make a copy of the newly created `RPIDIRCT SYSUT1` file to have a reference:

```
==> copy rpidirct sysut1 a = sysuorig =
```

- In the newly created RPIDIRCT SYSUT1 file, remove all lines with the text VMBATCH. A generic VMBATCH profile will be created shortly. All lines can be deleted with the **ALL** subcommand and the prefix command **d*** (hidden lines will not be deleted):

```
==> x rpidirct sysut1
====> all /VMBATCH/
====> top
d*=== * * * Top of File * * *
===== --------------------  22  line(s) not displayed --------------------
===== RDEFINE VMBATCH $ALLOC$ OWNER($ALLOC$) UACC(NONE)
...
====> all
```

All lines with VMBATCH should now be deleted.

- Add the following lines to the bottom of the RPIDIRCT SYSUT1 file:

```
====> bot
====> a 4
setropts generic(vmbatch) gencmd(vmbatch)
rdefine vmbatch ** uacc(none)
permit ** class(vmbatch) id(ftpserve vmnfs dirmsat dirmsat2) acc(control)
setropts classact(vmbatch vmmdisk vmcmd vmlan surrogat)
====> file
```

Notes:

– The first two lines make VMBATCH a generic class.

– The third line permits the FTP, NFS and DirMaint satellite servers to the VMBATCH class. The number of DIRMSAT* entries should correspond to the number of members in the SSI (for example, if you have a four member SSI, add DIRMSAT3 and DIRMSAT4).

– The fourth line activates the classes VMBATCH, VMMDISK, VMCMD, VMLAN and SURROGAT.

- Move the file to the 6VMRAC20 191 disk (F) with the following commands:

```
==> copy rpidirct sysut1 a = = f
==> erase rpidirct sysut1 a
```

The modified RPIDIRCT SYSUT1 file should now be on the 6VMRAC20 191 disk.

18.3.2 Customize SMF

One of the reasons that you run RACF on your z/VM system is to be able to audit who is doing what on the system. In order to do that, the audit records must be managed. This is managed through the RACFSMF virtual machine.

To create a **PROFILE EXEC** for the RACFSMF virtual machine, perform the following steps:

- Link the RACFSMF 191 disk read-write and access it as file mode H:

```
==> link racfsmf 191 2191 mr
==> acc 2191 h
```

- Copy the sample profile SMFPROF EXEC to the RACFSMF 191 disk (H) as the file PROFILE EXEC:

```
==> copy smfprof exec g profile = h
```

- Edit the **PROFILE EXEC** and change the value of Smffreq to **AUTO** and Smfswtch to **NO**:

```
==> x profile exec h
====> /Smfdisk
====> =
...
Smfdisk  = 192
```

```
Smfpct   = 80
Smfinfo  = 'OPERATOR'     /* Default message receiver        @VA45455*/
Smffreq  = 'AUTO'         /* Valid values: DAILY, WEEKLY, MONTHLY,    */
                          /*               AUTO              @VA45455*/
Smfday   = 'MONDAY'       /* Valid values: SATURDAY - FRIDAY  @VA45455*/
Smfswtch = 'NO'           /* Valid values: YES NO            @VA45455*/
...
====> file
```

The **PROFILE EXEC** is now configured for the RACFSMF virtual machine.

Modify the SMF CONTROL file

To set SEVER YES in the SMF CONTROL file on the RACFVM 191 disk, perform the following steps:

- Link to the RACFVM 191 disk read-write and access it as file mode I:

```
==> link racfvm 191 3191 mr
==> acc 3191 i
```

- Edit the SMF CONTROL file and change SEVER NO to **SEVER YES**:

```
==> x smf control i
====> pre off
* * * Top of File * * *
CURRENT 301 K PRIMARY 301 K SECONDARY 302 K 10000 VMSP CLOSE 001 SEVER YES 0 RAC
====> file
```

  Setting this value to **YES** will cause RACF to disconnect from CP if the SMF disk is full.

- Copy the modified SMF CONTROL file to the RACFSMF 191 (H) disk:

```
==> copy smf control i = = h
```

- Link the RACMAINT 191 disk read-write and access it as file mode J:

```
==> link racmaint 191 4191 mr
==> acc 4191 j
```

- Copy the modified SMF CONTROL file to the RACMAINT 191 disk (J) with the **REPLACE** option:

```
==> copy smf control i = = j (rep
```

- Log off of MAINT.

The SMF configuration of RACF should now be complete.

18.3.3 Delete the ICHRCX02 exit

Modify the RACF exit named ICHRCX02 to not allow alternate users to access resources that can be accessed by the FTP and NFS servers. To do so, perform the following steps:

- Log on as 6VMRAC20 on the first SSI member.
- Issue the following **VMFSETUP** command:

```
==> vmfsetup 6vmrac20 racf
VMFSET2760I VMFSETUP processing started for 6VMRAC20 RACF
VMFUTL2205I Minidisk|Directory Assignments:
            String    Mode  Stat  Vdev  Label/Directory
VMFUTL2205I LOCALSAM  E      R/W  2C2   RAC2C2
VMFUTL2205I APPLY     F      R/W  2A6   RAC2A6
VMFUTL2205I           G      R/W  2A2   RAC2A2
VMFUTL2205I DELTA     H      R/W  2D2   RAC2D2
VMFUTL2205I BUILD0    I      R/W  29E   RAC29E
VMFUTL2205I BUILD6    J      R/W  599   RAC599
VMFUTL2205I BUILD4    K      R/W  505   RAC505
```

```
VMFUTL2205I BUILD2    T      R/W  590   RAC590
VMFUTL2205I BASE      U      R/W  2B2   RAC2B2
VMFUTL2205I --------  A      R/W  191   RAC191
VMFUTL2205I --------  B      R/O  5E5   MNT5E5
VMFUTL2205I --------  D      R/W  51D   MNT51D
VMFUTL2205I --------  S      R/O  190   MNT190
VMFUTL2205I --------  Y/S    R/O  19E   MNT19E
VMFSET2760I VMFSETUP processing completed successfully
```

- Note that many RACF disks have been accessed with the **QUERY DISK** command:

```
==> q disk
LABEL  VDEV M  STAT   CYL TYPE BLKSZ   FILES  BLKS USED-(%) BLKS LEFT  BLK TOTAL
RAC191 191  A   R/W    25 3390 4096        8        153-03       4347       4500
MNT5E5 5E5  B   R/O    18 3390 4096      131       1265-39       1975       3240
MNT51D 51D  D   R/W    26 3390 4096      299       1731-37       2949       4680
RAC2C2 2C2  E   R/W     9 3390 4096        0          7-00       1613       1620
RAC2A6 2A6  F   R/W     9 3390 4096        2          9-01       1611       1620
RAC2A2 2A2  G   R/W     9 3390 4096        0          7-00       1613       1620
RAC2D2 2D2  H   R/W    70 3390 4096        0          7-00      12593      12600
RAC29E 29E  I   R/W    10 3390 4096       60        307-17       1493       1800
RAC599 599  J   R/W    31 3390 4096       44       2506-45       3074       5580
RAC505 505  K   R/W    41 3390 4096      133       5169-70       2211       7380
MNT190 190  S   R/O   207 3390 4096      694      16694-45      20566      37260
RAC590 590  T   R/W    64 3390 4096       48       3853-33       7667      11520
RAC2B2 2B2  U   R/W    85 3390 4096     2351      12260-80       3040      15300
MNT19E 19E  Y/S R/O   500 3390 4096     1125      29764-33      60236      90000
```

- Obtain the latest level of the **RPIBLLPA EXEC** with the following **VMFSIM** command:

```
==> vmfsim getlvl 6vmrac20 racf tdata :part rpibllpa exc (history
:PART RPIBLLPA EXC00000 BASE-FILETYPE
```

The output gives the file name and file type (`RPIBLLPA EXC00000`) that you need to copy to create your new file. You should see the output **BASE-FILETYPE** in the last field. In VMSES/E terminology it means that there has been no service to this part by IBM or locally by a system programmer (no entries in the IBM and Local Version Vector Tables):

- Copy the **RPIBLLPA EXEC** from the `U` disk to the `2C2` (`E`) disk:

```
==> copy rpibllpa exec u = excl0001 e
```

- Edit the newly copied file and comment out the five lines for the `ICHRCX02` member:

```
==> x rpibllpa excl0001 e
====> /ichrcx02
...
*:OBJNAME. ICHRCX02 LEPARMS RENT REUS LET NCAL XREF DCBS SIZE 100K,80K
*:OPTIONS. CONCAT SYSLIB RACFOBJ
*:PARTID. ICHRCX02 TXT
*:OPTIONS. ENTRY ICHRCX02
*:EOBJNAME.
*
:OBJNAME. ICHSFR00 LEPARMS RENT REUS LET NCAL XREF DCBS SIZE 100K,80K
...
====> file
```

- Log this local modification to the `RPIBLLPA EXEC` into the local version vector table with the following **VMFSIM** command:

```
==> vmfsim logmod 6vmrac20 vvtlcl e tdata :mod lcl0001 :part rpibllpa exc
```

- The `2C2` disk (`E`) should now contain `6VMRAC20 VVTLCL` and `RPIBLLPA EXCL0001` files:

```
==> listfile * * e
RPIBLLPA EXCL0001 E2
6VMRAC20 VVTLCL   E1
```

- Type the contents of the `6VMRAC20 VVTLCL` file:

```
==> type 6vmrac20 vvtlcl e

 :PART.RPIBLLPA EXC :MOD.LCL0001
```

- Generate a new `RACFLPA LOADLIB` using the **VMFBLD** command:

```
==> vmfbld ppf 6vmrac20 racf rpibllpa (all)
VMFUTL2767I Reading VMFINS DEFAULTS B for additional options
VMFBLD2760I VMFBLD processing started
VMFBLD1851I Reading build lists
VMFBLD2182I Identifying new build requirements
VMFBLD2182I New build requirements identified
VMFBLD1851I (1 of 1) VMFBDLLB processing RPIBLLPA EXCL0001 E, target is BUILD4
            505 (K)
VMFLLB2217I RACFLPA LOADLIB will be rebuilt because all members must be rebuilt
...
VMFBLD1851I (1 of 1) VMFBDLLB completed with return code 0
VMFBLD2180I There are 0 build requirements remaining
VMFBLD2760I VMFBLD processing completed successfully
```

  Be sure the success message is issued.

- Link the `RACFVM 305` disk read-write and access it as file mode `L`. If you did not change the passwords, the link password will be **multiple**.

```
==> link racfvm 305 305 mr
ENTER MULT PASSWORD:
==> acc 305 l
```

- Use the **VMFCOPY** command to copy the files from the `RACFVM 505` disk (`K`) to the production disk (`L`):

```
==> vmfcopy racflpa * k = = l (prodid 6vmrac20%racf replace oldd
```

- Log off of `6VMRAC20`.

The RACF exit `ICHRCX02` will now be disabled.

18.3.4 Copy the RACF databases

In an SSI, the RACF database must be shared among all members. If you are just installing RACF in a single z/VM LPAR, you can skip this section, which is comprised of the following sub-sections:

- “Copy the RACFVM 200 and 300 minidisks”
- “Change RACFVM to shared disks” on page 292
- “Modify the RACMAINT identity” on page 293
- “Define the shared disks in the SYSTEM CONFIG file” on page 294

Copy the RACFVM 200 and 300 minidisks

To copy the `RACFVM 200` and `300` mindisks to the volumes that will be shared, perform the following steps:

- Logon to the first SSI member as `MAINT`.

Important: The writing of this section was accomplished using two second level z/VM virtual machines, so there was the ability to create 17-cylinder minidisks at the first level that appear as real volumes at the second level. If your SSI is on LPARs at the first level, you must use real volumes for the 200 and 300 RACF database. Use the smallest volumes that you can get.

- Attach the DASD volumes that will be shared. In this example, they are real device addresses 200 and 300:

```
==> q 200 300
DASD 0200 FREE , DASD 0300 FREE
==> att 200 300 *
0200 0300 ATTACHED TO MAINT
```

- Link to the RACFVM 200 and RACFVM 300 disks read-only with the **VMLINK** command:

```
==> vmlink racfvm 200
DMSVML2060I RACFVM 200 linked as 0120 file mode Z
==> vmlink racfvm 300
DMSVML2060I RACFVM 300 linked as 0121 file mode X
```

  Note the virtual device addresses are 120 and 121 respectively

- Copy the RACFVM 200 disk (120) to the 200 volume with the **DDR** command and the following subcommands:

```
==> ddr
z/VM DASD DUMP/RESTORE PROGRAM
ENTER:
====> sysprint cons
ENTER:
====> in 120 3390
ENTER:
====> out 200 3390
ENTER:
====> copy 0 to 16
HCPDDR711D VOLID READ IS RACF
DO YOU WISH TO CONTINUE?  RESPOND YES, NO OR REREAD:
====> yes
ENTER NEXT EXTENT OR NULL LINE
ENTER:
Enter
HCPDDR711D VOLID READ IS VM0200
DO YOU WISH TO CONTINUE?  RESPOND YES, NO OR REREAD:
====> yes
COPYING   RACF
COPYING DATA  06/15/12 AT 15.56.40  GMT FROM RACF   TO VM0200
INPUT CYLINDER EXTENTS      OUTPUT CYLINDER EXTENTS
      START       STOP         START        STOP
          0         16             0          16
END OF COPY
ENTER:
Enter
END OF JOB
```

- Copy the RACFVM 300 disk (121) to the 300 volume with the **DDR** command and the following subcommands:

```
==> ddr
z/VM DASD DUMP/RESTORE PROGRAM
ENTER:
====> sysprint cons
```

```
ENTER:
====> in 121 3390
ENTER:
====> out 300 3390
ENTER:
====> copy 0 to 16
HCPDDR711D VOLID READ IS RACFBK
DO YOU WISH TO CONTINUE?  RESPOND YES, NO OR REREAD:
====> yes
ENTER NEXT EXTENT OR NULL LINE
ENTER:
Enter
HCPDDR716D NO VOL1 LABEL FOUND
DO YOU WISH TO CONTINUE?  RESPOND YES, NO OR REREAD:
====> yes
COPYING   RACFBK
COPYING DATA  06/15/12 AT 15.58.55  GMT FROM RACFBK
INPUT CYLINDER EXTENTS      OUTPUT CYLINDER EXTENTS
      START       STOP         START        STOP
          0         16             0          16
END OF COPY
ENTER:
Enter
END OF JOB
```

The contents of the RACF data sets on the RACFVM 200 and 300 minidisks have now been copied to the real devices (at addresses 200 and 300 in this example).

Change RACFVM to shared disks

Now that the 200 and 300 minidisks from one of the SUBCONFIGs of RACFVM have been copied to the DASD volumes that will be shared, these new disks can replace the individual minidisks. To do this, perform the following steps:

- Get the user directory entry of the RACFVM-1 SUBCONFIG:

```
==> dirm for racfvm-1 get
...
```

- Receive the file from the reader.
- Comment out the 200 and 300 disks:

```
==> x racfvm-1 direct
SUBCONFIG RACFVM-1
 LINK MAINT 0190 0190 RR * CMS system disk
 LINK MAINT 019D 019D RR * help disk
 LINK MAINT 019E 019E RR * Product code disk
 MDISK 191 3390 1931 009 M01W01 MR READ     WRITE    MULTIPLE
* MDISK 200 3390 1914 017 M01W01 MW READ     WRITE    MULTIPLE
 MDISK 490 3390 1940 070 M01W01 MR READ     WRITE    MULTIPLE
 MDISK 305 3390 2010 136 M01W01 MR READ     WRITE    MULTIPLE
* MDISK 300 3390 2146 017 M01W01 MW READ     WRITE    MULTIPLE
 MDISK 301 3390 2163 007 M01W01 MR READ     WRITE    MULTIPLE
 MDISK 302 3390 2170 007 M01W01 MR READ     WRITE    MULTIPLE
```

- Replace the RACFVM-1 SUBCONFIG definition:

```
==> dirm for racfvm-1 rep
...
```

- **Repeat the previous steps** for all other members in the SSI cluster. In this example, only the RACFVM-2 SUBCONFIG also had to be modified.
- Get the user directory entry of the IDENTITY RACFVM:

```
==> dirm for racfvm get
...
```

- Receive the file from the reader.
- Add the following two MDISK entries for 200 and 300:

```
==> x racfvm direct
IDENTITY RACFVM   RACFVM     20M   20M ABCDEGH
 BUILD ON LEFT620 USING SUBCONFIG RACFVM-1
 BUILD ON RIGHT620 USING SUBCONFIG RACFVM-2
* BUILD ON @@member3name USING SUBCONFIG RACFVM-3
* BUILD ON @@member4name USING SUBCONFIG RACFVM-4
 IUCV *RPI PRIORITY MSGLIMIT 100
 IUCV ANY PRIORITY MSGLIMIT 50
 IUCV ALLOW MSGLIMIT 255
 ACCOUNT SYSTEMS
 MACH XA
 IPL 490 PARM AUTOCR
 OPTION QUICKDSP MAXCONN 300
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
* Add 200 and 300 for a shared RACF database
 MDISK 200 3390 DEVNO 0200      MWV READ     WRITE    MULTIPLE
 MDISK 300 3390 DEVNO 0300      MWV READ     WRITE    MULTIPLE
...
```

The DEVNO operand on the MDISK statement specifies a full-pack minidisk, and allows CP to have no dependency on the volume labels of the disks.

- Replace the RACFVM SUBCONFIG definition:

```
==> dirm for racfvm rep
...
 DVHREQ2289I Your REPLACE request for RACFVM at * has completed; with
 DVHREQ2289I RC = 0.
```

Watch for a return code of 0.

The RACFVM virtual machine now references the two shared DASD volumes

Modify the RACMAINT identity

The IDENTITY RACMAINT has link modes to the RACFVM 200 and 300 minidisks of MR. They must be changed to **MW** in order to share the RACF database. To accomplish this, perform the following steps:

- Get the user directory entry of the RACMNT-1 SUBCONFIG:

```
==> dirm for racmnt-1 get
...
```

- Receive the file from the reader.
- For the RACMAINT SUBCONFIGs, change the link modes to the RACFVM 200 and 300 disks to from MR to **MW**. First is the RACMNT-1 SUBCONFIG:

```
==> x racmnt-1 direct
 SUBCONFIG RACMNT-1
  LINK MAINT 0190 0190 RR * CMS system disk
  LINK MAINT 019D 019D RR * help disk
  LINK MAINT 019E 019E RR * Product code disk
  LINK 6VMRAC20 590 490 MR
  LINK 6VMRAC20 505 305 MR
```

```
LINK 6VMRAC20 29E 29E RR
LINK 6VMRAC20 191 192 RR
LINK RACFVM 200 200 MW
LINK RACFVM 300 300 MW
LINK RACFVM 301 301 MR
LINK RACFVM 302 302 MR
```

- Replace the user directory entry:

```
==> dirm for racmnt-1 rep
...
```

- **Repeat the previous steps** for all other members in the SSI cluster. In this example two member SSI cluster, only the RACMNT-2 SUBCONFIG had to be modified.

The RACF database should now be able to be shared on the volumes at real device addresses 200 and 300.

Define the shared disks in the SYSTEM CONFIG file

To define the RACF database DASD to CP as devices that can be shared concurrently between real systems, you must add the RDEVICE statements to the SYSTEM CONFIG file.

To to this perform the following steps:

- Verify you are logged on as MAINT.
- Access the PMAINT CF0 disk read-write. Use the **LINK** command with multi-read (**MR**) parameter:

```
==> link pmaint cf0 cf0 mr
```

- Use the **ACCESS** command to access it as F:

```
==> acc cf0 f
```

- Make a copy of the working SYSTEM CONFIG file:

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

- Edit the original file:

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

- Add two lines at the bottom specifying that the primary and backup RACF database disks are shared:

```
====> bot
====> a 3
...
/* Define RACF primary and backup databases as shared */
rdevice 0200 type dasd shared yes /* RACF primary database */
rdevice 0300 type dasd shared yes /* RACF backup database */
```

- Verify the syntax of the file:

```
==> acc 193 g
==> cpsyntax system config f (lpar pokdev62
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
==> cpsyntax system config f (lpar poktst62
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
```

- Release and detach the PMAINT CF0 (F) disk:

```
==> rel f (det
DASD 0CF0 DETACHED
```

It is also a requirement that CP does not cache data on the RACF database disks in the minidisk cache. Minidisk cache (MDC) is turned off as a result of specifying the DASD as shared in the system configuration file.

The RACF database and backup database should now be shared in the SSI cluster.

18.3.5 Set up the AUTOLOG1 and AUTOLOG2 virtual machines

At z/VM IPL time, the AUTOLOG1 virtual machine normally starts all necessary systems and virtual machines in its **PROFILE EXEC**. When RACF is running, the RACFVM virtual machine must be started first, or other virtual machines will not be able to log in. After the RACF environment is initialized, RACFVM starts the AUTOLOG2 virtual machine, which then starts the remaining servers for the system as AUTOLOG1 normally does. Therefore, the **PROFILE EXEC** needs to be copied from AUTOLOG1 to AUTOLOG2, then modified to start RACFVM.

To accomplish this, perform the following steps:

- Verify you are logged on as MAINT on the first member.
- Link the AUTOLOG1 and AUTOLOG2 191 disks read/write:

```
==> link autolog1 191 1191 mr
==> link autolog2 191 2191 mr
```

- Access the two disks as file modes F and G:

```
==> acc 1191 f
==> acc 2191 g
```

- Copy the PROFILE EXEC from AUTOLOG1 to AUTOLOG2:

```
==> copy profile exec f = = g
```

- Edit the PROFILE EXEC on the AUTOLOG1 191 disk and update it as follows to start RACFVM:

```
==> x profile exec f
/*******************************************************************/
/*  AUTOLOG1 PROFILE EXEC                                          */
/*******************************************************************/
Address Command
"CP XAUTOLOG RACFVM"
"CP LOGOFF"
====> file
```

- **Perform the steps in the section** on all other SSI members in the cluster.

The AUTOLOG1 virtual machine should now be configured start RACF (the RACFVM virtual machine). RACF will then start AUTOLOG2 and complete the bootstrapping of the z/VM system.

18.3.6 Enable RACF

To enable RACF, perform the following steps:

- **Shut down all other members** except the first SSI node. In this example, SSI member 2 was shut down:

```
==> shutdown
...
```

- Log on to MAINT620 on the first SSI member.
- Issue the following **SERVICE** command to enable RACF. This step needs to be performed on only one member. A number of screens will pass by:

```
==> service racf enable
...
VMFSRV1233I The following products have been serviced.
VMFSRV1233I CP RACF
VMFSRV2760I SERVICE processing completed successfully
```

RACF should now be enabled on the `CF2` disk. This disk is now on the release volume 1 volume in z/VM 6.2.

- Shutdown the first SSI member:

```
==> shutdown
...
```

RACF should now be enabled and all members in the SSI should be shut down.

18.3.7 Put RACF into production on all members

The **PUT2PROD** command must be run on each member of the SSI. Start with the first member. Perform all five of the following sub-sections on the first member (if you are in an SSI, you will later perform the first and last sub-section on the other members):

- "IPL the member and start RACMAINT"
- "Configure the initial RACF database"
- "Enable DirMaint to RACF on the first member"
- "Set DirMaint's use of the reader with RACF on the first member"
- "Put RACF into production on all members"

IPL the member and start RACMAINT

- IPL the next member from the HMC. At the SAPL screen, change the *Device Number* to that of the Release Volume 1. In this example it was real device address ***5704***. Press `F10` to IPL. This will load the **CPLOAD MODULE** from the `CF2` disk which has RACF enabled.

```
STAND ALONE PROGRAM LOADER: z/VM VERSION 6 RELEASE 2.0

 DEVICE NUMBER:   5704       MINIDISK OFFSET:   39         EXTENT:  1

 MODULE NAME:     CPLOAD     LOAD ORIGIN:       1000

 --------------------------------IPL PARAMETERS--------------------------------
fn=SYSTEM ft=CONFIG pdnum=1 pdvol=5703


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




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



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

- Supply the `NOAUTOLOG` parameter so that the PROFILE EXEC on AUTOLOG1 is not run and RACFVM is not started:

```
16:30:25 Start ((Warm|Force|COLD|CLEAN) (DRain) (DIsable)  (NODIRect)
16:30:25       (NOAUTOlog)) or (SHUTDOWN)
noautolog
```

```
...
```

- Continue to IPL the member. When the IPL process completes, you will be logged on as `OPERATOR`. Start the virtual machine `RACMAINT`. You should messages that the `200` and `300` disks are read-write. If you see errors about them, you have to fix the problem.

```
==> xautolog racmaint
15:27:40 Command accepted
15:27:40 AUTO LOGON  ***       RACMAINT USERS = 2     BY OPERATOR
15:27:40 HCPCLS6056I XAUTOLOG information for RACMAINT: The IPL command is verif
ied by the IPL command processor.
RACMAINT: RACFVM CMS XA Rel 14 11/18/2010
RACMAINT: DMSACP723I B (305) R/O
RACMAINT: RACF is defined to the Z/VM system and the current product status is E
NABLED
RACMAINT:
RACMAINT:       RACF
RACMAINT:  Feature for z/VM
RACMAINT:  Version 6.2.0
RACMAINT:
RACMAINT:  Licensed Materials - Property of IBM
RACMAINT:  5741-A07
RACMAINT:  (C) Copyright IBM CORP. 1981, 2010  All Rights Reserved.
RACMAINT:
RACMAINT: DMSACC723I R (0200) R/W - OS
RACMAINT: DMSACC723I Q (0300) R/W - OS
RACMAINT: CSTSET001I CMS SUB-TASKING SUPERVISOR INITIALIZED.
...
```

RACF should now be running on the SSI member with a skeleton database.

If you have already done the next three sections on the first SSI member, proceed to the section "Put RACF into production on all members" on page 300.

Configure the initial RACF database

The following set of steps need to be performed only once to populate and customize the RACF database.

- **On the first SSI member**, disconnect from `OPERATOR`.

```
==> disc
```

- log on to `IBMUSER` with a password of **SYS1**. This is a default virtual machine created for RACF configuration.
- You will see a message that the password has expired. Reset the password by typing in the new password twice separated by a "/". You will see resource errors - these are expected:

```
LOGON IBMUSER
RPIMGR042I PASSWORD EXPIRED

To change your password - enter: nnn/nnn  where nnn = new password
or,
enter LOGOFF to cancel

 ICH70001I IBMUSER  LAST ACCESS AT **:**:** ON ****, **** **,****
 HCPRPW004I Password changed
RPIMGR031E RESOURCE MAINT.190 SPECIFIED BY LINK COMMAND NOT FOUND
RPIMGR031E RESOURCE MAINT.19E SPECIFIED BY LINK COMMAND NOT FOUND
RPIMGR031E RESOURCE 6VMRAC20.29E SPECIFIED BY LINK COMMAND NOT FOUND
RPIMGR031E RESOURCE 6VMRAC20.505 SPECIFIED BY LINK COMMAND NOT FOUND
```

```
RPIMGR031E RESOURCE 6VMRAC20.191 SPECIFIED BY LINK COMMAND NOT FOUND
RPIMGR031E RESOURCE RACFVM.305 SPECIFIED BY LINK COMMAND NOT FOUND
RPIMGR031E RESOURCE IBMUSER.191 SPECIFIED BY LINK COMMAND NOT FOUND
z/VM Version 6 Release 2.0, Service Level 1101 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 13:24:34 EDT FRIDAY 06/22/12
z/VM V6.2.0    2012-06-21 16:54
...
```

- Set the F12 function key to the command **RETRIEVE**:

```
==> set pf12 ret
```

- Link and access 6VMRAC20's 505, 191, and 29E disks. Disregard any error messages.

```
==> link 6vmrac20 505 505 rr
RPIMGR031E RESOURCE 6VMRAC20.505 SPECIFIED BY LINK COMMAND NOT FOUND
DASD 0505 LINKED R/O; R/W BY RACMAINT
==> acc 505 c
DMSACP723I C (505) R/O
==> link 6vmrac20 191 192 rr
RPIMGR031E RESOURCE 6VMRAC20.191 SPECIFIED BY LINK COMMAND NOT FOUND
==> acc 192 b
DMSACP723I B (192) R/O
DMSACP725I 192 also = D disk
==> link 6vmrac20 29e 29e rr
RPIMGR031E RESOURCE 6VMRAC20.29E SPECIFIED BY LINK COMMAND NOT FOUND
==> acc 29e d
DMSACP724I 29E replaces D (192) R/O
DMSACP723I D (29E) R/O
```

- Update the RACF database with existing CP directory information using the **RPIBLDDS** command. The RPIDIRCT SYSUT1 file created earlier and copied to the 6VMRAC20 191 disk is used as input. You may again choose to issue the command **#CP TERM MORE 0 0** as many screens full of messages will be issued:

```
==> rpibldds rpidirct
Processing batch file RPIDIRCT SYSUT1 using “RAC” command interface
...
 => PERMIT LOGONBY.SSLDCSSM CLASS(SURROGAT) ID(TCPMAINT) ACCESS(READ)
 => PERMIT LOGONBY.SSLDCSSM CLASS(SURROGAT) ID(GSKADMIN) ACCESS(READ)
 => setropts generic(vmbatch) gencmd(vmbatch)
 => rdefine vmbatch ** uacc(none)
 => permit ** class(vmbatch) id(ftpserve vmnfs dirmsat dirmsat2) acc(control)
 => setropts classact(vmbatch vmmdisk vmcmd vmlan surrogat)
```

The RACF database should now be populated with the values from the user directory, and other modifications configured previously.

- Define the security administrator virtual machine. In this example the default of SYSADMIN is used.

```
==> rac alu sysadmin special
```

- Log off of IBMUSER
- Log on to SYSADMIN. You will be asked to change the password.
- Grant the following virtual machines OPERATIONS privileges:

```
==> rac alu datamove operations
==> rac alu maint620 operations
==> rac alu bldseg operations
==> rac alu lnxadmin operations
```

These commands give the four virtual machines specified access to all minidisks on the system.

- Revoke the privileges for the `IBMUSER` virtual machine as it is no longer needed:

```
==> rac alu ibmuser revoke
```

- Grant the `DIRMAINT` virtual machine `SPECIAL` privileges:

```
==> rac alu dirmaint special
```

- Grant the `MAINT` virtual machine `SPECIAL` and `OPERATIONS` privileges:

```
==> rac alu maint special operations
```

- Define the system virtual switches named `VSW1` and `VSW2` to the `VMLAN` class:

```
==> rac rdefine vmlan system.vsw1
==> rac rdefine vmlan system.vsw2
```

- Permit `TCPIP` to the virtual switch `VSW1`:

```
==> rac permit system.vsw1 class(vmlan) id(tcpip) access(update)
```

- Log off of `SYSADMIN`.

The initial RACF database should now be configured.

Enable DirMaint to RACF on the first member

Perform the following steps to enable DirMaint to run to RACF:

- Log on to `MAINT`. You should be asked to change the password.
- Link to the `6VMDIR20 2C2` disk read-only which has a sample `CONFIGRC DATADVH` file:

```
==> vmlink 6vmdir20 2c2
DMSVML2060I 6VMDIR20 2C2 linked as 0120 file mode Z
```

- Copy the sample `CONFIGRC` file from the `Z` disk to the `A` disk as file type `DATADVH`:

```
==> copy configrc sampdvh z = datadvh a
```

- Start DirMaint with the **`XAUTOLOG DIRMAINT`** command:

```
==> xautolog dirmaint
 ICH70001I DIRMAINT LAST ACCESS AT 15:38:05 ON WEDNESDAY, JUNE 20, 2012
Command accepted
Ready; T=0.01/0.01 15:50:02
AUTO LOGON  ***       DIRMAINT USERS = 5
HCPCLS6056I XAUTOLOG information for DIRMAINT: The IPL command is verified by th
e IPL command processor.
 DVHPRO2008I ROLE = DIRMAINT
```

- Add the `CONFIGRC DATADVH` configuration file to DirMaint with the **`DIRM FILE`** command. You can ignore error messages such as the `RPIMGR031E` message shown below:

```
==> dirm file configrc datadvh
RPIMGR031E RESOURCE DIRMAINT SPECIFIED BY SPOOL COMMAND NOT FOUND
RPIMGR031E RESOURCE POKDEV62 SPECIFIED BY TAG COMMAND NOT FOUND
PUN FILE 0011 SENT TO   DIRMAINT RDR AS  0004 RECS 0103 CPY  001 0 NOHOLD NOKEEP
DVHXMT1191I Your FILE request has been sent for processing to DIRMAINT
DVHXMT1191I at POKDEV62.
 DVHREQ2288I Your FILE request for MAINT at * has been accepted.
 DVHRCV3821I File CONFIGRC DATADVH A2 has been received; RC = 0.
 DVHREQ2289I Your FILE request for MAINT at * has completed; with RC = 0.
```

- Issue the **`DIRM RLDDATA`** command so the change is activated:

```
==> dirm rldd
DVHXMT1191I Your RLDDATA request has been sent for processing to
```

```
DVHXMT1191I DIRMAINT at POKDEV62.
DVHREQ2288I Your RLDDATA request for MAINT at * has been accepted.
 DVHITI6314E No DATAMOVE machines were defined in the config file.
 DVHREQ2289I Your RLDDATA request for MAINT at * has completed; with RC =
 DVHREQ2289I 0.
```

DirMaint should now be initially enabled to RACF.

Set DirMaint's use of the reader with RACF on the first member

Because the VMBATCH definitions were deleted in section 18.3.1, "Create the RACF command file" on page 285, RACF will give errors when DirMaint sends files to the reader. To address this issue, the CP **TRANSFER** and **TAG** commands need not be controlled.

In addition, SMAPI needs to issue commands for other users with the **FOR** command under privilege class C. To address this, the CP **FOR.C** commands need *not* be controlled.

To effect these settings, perform the following steps:

- Create a RACF profile for the VMXEVENT class named EVENT1:

```
==> rac rdefine vmxevent event1
```

- Add three members to the VMEVENT class for the **TRANSFER** (privilege class G), the **TAG** commands and for the **FOR** (privilege class C), and set them to no-control:

```
==> rac ralter vmxevent event1 addmem(transfer.g/noctl tag/noctl for.c/noctl)
```

- Activate the VMXEVENT class:

```
==> rac setropts classact(vmxevent)
```

- Refresh the VMEVENT class:

```
==> rac setevent refresh event1
RPISET113W TURNING CONTROL ON AUTOMATICALLY FOR: COUPLE
RPISET113W TURNING CONTROL ON AUTOMATICALLY FOR: FOR.G
RPISET113W TURNING CONTROL ON AUTOMATICALLY FOR: LINK
...
```

- Log off of MAINT.

DirMaint and SMAPI should now be enabled to run with RACF.

Put RACF into production on all members

RACF is now set to be in production on Put RACF into production with the following steps:

- From OPERATOR issue the **XAUTOLOG RACMAINT** command
- Disconnect from OPERATOR:

```
==> disc
...
```

- Log on to MAINT620 on the next member. You will be asked to change the password.
- Start the AUTOLOG2 virtual machine with the **XAUTOLOG** command to start the shared file pool server machines.

```
==> xautolog autolog2
 ICH70001I AUTOLOG2 LAST ACCESS AT **:**:** ON ****, **** **,****
Command accepted
AUTO LOGON  ***       AUTOLOG1 USERS = 5
HCPCLS6056I XAUTOLOG information for AUTOLOG1: The IPL command is verified by th
e IPL command processor.
```

- Put RACF into production with the **PUT2PROD RACF** command. Watch for the completed successfully message:

```
==> put2prod racf
RDR FILE 0049 SENT FROM MAINT620 CON WAS 0049 RECS 0018 CPY  001 T NOHOLD NOKEEP
VMFUTL2767I Reading VMFINS DEFAULTS B for additional options
VMFP2P2760I PUT2PROD processing started
VMFP2P2204I Linking PMAINT 41D with link mode M
DMSVML2060I PMAINT 41D linked M as 041D file mode Z
VMFP2P2760I PUT2PROD processing started for RACF
VMFSET2760I VMFSETUP processing started for SERVP2P RACFP2P
VMFSET2204I Linking 6VMRAC20 2C2 as 2C2 with link mode MR
RPIMGR031E RESOURCE 6VMRAC20.2C2 SPECIFIED BY LINK COMMAND NOT FOUND
VMFSET2204I Linking 6VMRAC20 2D2 as 2D2 with link mode MR
RPIMGR031E RESOURCE 6VMRAC20.2D2 SPECIFIED BY LINK COMMAND NOT FOUND
...
VMFP2P2760I PUT2PROD processing completed successfully
```

- Put CP into production with the **PUT2PROD CP** command. Watch for the completed successfully message:

```
==> put2prod cp
...                         // a number of screens pass by
VMFP2P2760I PUT2PROD processing completed successfully
```

  RACF should now be prepared to go into production at the next IPL.

- Log off of MAINT620.
- Log on to OPERATOR. You will be asked to change the password.
- Log the RACMAINT virtual machine off with the **FORCE** command:

```
==> force racmaint
RACMAINT: CONNECT= 00:37:57 VIRTCPU= 000:03.32 TOTCPU= 000:04.03
RACMAINT: LOGOFF AT 16:11:53 EDT WEDNESDAY 06/20/12 BY OPERATOR
16:11:53 USER DSC   LOGOFF AS  RACMAINT USERS = 22    FORCED BY OPERATOR
16:11:53 HCPRPI036E CP/RACF communication path broken to RACMAINT
```

- Start the RACFVM virtual machine with the **XAUTOLOG** command and watch for messages that RACF is starting:

```
==> xautolog racfvm
14:42:39 Command accepted
14:42:39 AUTO LOGON  ***       RACFVM   USERS = 23    BY OPERATOR
16:12:00 HCPCLS6056I XAUTOLOG information for RACFVM: The IPL command is verifie
d by the IPL command processor.
RACFVM  : RACFVM CMS XA Rel 14 11/18/2010
RACFVM  : DMSACP723I B (305) R/O
RACFVM  : RACF is defined to the Z/VM system and the current product status is E
NABLED
RACFVM  :
RACFVM  :        RACF
RACFVM  :  Feature for z/VM
RACFVM  :  Version 6.2.0
RACFVM  :
RACFVM  :  Licensed Materials - Property of IBM
RACFVM  :  5741-A07
RACFVM  :  (C) Copyright IBM CORP. 1981, 2010  All Rights Reserved.
RACFVM  :
RACFVM  : DMSACC723I R (0200) R/W - OS
RACFVM  : DMSACC723I Q (0300) R/W - OS
...
16:12:02 HCPRPI035I CP/RACF communication path established to RACFVM
...
```

RACF should now be running on the current member.

- Shut down the member:

```
==> shutdown
...
00: 13:52:25 HCPWRP961W SYSTEM SHUTDOWN COMPLETE FOR LEFT620  ON 2012-06-22
00: HCPGIR450W CP entered; disabled wait PSW 00020000 00000000 00000000 00000961
```

- **For SSI members other than the first**, perform the steps in only the first and last of the five subsections:
  - "IPL the member and start RACMAINT" on page 296
  - "Put RACF into production on all members" on page 300
- Once the **PUT2PROD** sections have been performed on all SSI members, IPL the members one at a time from the default (*RES*) volume. Do not specify the NOAUTOLOG parameter. You should see RACF start on the OPERATOR console.

When the system comes back up, RACF should be running.

18.3.8 Configure SMAPI to work with RACF

Perform the following steps to allow SMAPI to work with RACF:

- Logon to MAINT on the first SSI member.
- Allow VSMWORK1 to have CONTROL authority the z/VM minidisk (VMMDISK) that contains the SYSTEM CONFIG file (PMAINT CF0), perform the following command:

```
==> rac permit pmaint.cf0 class(vmmdisk) acc(control) id(vsmwork1)
==> rac permit maint.cf1 class(vmmdisk) acc(control) id(vsmwork1)
```

- Allow VSMWORK1 to have CONTROL access to the generic class:

```
==> rac permit ** class(vmbatch) id(vsmwork1) access(control)
```

- Allow SMAPI workers to read the TCPMAINT 198 disk:

```
==> rac permit tcpmaint.198 class(vmmdisk) acc(read) id(vsmguard)
==> rac permit tcpmaint.198 class(vmmdisk) acc(read) id(vsmwork1)
==> rac permit tcpmaint.198 class(vmmdisk) acc(read) id(vsmwork2)
==> rac permit tcpmaint.198 class(vmmdisk) acc(read) id(vsmwork3)
```

- Allow LNXADMIN to read certain disks:

```
==> rac permit pmaint.cf0 class(vmmdisk) acc(read) id(lnxadmin)
==> rac permit autolog1.191 class(vmmdisk) acc(read) id(lnxadmin)
==> rac permit tcpmaint.198 class(vmmdisk) acc(read) id(lnxadmin)
```

- Change default password expiration to 186 days:

```
==> rac setropts password(interval(186))
```

SMAPI should now work with RACF.

18.4 Getting DirMaint and RACF to work together

To add new virtual machines, some DirMaint and some RACF commands are needed: Perform the following steps:

- Log in as MAINT.
- Create a sample virtual machine prototype named LNXSAMPL PROTODIR:

```
==> x lnxsampl protodir a
USER LNXSAMPL LNX4VM 256M 2G G
    INCLUDE LNXDFLT
    MDISK 0100 3390 AUTOG 10016 POOL1 MR LNX4VM LNX4VM LNX4VM
    MDISK 0101 3390 AUTOG 10016 POOL1 MR LNX4VM LNX4VM LNX4VM
```

This definition will give each Linux virtual machine 256 MB of initial memory (with up to 2GB dynamic) and 2 3390-9 disks or about 14 GB of disk space. The **AUTOG** and **POOL1** keywords instruct DirMaint to automatically choose space from the pool of volumes in the pool named POOL1.

- Register the prototype with DirMaint using the **DIRM FILE** command:

```
==> dirm file lnxsampl protodir
10:08:53 PUN FILE 0069 SENT TO   DIRMAINT RDR AS  0086 RECS 0012 CPY  001 0 NOHO
LD NOKEEP
DVHXMT1191I Your FILE request has been sent for processing to DIRMAINT
DVHXMT1191I at POKDEV62.
 DVHREQ2288I Your FILE request for MAINT at * has been accepted.
 DVHRCV3821I File LNXSAMPL PROTODIR A has been received; RC = 0.
 DVHREQ2289I Your FILE request for MAINT at * has completed; with RC = 0.
```

- Create a new virtual machine with the **DIRM ADD** command and the **LIKE** parameter. In this example, the user ID is named LINUX76:

```
==> dirm add linux76 like lnxsampl pw lnx4vm
DVHXMT1191I Your ADD request has been sent for processing to DIRMAINT at
DVHXMT1191I POKDEV62.

 DVHREQ2288I Your ADD request for LINUX76 at * has been accepted.
 ...
 DVHSHN3430I AMDISK operation for LINUX76 address 0101 has finished (WUCF
 DVHSHN3430I 07101436).
 DVHREQ2289I Your ADD request for LINUX76 at * has completed; with RC =
 DVHREQ2289I 0.
```

- Allow the new user access to the virtual switches named VSW1 and VSW2:

```
==> rac permit system.vsw1 class(vmlan) id(linux76) access(update)
==> rac permit system.vsw2 class(vmlan) id(linux76) access(update)
```

This shows an example of DirMaint working with RACF when creating new virtual machines.

18.5 Some common DirMaint tasks

The following common DirMaint tasks are described in the sections that follow:

- “Update a user directory entry”
- “Edit the EXTENT CONTROL file” on page 304
- “Get a copy of the user directory” on page 306
- “Add an IDENTITY” on page 306
- “Clean up work units” on page 307

18.5.1 Update a user directory entry

There are many ways to modify virtual machine or PROFILE directory entries with DirMaint. Perhaps one of the simplest is with GET and REPLACE. The **GET** operand of the **DIRMAINT** command retrieves a copy of a user or profile directory entry. You can then introduce new control statements or modify existing ones. The updated directory entry can be restored to the directory using the **REPLACE** operand. Following is an example of modifying the directory entry for the virtual machine LINUX153:

- Log on to MAINT.
- Use the **GET** operand to unlock the directory entry.

```
==> dirm for linux153 get
DVHXMT1191I Your GET request has been sent for processing to DIRMAINT at
DVHXMT1191I POKDEV62.
Ready; T=0.01/0.01 10:14:45
 DVHREQ2288I Your GET request for LINUX153 at * has been accepted.
 DVHGET3304I Directory entry  LINUX153 is now locked.
 RDR FILE 0005 SENT FROM DIRMAINT PUN WAS 0029 RECS 0010 CPY  001 A NOHOLD NOKEEP
 DVHREQ2289I Your GET request for LINUX153 at * has completed; with RC
 DVHREQ2289I = 0.
```

Always watch for a 0 return code. Note that the file sent is number **5**.

- Receive the directory entry from the reader to MAINT's A disk with the **REPLACE** option:

```
==> receive 5 (rep
File LINUX153 DIRECT A0 replaced by LINUX153 DIRECT A0 received from DIRMAINT at
 POKDEV62
```

- Edit the directory entry and make the desired changes:

```
==> x linux153 direct
...
```

- Use the **REPLACE** operand to make the changes effective:

```
==> dirm for linux153 rep
PUN FILE 0006 SENT TO   DIRMAINT RDR AS  0033 RECS 0013 CPY  001 0 NOHOLD NOKEEP
DVHXMT1191I Your REPLACE request has been sent for processing to
DVHXMT1191I DIRMAINT at POKDEV62.
Ready; T=0.01/0.01 10:14:51
 DVHREQ2288I Your REPLACE request for LINUX153 at * has been accepted.
 ...
 DVHBIU3428I online.
 DVHREP3603I Directory entry LINUX153 is now unlocked.
 DVHREQ2289I Your REPLACE request for LINUX153 at * has completed; with
 DVHREQ2289I RC = 0.
```

This has shown how to modify a virtual machine or profile directory entry.

18.5.2 Edit the EXTENT CONTROL file

If you want to add disks to virtual machines, you should first let DirMaint know about them by updating the EXTENT CONTROL file. To do so, perform the following steps:

- Log on to any SSI member as MAINT.
- Invoke the command for DirMaint to send the current EXTENT CONTROL file:

```
==> dirm send extent control
DVHXMT1191I Your SEND request has been sent for processing to DIRMAINT
DVHXMT1191I at POKDEV62.
Ready; T=0.01/0.01 10:29:42
 DVHREQ2288I Your SEND request for MAINT at * has been accepted.
RDR FILE 0008 SENT FROM DIRMAINT PUN WAS 0049 RECS 0070 CPY  001 A NOHOLD NOKEEP
 DVHREQ2289I Your SEND request for MAINT at * has completed; with RC = 0.
```

- Receive the file from MAINT's reader:

```
==> receive 8 (rep
```

```
File EXTENT CONTROL A1 replaced by EXTENT CONTROL E1 received from DIRMAINT at P
OKDEV62
```

- Edit the file, adding disks to the `REGIONS` section. In the example that follows two volumes are added, `JM63AB` and `JM63AC`, The `RegionID`, field 1, is set to the same value as the `VolSer` (label), field 2. Fields 3 and 4 set the cylinder range to all cylinders except cylinder 0, and the last field informs DirMaint as to the size of the disk.

```
==> x extent control
...
:REGIONS.
*RegionId  VolSer    RegStart      RegEnd  Dev-Type  Comments
JM61A5     JM61A5    0001          END     3390-09
JM61B2     JM61B2    0001          END     3390-09
JM6289     JM6289    0001          END     3390-03
JM628C     JM628C    0001          END     3390-03
JM628D     JM628D    0001          END     3390-03
JM628E     JM628E    0001          END     3390-03
JM628F     JM628F    0001          END     3390-03
JM6290     JM6290    0001          END     3390-03
JM6293     JM6293    0001          END     3390-03
JM6294     JM6294    0001          END     3390-03
JM6327     JM6327    0001          END     3390-03
JM6328     JM6328    0001          END     3390-03
JM6339     JM6339    0001          END     3390-03
JM633A     JM633A    0001          END     3390-03
JM633B     JM633B    0001          END     3390-03
JM633C     JM633C    0001          END     3390-03
:END.
:GROUPS.

JMD9AB    JMD9AB           1       10016  3390-09

:END.
:GROUPS.
...
```

- Invoke the command for DirMaint to replace the current `EXTENT CONTROL` file:

```
==> dirm file extent control
PUN FILE 0009 SENT TO   DIRMAINT RDR AS  0053 RECS 0074 CPY  001 0 NOHOLD NOKEEP
DVHXMT1191I Your FILE request has been sent for processing to DIRMAINT
DVHXMT1191I at POKDEV62.
Ready; T=0.01/0.01 10:32:38
 DVHREQ2288I Your FILE request for MAINT at * has been accepted.
 DVHRCV3821I File EXTENT CONTROL E1 has been received; RC = 0.
 DVHREQ2289I Your FILE request for MAINT at * has completed; with RC = 0.
```

Always watch for a return code of 0.

- Invoke the command for DirMaint to reload the `EXTENT CONTROL` file in the current session:

```
==> dirm rlde
DVHXMT1191I Your RLDEXTN request has been sent for processing to
DVHXMT1191I DIRMAINT at POKDEV62.
Ready; T=0.01/0.01 10:34:04
 DVHREQ2288I Your RLDEXTN request for MAINT at * has been accepted.
 DVHILZ3510I Starting DVHINITL with directory: USER DIRECT E
 DVHILZ3510I DVHINITL Parms: BLDMONO BLDDASD BLDLINK
 DVHIZD3528W One or more DASD volume control files (CVD964) were
 DVHIZD3528W created using default values for device characteristics -
 DVHIZD3528W $ALLOC$ 0A00
 DVHREQ2289I Your RLDEXTN request for MAINT at * has completed; with RC =
```

```
DVHREQ2289I 0.
```

Always watch for a return code of 0.

This section has shown how to update the DirMaint EXTENT CONTROL file.

18.5.3 Get a copy of the user directory

Sometimes it is convenient to get a copy of the entire user directory similar to editing the USER DIRECT file. To do so, perform the following steps:

- Log on to any SSI member as MAINT.
- Use the **USER WITHPASS** operand of DirMaint to send a copy of the user directory to MAINT's reader:

```
==> dirm user withpass
DVHXMT1191I Your USER request has been sent for processing to DIRMAINT
DVHXMT1191I at POKDEV62.
Ready; T=0.01/0.01 10:57:20
 DVHREQ2288I Your USER request for MAINT at * has been accepted.
RDR FILE 0010 SENT FROM DIRMAINT PUN WAS 0057 RECS 5441 CPY  001 A NOHOLD NOKEEP
 DVHREQ2289I Your USER request for MAINT at * has completed; with RC = 0
```

Watch for a 0 return code. Also, note the number of the reader file - **10** in this example.

- Receive the file with the **REPLACE** option:

```
==> receive 10 (rep
File USER WITHPASS A0 replaced by USER WITHPASS A0 received from DIRMAINT at POK
DEV62
```

- Browse the user directory file, USER WITHPASS A.

```
==> browse user withpass
...
```

This section has shown how to get a copy of the current user directory.

18.5.4 Add an IDENTITY

Creating an IDENTITY (MCVM) is new to z/VM 6.2. Following is an example of defining a new MCVM that can be logged on to both SSI members at the same time:

- Create a file with a file name corresponding to the IDENTITY and a file type of DIRECT. The IDENTITY is added without the BUILD statements because DirMaint will add the BUILD statements to the IDENTITY automatically when the SUBCONFIG is added.

```
==> x zmapvm62 direct
IDENTITY ZMAPVM62 DRCT 512M 1G BDEG
 INCLUDE LNXDFLT
 OPTION LNKNOPAS LANG AMENG
```

- Add the IDENTITY to the user directory with the ADD operand:

```
==> dirm add zmapvm62
...
```

- Create a file with a file name corresponding to the SUBCONFIG of the first member and a file type of DIRECT:

```
==> type zmapvm-1 direct
SUBCONFIG ZMAPVM-1
 MDISK 0100 3390 0001 10016 JM61BE MR LNX4VM LNX4VM LNX4VM
 MDISK 0101 3390 0001 3338  JM6134 MR LNX4VM LNX4VM LNX4VM
```

```
MDISK 0102 3390 0001 3338  JM613F MR LNX4VM LNX4VM LNX4VM
MDISK 0103 3390 0001 3338  JM6140 MR LNX4VM LNX4VM LNX4VM
```

- Add the SUBCONFIG to the user directory with the ADD operand and the following parameters:

```
==> dirm add zmapvm-1 build on pokdev62 in zmapvm62
...
```

- Repeat the previous two steps for each additional member in the SSI.

This section has shown how to add a MCVM using DirMaint.

18.5.5 Clean up work units

Sometimes when you try to delete minidisks or entire virtual machines, there is an error because a disk to be deleted is linked or accessed. When this happens, failed work units can be the result. Before the deletion can be completed, the failed work units must be cleaned up.

To clean up failed work units, perform the following steps:

- List all work units:

```
==> dirm status workunit all
```

- List the specific work unit you want to delete. In this example, the work unit 20024211 is used:

```
==> dirm status workunit 20024211
```

- Delete a specific work unit:

```
==> dirm workunit 20024211 cancel
```

Also, when you can't get anywhere and want to start over, try:

```
==> dirm for datamove cleanup cancel
```

19

Monitor and tune z/VM and Linux

"Not everything that can be counted counts, and not everything that counts can be counted."

--Albert Einstein

This chapter briefly describes how to monitor z/VM and Linux. For another source on z/VM performance and monitoring, see Chapter 11, *Monitoring performance and capacity*, in the Manual *Getting Started With Linux*, SC24-6096 on the Web at:

`http://publib.boulder.ibm.com/cgi-bin/bookmgr/DOWNLOAD/hcsx0c10.pdf`

There are a number of z/VM monitoring tools such as CA's VM:Monitor, IBM's z/VM Performance Toolkit, IBM's Tivoli OMEGAMON XE for z/VM and Linux. and products from Velocity Software.The IBM z/VM Performance Toolkit is briefly described in this section.

This chapter contains the following sections:

19.1 Use basic z/VM 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. For more information, see the *z/VM Performance Resources* Web page on the Web at:

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

19.1.1 Use the INDICATE command

z/VM has some basic commands such as **`INDICATE`**. 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 virtual machine has privilege class G or class E. Class G users can use **INDICATE** to display recent contention for system resources, display environment characteristics and measurements of resources used by their virtual machine.

The output from virtual machines with class E privilege (e.g. MAINT, OPERATOR) is shown here. The lines are number for clarity of the description that follows:

```
==> 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 **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:

- AVGPROC on line **1** gives the overall processor utilization, 38% in this example. The number following it is the number of on-line processors, 3 in this example. The individual processor utilization is shown on lines **10** and **11**. Take a glance at these to see if they are somewhat balanced. There are cases where an imbalance is okay. 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. Line **2** describes paging to expanded storage. Most z/VM systems on z9 class machines can sustain several 1000s of this type of paging operations a second without any problems. z10 class machines will perform even better. 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 100s rather than 1000s. The higher values seen tend to build up over time, and are sustained over periods of intense system activity, however, there are times the MIGRATE value may spike for brief periods of time.
- (MDC) statistics are given on the third line. 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 which 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, this author has seen the avoidance as high as 50% in some cases. Conversely, however, a high HIT RATIO with a low value for the READS rate should not be taken as good, (100% hit ratio, when doing only 1 I/O per second is effectively meaningless).
- Line **4** describes more storage (memory) management. 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. Since 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 3 different classes (1 through 3) and a special additional class labelled zero. So the Column of Q_x values and E_x 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 hardware addition in extreme cases. Do not worry about the values in parenthesis.

INDICATE QUEUES EXP

Another useful command to understand the state of the system is the **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 displays the virtual processors associated with a given virtual machine (that can have multiple virtual processors) what queue (dispatch list, eligible list, limit list) they are in and what state they are in. This is a snapshot in time. Again you want to 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 also gives state of virtual processor. This can be helpful to get a idea of how the virtual processors might be constrained. Virtual processors that are actually running in the snapshot period are marked with and 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 not an available processor. (**Note**: the virtual machine that issues the **INDICATE** command will always be one of the running machines). Other states are documented in the help for **IND Q EXP**. One doesn't have to be concerned about the other columns unless detailed analysis is required or if IBM support requests it. Also, always remember that is just a snapshot in time so often repeating this command over time can give a more accurate picture of your z/VM system, a single snapshot cannot be regarded as indicative.

19.1.2 Use other basic commands

Some other useful basic commands are briefly mentioned. All examples are shown from the MAINT virtual machine. The results will be different for users with fewer privileges.

Getting help

To get help on the system use the **HELP** command. Sometimes it's hard to find help for exactly the command you're looking for. Some useful help commands are as follow

```
==> 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
DIRMSAT2 - SSI
ZMAPVM62 - DSC , LINUX153 - DSC , LNXADMIN - DSC , LINUX157 - DSC
VSMEVSRV - DSC , VSMPROXY - DSC , VSMREQIU - DSC , VSMREQI6 - DSC
VSMREQIN - DSC , DTCSMAPI - DSC , PERSMAPI - DSC , VSMWORK3 - DSC
VSMWORK2 - DSC , VSMWORK1 - DSC , FTPSERVE - DSC , VSMGUARD - DSC
TCPIP    - DSC , DIRMAINT - DSC , DTCVSW2  - DSC , DTCVSW1  - DSC
VMSERVP  - DSC , VMSERVR  - DSC , VMSERVU  - DSC , VMSERVS  - DSC
OPERSYMP - DSC , DISKACNT - DSC , EREP     - DSC , OPERATOR - DSC
MAINT    -L0004
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 = 16G CONFIGURED = 16G 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
```

This shows that there is 16 GB of central memory (storage) and 2 GB of expanded.

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 ALTERNATE CP
PROCESSOR 05 ALTERNATE CP
PROCESSOR 06 ALTERNATE CP
PROCESSOR 07 ALTERNATE CP
PROCESSOR 08 ALTERNATE CP
PROCESSOR 09 ALTERNATE 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 6 Release 1.0, service level 0901 (64-bit)
Generated at 09/11/09 16:51:48 EDT
IPL at 08/31/10 08:44:19 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
------ ---- ---------- ---------- ------ ------ ------ ---- -------------
JV6280 6280          1         20     20      1      1   5% DRCT ACTIVE
JS6281 6281          1       3338 600840 110530 115643  18% SPOOL
WS639C 639C          -          -      0      0      0   0  SHARED
JP628A 628A          0       3338 601020      0      0   0% PAGE
JP6288 6288          0       3338 601020      0      0   0% PAGE
JP6233 6233          0       3338 601020      0      0   0% PAGE
JP6232 6232          0       3338 601020     12     12   1% PAGE
JP6282 6282          1       3338 600840      0      0   0% 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 will report on the OSA resources. Finally, the **QUERY VIRTUAL ALL** command can be useful. The following list gives the short form of these commands without any of the associated output shown:

```
==> q da
==> q da free
==> q osa
==> q osa free
==> q v all
```

19.2 The z/VM Performance Toolkit

To use the z/VM Performance Toolkit, the product must be ordered. You should only configure the product if you have ordered it.

Much more detail can be found in the following books:

- *z/VM Performance Toolkit Guide*, SC24-6156, *z/VM Performance Toolkit Reference*, SC24-6157, on the Web starting at the z/VM 5.4 bookshelf:

  http://www-03.ibm.com/systems/z/os/zos/bkserv/zvmpdf/#zvm61

  Search for **Toolkit** on that page.

- *The Program Directory for Performance Toolkit for VM*, GI10-0785-00

  http://www.vm.ibm.com/progdir/6vmptk10.pdf

- The IBM Redbook *Linux on IBM zSeries and S/390: Performance Toolkit for VM*, SG24-6059, on the Web at:

  http://www.redbooks.ibm.com/abstracts/sg246059.html

The section that follow describe how to set up and use the IBM Performance Toolkit *very briefly*:

- “Configure the z/VM Performance Toolkit” on page 314
- “Use the z/VM Performance Toolkit” on page 318

19.2.1 Configure 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. Again, you should configure the product only if you have ordered it.

- Query which priced products are enabled with the **QUERY PRODUCT** command:

```
==> q product
Product  State    Description
6VMDIR10 Disabled 00/00/00.00:00:00.$BASEDDR DIRECTORY MAINTENANCE FL 610
6VMPTK10 Disabled 00/00/00.00:00:00.$BASEDDR PERFORMANCE TOOLKIT FOR VM
6VMRAC10 Disabled 00/00/00.00:00:00.$BASEDDR RACF for VM
6VMRSC10 Disabled 00/00/00.00:00:00.$BASEDDR RSCS Networking Version 6 Release 1
 Modification 0
```

- To enable The z/VM Performance Toolkit, log on **to MAINT620** 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 screens of messages scroll by and finally the success messages shown above. This will enable the Performance Toolkit for the current z/VM session.

- At IPL time the `SYSTEM CONFIG` file is modified by having a line appended to the end. Verify this has been added by the **SERVICE** command with the following commands:

```
==> vmlink pmaint cf0
DMSVML2060I PMAINT CF0 linked as 0120 file mode Z
==> type system config z
...    // many screens cleared
PRODUCT PRODID 6VMPTK20 STATE ENABLED DESCRIPTION '11/24/11.23:55:56.MAINT620 PE
RFKIT Minidisk Install and Service'
```

The Performance Toolkit is now enabled. You can also verify by running the **QUERY PRODUCT** command again.

19.2.2 Configure Web Browser support

Once the product is enabled, the TCPIP profile must be modified to enable Web access to the Performance Toolkit. The following example sets the port to 80, the default for a Web browser:

- Log On to TCPMAINT. Edit the TCPIP configuration file. In this example it is POKSND61 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 poksnd61 tcpip d
====> /port
```

- Add the following line 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 DHCPD               ; DHCP Server
; 69   UDP TFTPD               ; TFTPD (Trivial FTP) Server
; 69   UDP TFTPD               ; TFTPD (Trivial FTP) Server
  80   TCP PERFSVM             ; Performance Toolkit
; 111  TCP PORTMAP             ; Portmap Server
...
```

Save your changes.

- The TCPIP virtual machine needs to be recycled in order for our changes to take effect. You can **FORCE** and **XAUTOLOG TCPIP** from the Integrated 3270 console. Alternatively, if you are in a position to re-IPL the system, you can do that. Finally, the **OBEYFILE** command can be used, but it is not documented in this book (it is left as an exercise for the reader :)).
- When the system comes back, log on to TCPMAINT and check if everything was successful by issuing the **NETSTAT CLIENTS** command. You want to see that the service PERFSVM is a client (listening). This should be shown after a few screens of output:

```
==> netstat clients
...
Client: PERFSVM                  Authorization: {none}
Notes Handled: none
Last Touched:   0:01:22
Vmcf error count: 0
```

The entry for PERFSVM should be at the end of the output.

19.2.3 Configure PERFSVM

The PERFSVM virtual machine is the Performance Toolkit service machine.

- Log On to PERFSVM. If you successfully enabled the product, you should be put in a Performance Toolkit session and see the following text at the top of the screen:

```
FCX001                 Performance Toolkit for VM                Autoscroll 12
  FCXBAS500I Performance Toolkit for VM FL610
  Monitor event started -- recording is activated
  Monitor sample started -- recording is activated
  FCXPMN446E Incomplete monitor data: SAMPLE CONFIG size too small
```

- Press **F12 twice** to get to a CMS prompt.
- Copy the default configuration files, which are on PERFSVM's D disk, to your A disk:

```
==> copy * * d = = a
```

- The main configuration file is FCONX $PROFILE. Edit that file and search for the string VMCF:

```
==> x fconx $profile
====> /vmcf
```

This should take you to line 189 where the next 6 lines are comments starting with an `*`. Perform the following changes:

- Uncomment the 2nd, 4th, 6th and 8th lines by changing `*C` to **FC**
- Change port `81` to **80** on the fourth line - this will enable you to use a browser interface without having to specify port 81 on the URL (with a `:81` suffix).

The modified lines should be as follows. Save your changes with the **FILE** subcommand:

```
*    Following command activates VMCF data retrieval interface
FC MONCOLL VMCF ON
*    Define the maximum allowed number of Internet connections
FC MONCOLL WEBSERV MAXCONN 100
*    Define the timeout of inactive Internet connections in minutes
FC MONCOLL WEBSERV TIMEOUT 30
*    Following command activates Internet interface
FC MONCOLL WEBSERV ON TCPIP TCPIP 80
*    Following command activates Internet interface with SSL
...
====> file
```

- Create a remote data retrieval authorization file with your z/VM system identifier (replace ***POKSND61*** with your system identifier):

```
==> x fconrmt authoriz
====> a 2
POKSND61 PERFSVM  S&FSERV
POKSND61 MAINT    DATA CMD EXCPMSG
```

- Create a system identification file that links your z/VM system and `PERFSVM` to a special resource name called `FCXRES00`. (replace ***POKSND61*** with your system identifier):

```
==> x fconrmt systems
====> a
POKSND61 PERFSVM  ESA   N   FCXRES00
```

- Edit the `PROFILE EXEC` file, search for the word "once" and uncomment the five `MONITOR SAMPLE` and the two `MONITOR EVENT` statements:

```
==> x profile exec a
====> /once
```

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 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'

'PERFKIT'                  /* Invoke the PERFKIT module   @FC012BD*/

Exit
====> file
```

- Set the PERFSVM virtual machine to be started at z/VM IPL time. Edit the PROFILE EXEC on AUTOLOG1 so that PERFSVM is automatically started at IPL time.
- Log on to AUTOLOG1.
- Before pressing Enter at the VM READ prompt, type **acc (noprof** so that the **PROFILE EXEC** is not run.

```
LOGON AUTOLOG1
z/VM Version 6 Release 1.0, Service Level 1002 (64-bit),
built on IBM Virtualization Technology
There is no logmsg data
FILES:   NO RDR,   NO PRT,   NO PUN
LOGON AT 14:51:02 EDT THURSDAY 10/07/10
DMSIND2015W Unable to access the Y-disk. Filemode Y (19E) not accessed
z/VM V6.1.0    2010-09-23 11:31
==> acc (noprof
```

- Add a line so the virtual machine PERFSVM is started at z/VM IPL time:

```
==> x profile exec
/*********************************************************************/
/* Customer processing can be added here                             */
/*********************************************************************/
"CP XAUTOLOG TCPIP"              /* Start TCPIP                      */
"CP SET MDC STOR OM 256M"        /* Limit minidisk cache in CSTOR    */
"CP SET MDC XSTORE OM OM"        /* Disable minidisk cache in XSTOR  */
"CP SET SIGNAL SHUTDOWN 600"     /* Allow guests 10 min to shut down */
"CP XAUTOLOG VSMGUARD"           /* Start SMAPI                      */
"CP XAUTOLOG PERFSVM"            /* Start the Performance Toolkit    */
```

- Save the file and logoff of AUTOLOG1.

19.2.4 Start the z/VM Performance Toolkit

To start the Performance Toolkit, perform the following steps:

- Logon to the PERFSVM virtual machine.
- Press **Enter** and the performance toolkit should start through the PROFILE EXEC:

```
FCX001                 Performance Toolkit for VM                Autoscroll 12
 FCXBAS500I Performance Toolkit for VM FL610
 FCXAPP530I Connected to *IDENT for resource FCXRES00
 FCXAPF530I Connected to *IDENT for resource FCXSYSTM
 FCXTCP571I Connected to TCP/IP server TCPIP on path 0003
 FCXAPP527I User PERFSVM connected on path 0006
 FCXAPC535I Connected to resource FCXRES00 on path 0005, for S&F-Coll
 FCXTCP575I WebServer host IP address is 9.60.18.249:00080
 FCXTCP590I WebServer interface activated
```

```
Monitor event started -- recording is activated
Monitor sample started -- recording is activated
```

Disconnect from `PERFSVM` now.

```
Command ===> disc
```

The Performance Toolkit should now be configured and running.

19.2.5 Use the z/VM Performance Toolkit

The Performance Toolkit can be used with a Web browser or 3270 interface.

Using a Web browser interface

To use the Web-enabled Performance Toolkit, perform the following steps:

- Point a browser to your z/VM system. For example:

  `http://9.60.18.249`

- You should see a splash screen, then the Web Server Logon screen as shown in Figure 19-1 on page 318:

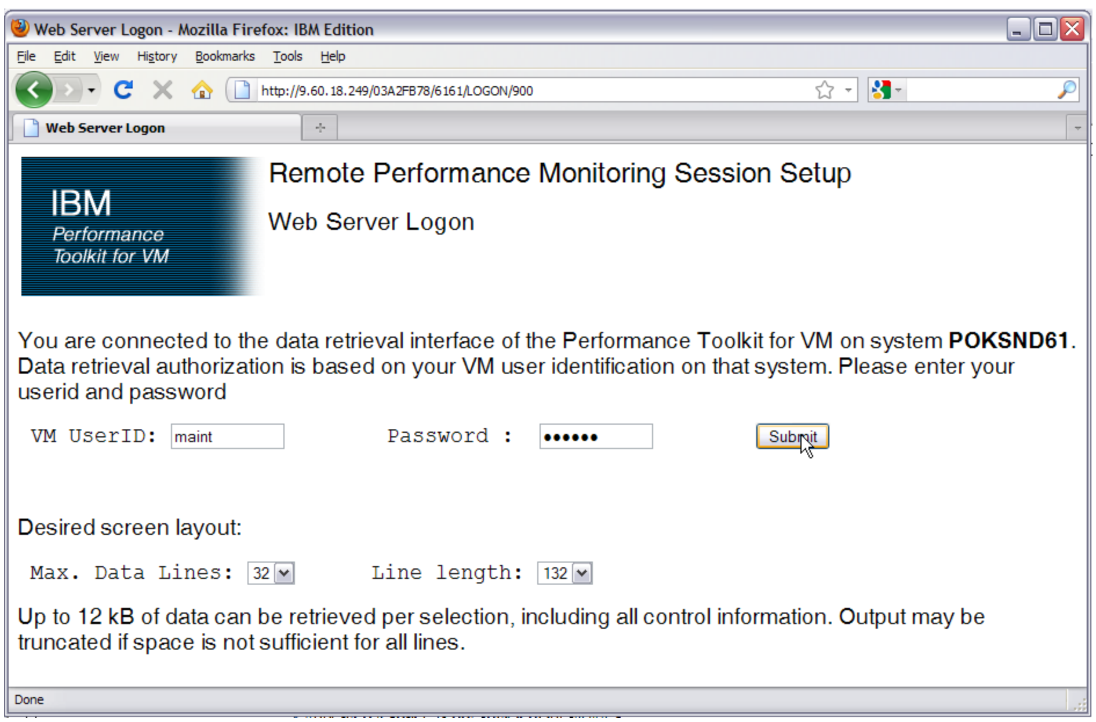


Figure 19-1 Performance Toolkit logon screen

- Enter any valid user ID and password and click **Submit**. In this example `MAINT` is used.
- The *Central Monitoring System Load Overview* appears with your system identifier (*Node-ID*) on the left side.
- Click on your system identifier and the *Initial Performance Data Selection Menu* screen appears as shown in Figure 19-2 on page 319.
- From this screen, you can drill down into many different types of reports.
- Screen 21, *User resource usage*, provides useful monitoring information.

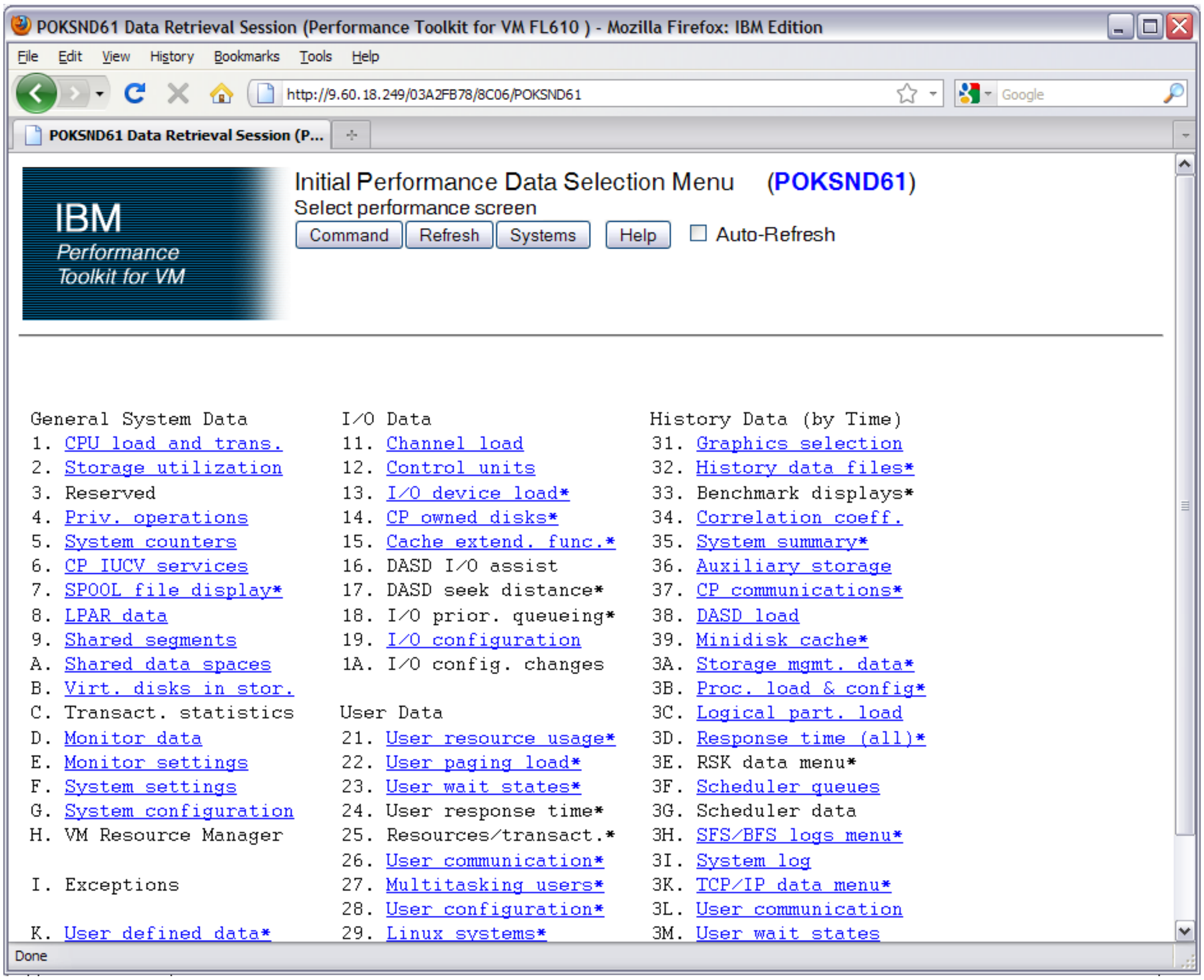


Figure 19-2 Browser interface to the Performance Toolkit

Use a 3270 interface

You can also use a 3270 interface as well as a browser interface. To do so, perform the following steps:

- Logon to `PERFSVM`.
- If you had disconnected, pressing Enter should get you back to the Performance Toolkit command line. If the virtual machine was logged off, the **`PROFILE EXEC`** should run and get you to the command line. Enter the command **`MONITOR`**:

```
Command ==> monitor
```

```
FCX124         Performance Screen Selection  (FL610       )    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. *
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 19-3 Performance Toolkit 3270 Interface Main Menu screen

Drilling down into report screens

You should now be able to use the active report screens. To drill down into these screens, 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 screens to drill down into are:

```
21. User resource usage
22. User paging load
23. User wait states
28. User configuration
29. Linux systems
33. Benchmark displays
```

For example to drill down into the Benchmark submenu screen, enter the following command:

```
Command ===> 33
```

Then type **S** over the period on the left side of the submenu screen in the row corresponding to the report you wish to see.

19.3 Monitor Linux performance data from the kernel

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 wish to monitor. There are different ways of gathering this data. It is recommended that data be gathered in the kernel. All modern Linux distributions have been enabled for the kernel to gather performance data.

To monitor Linux performance data directly from the kernel, the following 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 as the `OPTION APPLMON` was set for the Linux administration system, the golden image and for Linux virtual machines in earlier sections.

For the second requirement, details of this function are described in the Chapter, *Linux monitor stream support for z/VM* in the manual Linux on *System z Device Drivers, Features and Commands on Red Hat Enterprise Linux 6*, on the Web at:

http://www.ibm.com/developerworks/linux/linux390/documentation_red_hat.html

A quick description of how to use this built-in monitoring function follows.

- Start an SSH session to a Linux system. In this example, `LINUX153` is used.
- There are three modules that are built into the kernel but are not loaded by default. They are named `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
```

- There is no output so no modules with the string `appldata` are loaded. Load those modules with the **modprobe** command and verify they have been loaded:

```
# modprobe appldata_mem
# modprobe appldata_os
# modprobe appldata_net_sum
```

- Now if you repeat the **lsmod** command, you should see the following:

```
# lsmod | grep appldata
appldata_net_sum        1844  0
appldata_os             2987  0
appldata_mem            1966  0
```

- The directory in the virtual `/proc/` file system where the monitoring variables exist is `/proc/sys/appldata/`. In this directory there are five files as follow:

`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

- 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
```

Built-in kernel monitoring should now be turned on.You may only want to leave the monitoring on for specific periods of time. As Linux monitoring data is captured, the Performance Toolkit's minidisk space can fill up relatively quickly.

19.4 View 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. 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 the following figure.

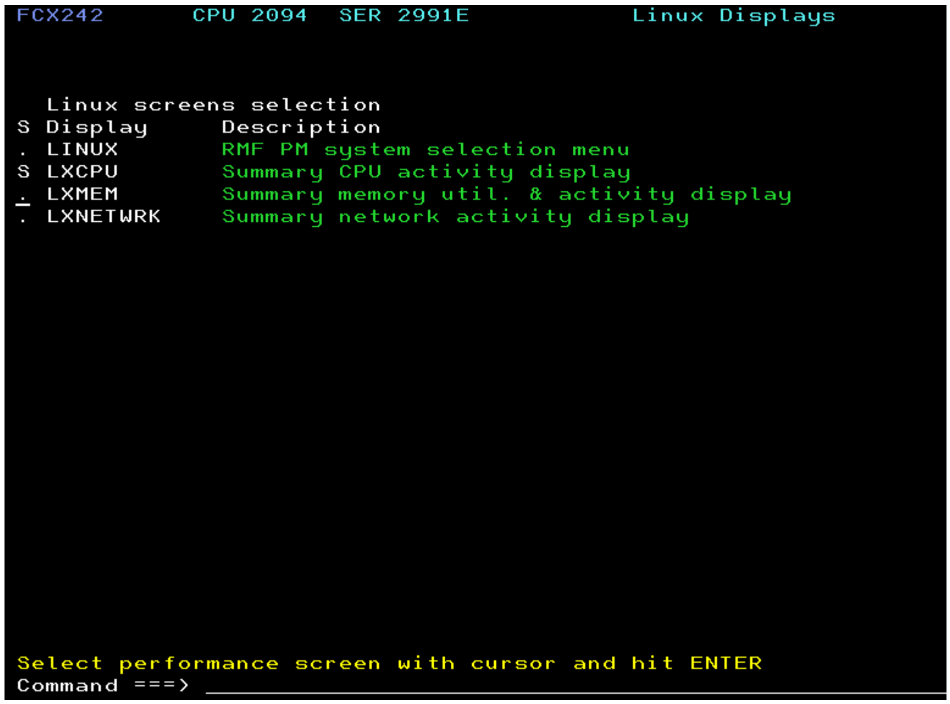


Figure 19-4 Linux Guest Systems sub menu

Then type **S** over the period on the left side of the submenu screen in the row corresponding to the report you wish to see. You should see a new report screen with the Linux guest systems CPU overview.

You can also use a Web interface to view the same data. You would drill down into menu 29 Linux systems and should see be able to drill down into **`LXCPU`** (Linux CPU), **`LXMEM`** (Linux memory) and **`LXNET`** (Linux Network) links.

19.5 Monitor Linux with sysstat

> **Credit:** the content for this section was shamelessly copied and adapted from the *Linux and Mainframe* blog, on the Web at:
>
> http://linuxmain.blogspot.com/

The *sysstat* package is included in both RHEL 6.2 and SLES 11 SP2, but not always installed by default. It's a collection of performance monitoring tools and you can check the options for your current version in the manual pages. It's platform independent and so this works on Linux in general.

After installation of the package there are two different ways to gather data - either from the command line, or by configuring **sysstat** to run as a service.

Using the command line

Ad hoc data can be gathered by calling the data collector with the **sysstat** command:

```
# /usr/lib64/sa/sadc -S DISK -F 5 outfile.sa
```

Attention: For older versions of **sysstat**, the command is:

```
# /usr/lib64/sa/sadc -d -F 5 outfile.sa
```

Data will be collected in 5 second intervals until it's interrupted by ^C. The `-F` flag forces a file compatible with the current **sysstat** version and the `-S DISK` or `-d` enables the collection of statistics for the block devices.

Convert the binary file created by the previous command to a text report file with the following **sar** command:

```
# sar -A -f outfile.sa > outfile.txt
```

See the **sar** man page for the many other options that exist. For example, the **-A** option shows all the data.

Sar data is stored in the directory `/var/log/sa`. Data files are of the form `sa<dd>` where dd is the day of the month. Text files are of the form `sar<dd>`. For example:

```
# cd /var/log/sa
# file *
sa09:  data
sa10:  data
sa11:  data
sa12:  data
sa13:  data
sar09: ASCII text
sar10: ASCII text
sar11: ASCII text
sar12: ASCII text
```

Configure sysstat on RHEL 6.2

On RHEL 6.2, **sysstat** should be installed and running as a service:

```
# rpm -qa | grep sysstat
sysstat-9.0.4-18.el6.s390x
# chkconfig --list sysstat
sysstat         0:off   1:on    2:on    3:on    4:on    5:on    6:off
```

The configuration file is `/etc/sysconfig/sysstat`:

```
# cat /etc/sysconfig/sysstat
# sysstat-9.0.4 configuration file.

# How long to keep log files (in days).
# If value is greater than 28, then log files are kept in
# multiple directories, one for each month.
HISTORY=7

# Compress (using gzip or bzip2) sa and sar files older than (in days):
COMPRESSAFTER=10
```

For more information, see the man pages for **sar** and **sadc**.

Configure sysstat on SLES 11 SP2

To install and configure sysstat on SLES 11 SP2, perform the following steps:

- Install sysstat with the **zypper** command and

```
# zypper install sysstat
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW package is going to be installed:
  sysstat

1 new package to install.
Overall download size: 172.0 KiB. After the operation, additional 661.0 KiB
will be used.
Continue? [y/n/?] (y): y
Retrieving package sysstat-8.1.5-7.27.1.s390x (1/1), 172.0 KiB (661.0 KiB unpacked)
Installing: sysstat-8.1.5-7.27.1 [done]
```

- Verify it was installed with **rpm -qa**:

```
# rpm -qa | grep sysstat
sysstat-8.1.5-7.27.1
```

- Turn the sysstat service on at boot time with the **chkconfig** command and for this session with the **service** command:

```
# chkconfig boot.sysstat on
# service boot.sysstat start
Running sadc                                                                    done
```

- In the configuration file `/etc/sysstat/sysstat` the archive settings for this history can be configured:

```
# cat /etc/sysstat/sysstat
# sysstat-8.1.5 configuration file.

# How long to keep log files (in days).
# If value is greater than 28, then log files are kept in
# multiple directories, one for each month.
HISTORY=7

# Compress (using gzip or bzip2) sa and sar files older than (in days):
COMPRESSAFTER=10
```

At the end there is a file/link to `/etc/cron.d/` that you can adapt and e.g. changing the 10 min collection interval or adapting the reports. As usual the documentation for the sa1 and sa2 commands used there are in the respective man page.

20

Miscellaneous recipes

“Two things are infinite: the universe and human stupidity; and I'm not sure about the universe.”

--Albert Einstein

This chapter has the following sections of miscellaneous tasks that you might want to perform:

20.1 Add disk space to virtual machines

The following process describes how to add additional DASD to a Linux virtual machine. The overall steps are:

20.1.1 Add minidisks to a virtual machine

Following are the high level steps to add a new 3390-3 and a new 3390-9 to `LNXADMIN` on the first SSI member. The 3390-3 is split in half to show an example of creating a new logical volume. The 3390-9 is used to extend the existing logical volume mounted over `/var/`.

To add new minidisks to a virtual machine, perform the following steps:

- Determine the volumes that will be added. In this example, a 3390-3 at real device address 633A (which has been formatted and attached to SYSTEM) is split into two smaller minidisks, and a 3390-9 at address 63AA (which has not been formatted and attached to SYSTEM) is added as a larger minidisk.
- Format the volumes and attach to SYSTEM. This can be done manually, one at a time, with the CP command **CPFMTXA**, or to multiple volumes with the **CPFORMAT** EXEC, supplied in the tar file associated with this book.
- Add minidisk statements to define minidisks. In this example three minidisks at virtual addresses 104-106 are defined to the LNXADM-1 SUBCONFIG. Following is the updated directory entry:

```
SUBCONFIG LNXADM-1
 MDISK 0100 3390 0001  3338 JM6290 MR LNX4VM LNX4VM LNX4VM
 MDISK 0101 3390 0521  2818 JM6289 MR LNX4VM LNX4VM LNX4VM
 MDISK 0102 3390 0001 10016 JM61A5 MR LNX4VM LNX4VM LNX4VM
 MDISK 0103 3390 0001 10016 JM61B2 MR LNX4VM LNX4VM LNX4VM
 MDISK 0104 3390 0001  1669 JM633A MR LNX4VM LNX4VM LNX4VM
 MDISK 0105 3390 1670  1669 JM633A MR LNX4VM LNX4VM LNX4VM
 MDISK 0106 3390 0001 10016 JM63AA MR LNX4VM LNX4VM LNX4VM
```

- Bring the changes to the user directory online either with DirMaint or DIRECTXA.

  – **Attention**: If you add disk(s) to the user directory, it is possible to attach them to a running Linux system without "bouncing" it. For example, if you added a new minidisk at virtual address 104, use the command: **vmcp link * 104 104 mr** to link to the minidisk, then **chccwdev -e 104** to enable it in Linux. Then see 20.3, "Extend an existing logical volume" on page 332

- Shutdown the Linux system.
- Logoff the virtual machine.
- Log back on to it and IPL Linux.

20.1.2 Make new minidisks available

To make the new minidisks available, perform the following steps:

- When your system comes back up, **start an SSH session** to it. Use the **lsdasd** command to verify that the new minidisks are not seen yet:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0301   active      dasdb     94:4    FBA   512    512MB     1048576
0.0.0300   active      dasdc     94:8    FBA   512    256MB     524288
0.0.0103   active      dasdd     94:12   ECKD  4096   7042MB    1802880
0.0.0102   active      dasde     94:16   ECKD  4096   7042MB    1802880
0.0.0101   active      dasdf     94:20   ECKD  4096   1981MB    507240
```

- Enable the disks with the **chccwdev -e** command:

```
# chccwdev -e 104 105 106
Setting device 0.0.0104 online
Done
Setting device 0.0.0105 online
Done
Setting device 0.0.0106 online
Done
```

- View the available disks again with the **lsdasd** command:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0301   active      dasdb     94:4    FBA   512    512MB     1048576
0.0.0300   active      dasdc     94:8    FBA   512    256MB     524288
0.0.0103   active      dasdd     94:12   ECKD  4096   7042MB    1802880
0.0.0102   active      dasde     94:16   ECKD  4096   7042MB    1802880
0.0.0101   active      dasdf     94:20   ECKD  4096   1981MB    507240
0.0.0104   active      dasdg     94:24   ECKD  4096   1173MB    300420
0.0.0105   active      dasdh     94:28   ECKD  4096   1173MB    300420
0.0.0106   active      dasdi     94:32   ECKD  4096   7042MB    1802880
```

- Format the disks in parallel with the **dasdfmt** command using a **for** loop and putting them in the background:

```
# for i in g h i
> do
>   dasdfmt -b 4096 -y -f /dev/dasd$i &
> done
[1] 1923
[2] 1924
[3] 1925
[root@gpok151 ~]# Finished formatting the device.
Rereading the partition table... ok
Finished formatting the device.
Rereading the partition table... ok
Finished formatting the device.
Rereading the partition table... ok

[1]   Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
[2]-  Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
[3]+  Done                    dasdfmt -b 4096 -y -f /dev/dasd$i
```

- Create one partition from each of the minidisks using a bash **for** loop and the **fdasd -a** command:

```
# for i in g h i
> do
>   fdasd -a /dev/dasd$i
> done
reading volume label ..: VOL1
reading vtoc ..........: ok
auto-creating one partition for the whole disk...
...
```

- **If you are adding disks to RHEL 6.2,** make a backup of `/etc/dasd.conf`, then add minidisks 104, 105 and 106 to it:

```
# cd /etc
# cp dasd.conf dasd.conf.orig
# vi dasd.conf
0.0.0103 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0102 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0301 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0300 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0101 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0100 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0103 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0102 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0301 use_diag=0 readonly=0 erplog=0 failfast=0
```

```
0.0.0300 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0101 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0100 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0104 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0105 use_diag=0 readonly=0 erplog=0 failfast=0
0.0.0106 use_diag=0 readonly=0 erplog=0 failfast=0
```

- If you are adding disks to SLES 11 SP2, use the **dasd_configure** command to enable minidisks 104, 105 and 106:

```
# dasd_configure 0.0.0104 1
Configuring device 0.0.0104
Setting device online
# dasd_configure 0.0.0105 1
Configuring device 0.0.0105
Setting device online
# dasd_configure 0.0.0106 1
Configuring device 0.0.0106
Setting device online
```

The three new minidisks should now be formatted, partitioned and configured to be active at boot time.

If you are creating a new logical volume, see 20.2.1, "Create a logical volume and file system" on page 328. If you are extending an existing logical volume, skip ahead to 20.3, "Extend an existing logical volume" on page 332

20.2 Add 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 for many users to write data to. When this is the case, you can use the Logical Volume Manager (LVM) to combine multiple DASD volumes into one logical volume. This example does not create a large logical volume, but it shows all the steps necessary to do so.

The following process describes a logical volume with additional DASD on a Linux guest. The overall steps in adding a logical volume are:

- "Add disk space to virtual machines" on page 325
- "Create a logical volume and file system" on page 328
- "Update the file system table" on page 331

20.2.1 Create a logical volume and file system

The overall steps involved in creating a logical volume are:

- Create physical volumes from the two partitions
- Create a single volume group
- Create a single logical volume
- Make a file system from the logical volume

Figure 20-1 on page 329 shows a block diagram of the logical volume manager reflecting this example.

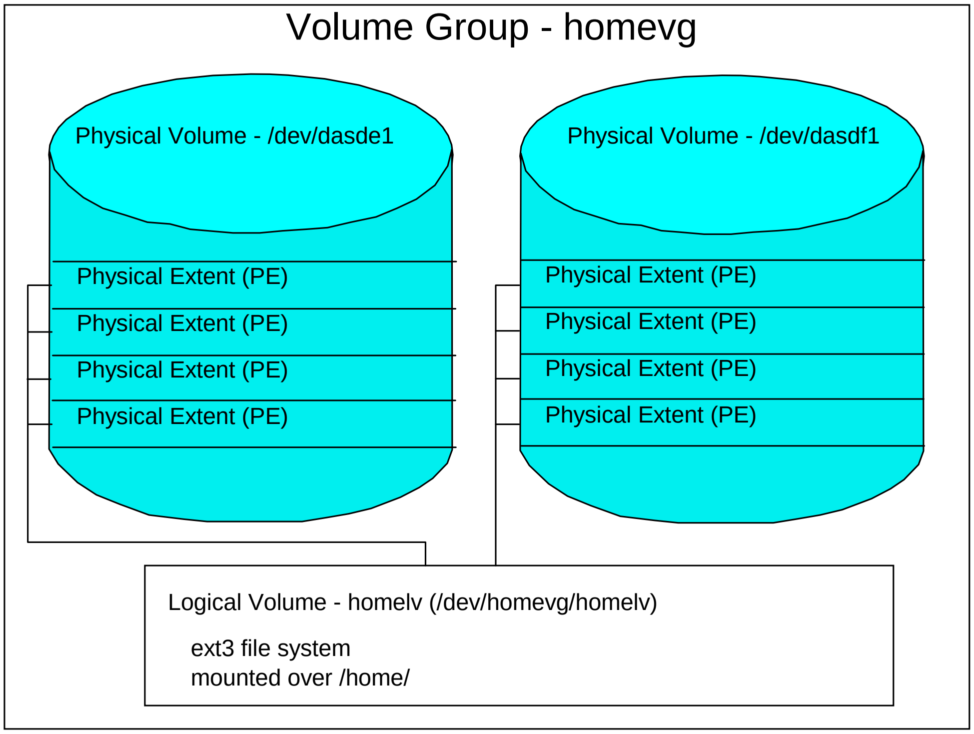


Figure 20-1 LVM block diagram

Create physical volumes from the two DASD

To create physical volumes from the new minidisks at virtual device addresses `104` and `105`, perform the following steps:

- The **`pvcreate`** command initializes partitions for use by LVM. Initialize the two new DASD partitions:

```
# pvcreate /dev/dasdg1 /dev/dasdh1
  Writing physical volume data to disk "/dev/dasdg1"
  Physical volume "/dev/dasdg1" successfully created
  Writing physical volume data to disk "/dev/dasdh1"
```

- Verify that the physical volumes were created with the **`pvdisplay`** command:

```
# pvdisplay /dev/dasdg1 /dev/dasdh1
  "/dev/dasdg1" is a new physical volume of "1.15 GiB"
  --- NEW Physical volume ---
  PV Name               /dev/dasdg1
  VG Name
  PV Size               1.15 GiB
  Allocatable           NO
  PE Size               0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               RGzGgx-ZR56-tNEx-sOOU-ih03-GK38-DgSRu5

  "/dev/dasdh1" is a new physical volume of "1.15 GiB"
  --- NEW Physical volume ---
  PV Name               /dev/dasdh1
  VG Name
  PV Size               1.15 GiB
  Allocatable           NO
```

```
  PE Size               0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               qpZfzj-6NzG-9a8k-lFAB-APFd-a81t-raVj7I
```

Create a single volume group

The **vgcreate** command is used to create a volume group named `homevg` from the two partitions. Use the **vgdisplay homevg** command to verify the volume group was created:

```
# vgcreate homevg /dev/dasdg1 /dev/dasdh1
Volume group "homevg" successfully created
# vgdisplay homevg
  --- 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               2.29 GiB
  PE Size               4.00 MiB
  Total PE              586
  Alloc PE / Size       0 / 0
  Free  PE / Size       586 / 2.29 GiB
  VG UUID               iGkyf2-eIVf-BmGE-unNh-hSaz-eSB2-yaTKxm
```

In this example, there are 586 free physical extents.

Create a single logical volume

The **lvcreate** command is used to create a logical volume. The **-l** flag specifies to use all free extents, 586 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:

```
# lvcreate -l 586 -n homelv homevg
  Logical volume "homelv" created
```

Use the **lvdisplay** command to verify. The parameter is the full path of the logical volume, not just the logical volume name:

```
# lvdisplay /dev/homevg/homelv
  --- Logical volume ---
  LV Name                /dev/homevg/homelv
  VG Name                homevg
  LV UUID                Jxstaw-WkCx-wDcf-a6Dd-26ac-2uZ2-ivBZwz
  LV Write Access        read/write
  LV Status              available
  # open                 0
  LV Size                2.29 GiB
  Current LE             586
  Segments               2
  Allocation             inherit
```

```
Read ahead sectors     auto
- currently set to     1024
Block device           253:1
```

Make a file system from the logical volume

Create a file system from the new logical volume.

If you are on RHEL 6.2, ext4 is the recommended file system. Create an ext4 file system out of it using the **mkfs.ext4** command:

```
# mkfs.ext4 /dev/homevg/homelv
mke2fs 1.41.12 (17-May-2010)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=1 blocks, Stripe width=0 blocks
150176 inodes, 600064 blocks
30003 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=616562688
19 block groups
32768 blocks per group, 32768 fragments per group
7904 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 33 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

If you are on SLES 11 SP2, ext3 is the recommended file system.

```
# mkfs -j /dev/homevg/homelv
...
```

The file system created from the logical volume is now ready to be mounted.

20.2.2 Update 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. Make a backup copy then add the following line to the file:

```
# cd /etc
# cp fstab fstab.works
```

- add one line to the fstab file:

```
# vi fstab
... // For RHEL 6.2:
/dev/homevg/homelv      /home                   ext4    defaults        0 0
... // For SLES 11 SP2:
/dev/homevg/homelv      /home                   ext3    defaults        0 0
...
```

- Before mounting over `/home/`, you may want to check that it is empty. If a non-root user exists and a new file system is mounted over it, the contents of the directory will be *covered*. In this example there is no data in the file system.

```
# ls -a /home
.  ..
```

- Mount the `/home/` file system with one argument. By using just one argument, you are testing the change to `/etc/fstab`. Use the **`df -h`** command to verify that it is mounted:

```
# mount /home
# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/dasda1           1.8G  1.6G  123M  93% /
tmpfs                 121M     0  121M   0% /dev/shm
/dev/mapper/nfs_vg-nfs_lv
                       16G   13G  2.6G  83% /var
/dev/mapper/homevg-homelv
                      2.3G   68M  2.1G   4% /home
```

- Test a reboot to verify the new logical volume is successfully mounted over `/home/`.

```
# reboot

Broadcast message from root (pts/0) (Thu Sep  2 15:08:07 2010):

The system is going down for reboot NOW!
```

When the system comes back, you should see the new logical volume mounted over `/home/`.

20.3 Extend 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 `/var/` file system is filling up on SSI member 1:

```
# df -h /var/
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/nfs_vg-nfs_lv
                       16G   13G  2.6G  83% /var
```

A 3390-9 was added as minidisk `106` in section 20.1, “Add disk space to virtual machines” on page 325.

> **Attention:** It is possible to attach minidisk(s) to a running Linux system without “bouncing” it. For example, if you added a new minidisk at virtual address `106`, from a root SSH session, use the command: **`vmcp link * 106 106 mr`** to link to the minidisk, then **`chccwdev -e 106`** to enable it.

To extend the logical volume using this disk, perform the following steps:

- Use the **`vgdisplay`** command to see the free space in the volume group `nfs_vg`:

```
# vgdisplay nfs_vg
  --- Volume group ---
  VG Name               nfs_vg
  System ID
  Format                lvm2
  Metadata Areas        3
```

```
  Metadata Sequence No  2
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               1
  Max PV                0
  Cur PV                3
  Act PV                3
  VG Size               15.68 GiB
  PE Size               4.00 MiB
  Total PE              4015
  Alloc PE / Size       4015 / 15.68 GiB
  Free  PE / Size       0 / 0
  VG UUID               34Wjed-sGHk-eBxt-R81d-hPUi-SxhB-81szCp
```

This shows that there are no free extents in the volume group

- Use the **lsdasd** command to show the enabled disks:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================================
0.0.0100   active      dasda     94:0    ECKD  4096   2347MB    600840
0.0.0301   active      dasdb     94:4    FBA   512    512MB     1048576
0.0.0300   active      dasdc     94:8    FBA   512    256MB     524288
0.0.0103   active      dasdd     94:12   ECKD  4096   7042MB    1802880
0.0.0102   active      dasde     94:16   ECKD  4096   7042MB    1802880
0.0.0101   active      dasdf     94:20   ECKD  4096   1981MB    507240
0.0.0104   active      dasdg     94:24   ECKD  4096   1173MB    300420
0.0.0105   active      dasdh     94:28   ECKD  4096   1173MB    300420
0.0.0106   active      dasdi     94:32   ECKD  4096   7042MB    1802880
```

This shows that minidisk 106 is at /dev/dasdi.

- Make minidisk 106 a physical volume with the **pvcreate** command:

```
# pvcreate /dev/dasdi
  Writing physical volume data to disk "/dev/dasdi"
  Physical volume "/dev/dasdi" successfully created
```

- Use the **vgextend** command to add the minidisk to the volume group:

```
# vgextend nfs_vg /dev/dasdi1
  No physical volume label read from /dev/dasdi1
  Writing physical volume data to disk "/dev/dasdi1"
  Physical volume "/dev/dasdi1" successfully created
  Volume group "nfs_vg" successfully extended
```

- Use the **vgdisplay** command again to show the free extents in the volume group:

```
# vgdisplay nfs_vg
  --- Volume group ---
  VG Name               nfs_vg
  System ID
  Format                lvm2
  Metadata Areas        4
  Metadata Sequence No  3
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               1
  Max PV                0
  Cur PV                4
```

```
  Act PV                4
  VG Size               22.56 GiB
  PE Size               4.00 MiB
  Total PE              5775
  Alloc PE / Size       4015 / 15.68 GiB
  Free  PE / Size       1760 / 6.88 GiB
  VG UUID               34Wjed-sGHk-eBxt-R81d-hPUi-SxhB-81szCp
```

This shows that there are now 1760 free extents in the volume group.

- Use the mount command to determine the name of the logical volume mounted over `/var/`:

```
# mount | grep /var
/dev/mapper/nfs_vg-nfs_lv on /var type ext4 (rw)
```

In this example it is `/dev/mapper/nfs_vg-nfs_lv/`.

- Use the **`lvextend`** command to extend the volume group with all of the new extents:

```
# lvextend -l +1760 /dev/mapper/nfs_vg-nfs_lv
  Extending logical volume nfs_lv to 22.56 GiB
  Logical volume nfs_lv successfully resized
```

- Use the **`resize2fs`** command to increase the size of the file system while it is still mounted.

```
# resize2fs /dev/mapper/nfs_vg-nfs_lv
resize2fs 1.41.12 (17-May-2010)
Filesystem at /dev/mapper/nfs_vg-nfs_lv is mounted on /var; on-line resizing required
old desc_blocks = 1, new_desc_blocks = 2
Performing an on-line resize of /dev/mapper/nfs_vg-nfs_lv to 5913600 (4k) blocks.
The filesystem on /dev/mapper/nfs_vg-nfs_lv is now 5913600 blocks long.
```

- Use the **`df`** command to show the file system size before and after you extend it as the following example shows:

```
# df -h /var
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/nfs_vg-nfs_lv
                       23G   13G  9.0G  58% /var
```

This shows that the `/var/` file system now has 9 GB of free space.

20.4 Add SCSI/FCP disks

This book has only described ECKD disks, also known as DASD. In addition, z/VM and Linux support SCSI/FCP disks.

The Fibre Channel (FC) standard was developed by the National Committee of Information Technology Standards (NCITS). The System z FCP I/O architecture conforms to these standards. System z FCP support enables z/VM and Linux running on System z to access industry-standard SCSI devices. For disk applications, these FCP storage devices utilize Fixed Block (512-byte) sectors rather than Extended Count Key Data (ECKD™) format. A new channel-path identifier (CHPID) type has been defined called `FCP`. The `FCP` CHPID type is supported on the FICON and FICON Express features of all System z processors.

This is only a brief introduction to SCSI/FCP disks and multipathing. For more complete documentation, see the Redbook *Fibre Channel Protocol for Linux and z/VM on IBM System z* on the Web at:

http://www.redbooks.ibm.com/abstracts/sg247266.html?Open

In addition, see the Redbook *Introducing N_Port Identifier Virtualization for IBM System z9*, on the Web at:

http://www.redbooks.ibm.com/abstracts/redp4125.html?Open

20.4.1 Add a single LUN

You can determine if your LPAR has these types of disks defined with the z/VM **QUERY FCP** and **QUERY FCP FREE** commands. Following is an example from a MAINT 3270 session:

```
==> q fcp
An active FCP was not found.
==> q fcp free
FCP  1F20 FREE    , FCP  1F21 FREE    , FCP  1F50 FREE    , FCP  1F51 FREE
FCP  3B00 FREE    , FCP  3B01 FREE    , FCP  3B02 FREE    , FCP  3B03 FREE
FCP  3B04 FREE    , FCP  3B05 FREE    , FCP  3B06 FREE    , FCP  3B07 FREE
FCP  3B08 FREE    , FCP  3B09 FREE    , FCP  3B0A FREE    , FCP  3B0B FREE
FCP  3B0C FREE    , FCP  3B0D FREE    , FCP  3B0E FREE    , FCP  3B0F FREE
FCP  3B10 FREE    , FCP  3B11 FREE    , FCP  3B12 FREE    , FCP  3B13 FREE
FCP  3B14 FREE    , FCP  3B15 FREE    , FCP  3B16 FREE    , FCP  3B17 FREE
...
```

The output shows that LPAR has many FCP devices free, but none of them are in use. Associated with FCP devices are World Wide Port Numbers (WWPNs) and Logical Unit Numbers (LUNs). Often, this information may be available as part of the LPAR definition. However, you may not have this information handy. If you do not have this information, it can be queried on RHEL 6. In the following section an FCP/SCSI disk is attached to LINUX153.

Perform the following steps:

- Start an SSH session as root to LINUX153.
- Verify that the zfcp module is loaded with the following command:

```
# lsmod | grep zfcp
zfcp                  144433  0 [permanent]
scsi_transport_fc      68240  1 zfcp
scsi_mod              296490  3 zfcp,scsi_transport_fc,scsi_tgt
qdio                   61977  3 zfcp,qeth_l3,qeth
```

- Change directory to /sys/bus/ccw/drivers/ and list the contents:

```
# cd /sys/bus/ccw/drivers
# ls -F
3215/  3270/  dasd-eckd/  dasd-fba/  qeth/  vmur/
```

  Note that there is no directory named zfcp/.

- Go back to the MAINT 3270 session and attach an FCP device to LINUX153 with the **ATTACH** command:

```
==> att 3b16 linux153
FCP 3B16 ATTACHED TO LINUX153 3B16
```

- Return the Linux SSH session and list the contents of the directory again. This time you should see a new directory zfcp/:

```
# ls -F
3215/  3270/  dasd-eckd/  dasd-fba/  qeth/  vmur/  zfcp/
```

- Change into that directory and list the contents:

```
# cd zfcp
# ls -F
0.0.3b16@  bind  module@  uevent  unbind
```

- Note that a symbolic link (identified by the trailing ampersand, @, after the file name in conjunction the -F flag of ls) to a new directory 0.0.3b16. Change into that directory and list the contents

```
# ls -F
availability  cutype   driver@   online  subsystem@
cmb_enable    devtype  modalias  power/  uevent
```

- Type the contents of the online file:

```
# cat online
0
```

A value of 0 shows that the device is offline.

- Echo a 1 into the file and it will be put online (you could also use the **chccwdev -e** command):

```
# echo 1 > online
# cat online
1
```

- List the contents of the directory again. You should see that many entries were added after the device was put online. The four entries in bold are the WWPNs available from this FCP device.

```
# ls -F
0x5005076306138411/  cmb_enable        host0/       peer_wwnn    subsystem@
0x500507630613c411/  cutype            in_recovery  peer_wwpn    uevent
0x500507630a10016c/  devtype           lic_version  port_remove
0x500507630a13016c/  driver@           modalias     port_rescan
availability         failed            online       power/
card_version         hardware_version  peer_d_id    status
```

- The **lsluns** command will show all of the available LUNs from a single WWPN. In the following example, the first WWPN is used

```
# lsluns -p 0x5005076306138411
Scanning for LUNs on adapter 0.0.3b16
        at port 0x5005076306138411:
                0x4010400000000000
                0x4010400100000000
                0x4010400200000000
               ...
```

- Bring a LUN online. In this example, the next free LUN is 0x4010400000000000. Change directory into the first WWPN and list the contents:

```
# cd 0x5005076306138411
# ls
access_denied  in_recovery  status  unit_add
failed         power        uevent  unit_remove
```

- The output shows that there is no active LUN under this WWPN. Bring the LUN online by echoing the value into the file unit_add and list the contents of the directory:

```
# echo  0x0x4010400000000000 > unit_add
# ls -F
0x000e4313f0f55a00/  failed       power/  uevent    unit_remove
access_denied        in_recovery  status  unit_add
```

- Note that a new directory with the LUN value is created.

```
# lszfcp -D
0.0.010a/0x500507630503c73d/0x4020400800000000 0:0:0:1074282528

# cat /proc/scsi/scsi
```

```
Attached devices:
Host: scsi0 Channel: 00 Id: 00 Lun: 1074282528
  Vendor: IBM      Model: 2107900         Rev: .310
  Type:   Direct-Access                   ANSI  SCSI revision: 05
```

- Now a /dev/sda exists, check that there are no partitions:

```
# fdisk -l /dev/sda

Disk /dev/sda: 8589 MB, 8589934592 bytes
64 heads, 32 sectors/track, 8192 cylinders
Units = cylinders of 2048 * 512 = 1048576 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
```

- Create a partition with the **fdisk** command:

```
# fdisk /dev/sda

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
         switch off the mode (command 'c') and change display units to
         sectors (command 'u').

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-8192, default 1):
Using default value 1
Last cylinder, +cylinders or +size{K,M,G} (1-8192, default 8192):
Using default value 8192

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

- Create an ext4 file system with the **mkfs.ext4** command:

```
# mkfs.ext4 /dev/sda1
mke2fs 1.41.12 (17-May-2010)
Filesystem label=
OS type: Linux
...
```

- You should now be able to mount it and see the size:

```
# mount /dev/sda1 /mnt
# df -h /mnt
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda1             7.9G  146M  7.4G   2% /mnt
```

- Create a test file:

```
# echo "this is the file foo" > /mnt/foo
# umount /mnt
```

20.4.2 Configure multipath

Multipathing allows for better availability. To set up multipathing, perform the following steps:

- Create a second WWPN

```
# cd /sys/bus/ccw/drivers/zfcp/0.0.010a
# ls
availability  cutype   driver    online  subsystem
cmb_enable    devtype  modalias  power   uevent
# echo 1 > online
```

- Note the second WWPN. In this example it is **0x500507630503c73d**:

```
# ls
0x500507630503c73d  devtype           in_recovery  peer_wwnn    status
availability        driver            lic_version  peer_wwpn    subsystem
card_version        failed            modalias     port_remove  uevent
cmb_enable          hardware_version  online       port_rescan
cutype              host2             peer_d_id    power
# cd 0x500507630503c73d
```

- Echo the same LUN into the file `unit_add` This will enable the same LUN, but from a different WWPN.

```
# cd /sys/bus/ccw/drivers/zfcp/0.0.010a
# ls
0x500507630513c73d  devtype           in_recovery  peer_wwnn    status
availability        driver            lic_version  peer_wwpn    subsystem
card_version        failed            modalias     port_remove  uevent
cmb_enable          hardware_version  online       port_rescan
cutype              host1             peer_d_id    power
# cd 0x500507630513c73d/
# ls
access_denied  in_recovery  status  unit_add
failed         power        uevent  unit_remove
# echo 0x4020400800000000 > unit_add
# cat /proc/scsi/scsi
Attached devices:
Host: scsi0 Channel: 00 Id: 00 Lun: 1074282528
  Vendor: IBM      Model: 2107900          Rev: .310
  Type:   Direct-Access                    ANSI  SCSI revision: 05
Host: scsi1 Channel: 00 Id: 00 Lun: 1074282528
  Vendor: IBM      Model: 2107900          Rev: .310
  Type:   Direct-Access                    ANSI  SCSI revision: 05
```

- At this point the system thinks there are two LUNs, but actually there are two paths to the same LUN.
- Install the `device-mapper-multipath` RPM:

```
# yum -y install device-mapper-multipath
...
```

- Create a file `/etc/multipath.conf`:

```
# cd /etc
# vi multipath.conf
defaults {
        user_friendly_names yes
}
```

- Turn the multipath service on for this session and across reboots:

```
# service multipathd start
Starting multipathd daemon: [  OK  ]
```

```
# chkconfig multipathd on

# multipath -ll
mpatha (36005076305ffc73d0000000000002008) dm-4 IBM,2107900
size=8.0G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=1 status=active
  |- 0:0:0:1074282528 sda   8:0   active ready  running
  `- 1:0:0:1074282528 sdb   8:16  active ready  running
```

- Add an entry to /etc/multipath.conf similar to the following:

```
defaults {
        user_friendly_names yes
}
# create a friendly name - test_lun
multipaths {
        multipath {
                wwid            36005076305ffc73d0000000000002008
                alias           test_lun
                no_path_retry   5
        }
}
```

- Restart the multipath service and verify that the new test_lun friendly name has been added:

```
# service multipathd restart
Stopping multipathd daemon: [  OK  ]
Starting multipathd daemon: [  OK  ]
# ls /dev/mapper
control           system_vg-tmp_lv  system_vg-var_lv  test_lunp1
system_vg-opt_lv  system_vg-usr_lv  test_lun
```

- Mount the multipathed LUN with the new name and see that the test file exists:

```
# mount /dev/mapper/test_lunp1 /mnt
# ls /mnt
foo  lost+found
```

20.4.3 Make the changes persistent

In order to make the changes persistent, two steps must be performed:

1. Put the FCP device in the virtual machines user directory entry.
2. Put the WWPN and LUN into a Linux configuration file.

Perform the following steps:

- Add a **DEDICATE** statement to virtualize 3B16 (which is the FCP device) as virtual device 200:

```
USER LINUX153 LINUX153 256M 1G G
 INCLUDE LNXDFLT
 OPTION APPLMON
 DEDICATE 0200 3B16
 MDISK 0100 3390 0001 3338 JMD974 MR LNX4VM LNX4VM LNX4VM
 MDISK 0101 3390 0001 3338 JMCD07 MR LNX4VM LNX4VM LNX4VM
```

- Bring the changes online either DirMaint, or with **DIRECTXA** if you are not using DirMaint.
- Create the file /etc/zfcp.conf. As a shortcut, you can use the output of lszfcp -D

```
# cd /etc
# lszfcp -D > zfcp.conf
```

```
# vi zfcp.conf
0.0.010a 0x500507630503c73d 0x4020400800000000
0.0.010b 0x500507630513c73d 0x4020400800000000
```

20.5 Rescue 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.

20.5.1 Enter 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:

```
zIPL v1.8.2-28.el6 interactive boot menu

 0. default (linux)

 1. linux

Note: VM users please use '#cp vi vmsg <input>'

Please choose (default will boot in 5 seconds):
```

You can use the **#cp vi vmsg** command to boot the desired menu option (zero in this example), followed by the number one for single user mode. After you log in to Linux from the 3270 emulator, you can use the **runlevel** command to verify the run level you are in:

```
==> #cp vi vmsg 0 1
...
# runlevel
1 S
```

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 **telinit 3** to continue booting normally.

20.5.2 Enter 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 `RH62GOLD` virtual machine:

- Logon to `LNXMAINT`. Copy the **RHEL62 EXEC** file to a new file named **RESCUE EXEC**, and copy the user's `PARM-RH6` file to a new file (`RH62GOLD RESCUE` in this example):

  ```
  ==> copy rhel62 exec d rescue = =
  ==> copy rh62gold parm-rh6 d = rescue =
  ```

- Edit `RESCUE EXEC` to point to the new `RESCUE` file:

```
==> x rescue exec d
...
Address 'COMMAND'
  'CP SPOOL PUN *'
  'CP CLOSE RDR'
  'CP PURGE RDR ALL'
  'PUNCH RHEL62 KERNEL * (NOHEADER'
  'PUNCH' 'USERID'() 'RESCUE   * (NOHEADER'
  'PUNCH RHEL62 INITRD * (NOHEADER'
  'CP CHANGE RDR ALL KEEP'
  'CP IPL 00C CLEAR'
  Exit
```

- Edit the RH62GOLD RESCUE file, replacing any kickstart or VNC lines with the **rescue** command line option:

```
==> x rh62gold rescue d
root=/dev/ram0 ro ip=off ramdisk_size=40000
CMSDASD=191 CMSCONFFILE=RH62GOLD.CONF-RH6
rescue
```

- Logoff of LNXMAINT.
- Logon to RH62GOLD and answer **no** to IPL from 100 question.
- Define storage to 1GB.

```
==> def stor 1g
00: STORAGE = 1G
00: Storage cleared - system reset.
Enter
```

- IPL CMS:

```
==> ipl cms
```

- Run the **RESCUE EXEC**:

```
==> rescue
00:      NO FILES PURGED
00: RDR FILE 0001 SENT FROM RH62GOLD PUN WAS 0001 RECS 113K CPY  001 A NOHOLD NO
KEEP
00: RDR FILE 0005 SENT FROM RH62GOLD PUN WAS 0005 RECS 0003 CPY  001 A NOHOLD NO
KEEP
00: RDR FILE 0009 SENT FROM RH62GOLD PUN WAS 0009 RECS 224K CPY  001 A NOHOLD NO
KEEP
00: 0000003 FILES CHANGED
00: 0000003 FILES CHANGED
...
Kernel command line: root=/dev/ram0 ro ip=off ramdisk_size=40000
                     CMSDASD=191 CMSCONFFILE=RH62GOLD.CONF-RH6
                     rescue
...
Starting sshd to allow login over the network.

Connect now to 9.60.18.144 and log in as user 'install' to start the installatio
n.
E.g. using: ssh -x install@9.60.18.144
For VNC or text mode, disable X11 forwarding (recommended) with 'ssh -x'.
For X11, enable X11 forwarding with 'ssh -X'.

You may log in as the root user to start an interactive shell.
```

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.

- Use SSH to connect to the IP address and log in as **install**.
- You should get an SSH session and see a *Choose a Language* panel. Choose your language.
- The rescue environment will prompt you for the location of the rescue image, which is located in the install tree on the Linux administration system. Choose **NFS directory**, then enter the IP address of the Linux administration system and the path **/var/nfs/rhel62**.

```
+------------------------------¦ NFS Setup +------------------------------+
¦                                                                         ¦
¦ Please enter the server and NFSv3 path to your Red Hat Enterprise Linux ¦
¦ installation image and optionally additional NFS mount options.         ¦
¦                                                                         ¦
¦      NFS server name:                   9.60.18.151_____________        ¦
¦      Red Hat Enterprise Linux directory: /var/nfs/rhel62_________       ¦
```

- The *Rescue* window appears. Choose **Continue**. The rescue image will search for your Linux installation.
- Hopefully it will prompt you to mount the partitions it finds.

```
+--------------¦ Rescue +---------------+
¦                                       ¦
¦ Your system has been mounted under    ¦
¦ /mnt/sysimage.                        ¦
¦                                       ¦
¦ Press <return> to get a shell. If you ¦
¦ would like to make your system the    ¦
¦ root environment, run the command:    ¦
¦                                       ¦
¦         chroot /mnt/sysimage          ¦
¦                                       ¦
¦ The system will reboot automatically  ¦
¦ when you exit from the shell.         ¦
```

Note: if the rescue image cannot find your partition, you can try to mount it yourself with the **mount** command. For example:

```
# mount /dev/dasda1 /mnt/runtime/
# ls /mnt/runtime/
bin         home        media       root        sys
boot        lib         mnt         sbin        tmp
dev         lib64       opt         selinux     usr
etc         lost+found  proc        srv         var
```

3. Type **exit** to leave the shell and exit rescue mode.

20.6 Set up Memory Hotplugging

Linux *Memory Hotplug* allows the amount of memory in a Linux system to be increased or decreased without a reboot. You must first have standby memory defined to the virtual machine in which Linux is running. You can issue the **CP DEFINE STORAGE** command to configure standby memory (storage). Linux can then exploit the standby memory using the Service Call (**SERVC**) instruction.

To set up standby storage for Linux memory hotplug, using LINUX153 as the virtual machine, perform the following steps.

- Modify the LINUX153 directory entry by adding a **COMMAND** statement. This will give the virtual machine an additional 768 MB of standby memory:

```
USER LINUX153 LNX4VM 256M 1G G
  INCLUDE LNXDFLT
  COMMAND DEFINE STORAGE 256M STANDBY 768M
  OPTION APPLMON
  MDISK 100 3390 3339 3338 UM63A9 MR LNX4VM LNX4VM LNX4VM
  MDISK 101 3390 6677 3338 UM63A9 MR LNX4VM LNX4VM LNX4VM
```

- Bring the changes online with DirMaint, or with **DIRECTXA** if you are not using DirMaint.
- Shutdown the Linux system running on `LINUX153`. This can be done a number of ways, but because you are logged onto MAINT, it can be accomplished with the **SIGNAL SHUTDOWN** command:

```
==> signal shutdown linux153
```

- Within about 30 seconds, you should see notification that the system went down cleanly and the virtual machine was logged off:

```
HCPSIG2113I User LINUX153 has reported successful termination
USER DSC   LOGOFF AS  LINUX153  USERS = 16    AFTER SIGNAL
```

- Logon to `LINUX153`. You should see the standby memory reported:

```
LOGON LINUX153
00: z/VM Version 6 Release 2.0, Service Level 0000 (64-bit),
00: built on IBM Virtualization Technology
00: There is no logmsg data
00: FILES: 0001 RDR,   NO PRT,   NO PUN
00: LOGON AT 07:34:25 EDT SATURDAY 10/08/11
00: Command complete
00: NIC 0600 is created; devices 0600-0602 defined
00: NIC 0600 is connected to VSWITCH SYSTEM VSW1
00: Command complete
00: NIC 0700 is created; devices 0700-0702 defined
00: NIC 0700 is connected to VSWITCH SYSTEM VSW2
00: STORAGE = 256M MAX = 1G INC = 2M STANDBY = 768M RESERVED = 0
00: Storage cleared - system reset.
```

- Answer **y**es to boot Linux:

```
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
y
...
```

- Start an SSH session as root and view the memory in the `/sys/` file system. Change directory to `/sys/devices/system/memory/` and list the files:

```
# cd /sys/devices/system/memory
# ls
block_size_bytes  memory0  memory1  memory2  memory3
```

- Type the `block_size_bytes` file with the **cat** command:

```
# cat block_size_bytes
10000000
```

This number is the number of bytes in hexadecimal. 10000000 in hex is 256 M in decimal. So the block size is 256 MB and there are four blocks: `memory0-memory3`, which are represented as directories. Each of the memory blocks has a state, which is represented as a file.

- Show the state of each memory block with the following command:

```
# cat memory*/state
```

```
online
offline
offline
offline
```

This shows that the first 256 MB is online and the next three blocks are offline.

- You can also show information about memory with the **free -m** command:

```
# free -m
             total       used       free     shared    buffers     cached
Mem:           241        165         75          0         18         54
-/+ buffers/cache:         92        148
Swap:          761          0        761
```

This shows 241 MB of free memory available (some of the memory is used internally by Linux).

- You can turn on memory by sending the string **online** to the state file. Turn on an additional 512 MB of memory with the following commands:

```
# echo online > memory1/state
# echo online > memory2/state
```

- Show that the memory is now online:

```
# cat memory*/state
online
online
online
offline
```

- Again, confirm with the **free -m** command:

```
# free -m
total       used       free     shared    buffers     cached
Mem:           752        167        584          0         11         60
-/+ buffers/cache:         96        655
Swap:         1273          0       1273
```

This shows that 752 MB of free memory is now available - 511 MB more than before.

- You can also give the memory back by echoing **offline** to the state file:

```
# echo offline  > memory1/state
# echo offline  > memory2/state
```

- Verify the memory has be returned:

```
# cat memory*/state
online
offline
offline
offline
# free -m
total       used       free     shared    buffers     cached
Mem:           240        163         76          0         11         60
-/+ buffers/cache:         92        147
Swap:         1273          0       1273
```

This section has shown how to configure virtual machines with standby memory and how to "hot-plug" the memory from Linux. This function can increase your system's performance and availability.

20.7 Utilize the cpuplugd service

The **cpuplugd** service allows Linux to enable or disable CPUs and memory, based on a set of rules. It can improve performance by setting the correct number of processors and amount of memory for Linux systems depending on their current load. It can also prevent the Linux scheduler from queue balancing in partial load situations.

More information on **cpuplugd** can be found in the manual *Linux on System z Device Drivers, Features and Commands* for RHEL and SLES, on the Web at:

http://www.ibm.com/developerworks/linux/linux390/documentation_red_hat.html
http://www.ibm.com/developerworks/linux/linux390/documentation_novell_suse.html

20.7.1 Determine the virtual CPUs being used

To start work with **cpuplugd**, perform the following steps:

- Start an SSH session to a Linux and determine how many CPUs Linux has online. Write a short bash script, **lscpus**, to save typing:

```
# cd /usr/local/sbin
# vi lscpus
#!/bin/bash
# script to list the number and status of virtual CPUs
for i in /sys/devices/system/cpu/cpu*
do
  echo $i
  cat $i/online
done
```

- Save the file and the set it to be executable:

```
# chmod +x lscpus
```

- Observe the status of the **cpuplugd** service:

```
# service cpuplugd status
cpuplugd (pid  1574) is running...
```

  This shows that **cpuplugd** is running

- Wait a few minutes and run the **lscpus** script again:

```
# lscpus
/sys/devices/system/cpu/cpu0
1
/sys/devices/system/cpu/cpu1
0
/sys/devices/system/cpu/cpu2
0
/sys/devices/system/cpu/cpu3
0
/sys/devices/system/cpu/cpu4
0
/sys/devices/system/cpu/cpu5
0
/sys/devices/system/cpu/cpu6
0
/sys/devices/system/cpu/cpu7
0
/sys/devices/system/cpu/cpu8
0
/sys/devices/system/cpu/cpu9
```

```
0
```

The output shows that now only one of the ten virtual CPUs are active. The **cpuplugd** service turned off the other 9.

- The **cpuplugd** configuration file is `/etc/sysconfig/cpuplugd`. Some middleware products recommend a minimum of two virtual processors. If the majority of your Linux servers will be running a workload which recommends two processors, changed the default for `CPU_MIN` to 2. An exception would be when only a single physical processor is available. View the non-comments and lines that are not blank in the configuration file with the following command:

```
# cd /etc/sysconfig
# egrep -v '^$|^#' cpuplugd
CPU_MIN="1"
CPU_MAX="0"
UPDATE="10"
CMM_MIN="0"
CMM_MAX="8192"
CMM_INC="256"
HOTPLUG="(loadavg > onumcpus + 0.75) & (idle < 10.0)"
HOTUNPLUG="(loadavg < onumcpus - 0.25) | (idle > 50)"
MEMPLUG="0"
MEMUNPLUG="0"
```

The default rules for the plugging and unplugging of CPUs in the configuration file is as follow:

```
HOTPLUG = "(loadavg > onumcpus +0.75) & (idle < 10.0)"
HOTUNPLUG = "(loadavg < onumcpus -0.25) | (idle > 50)"
```

Where the variables in the statements have the following meaning:

`loadavg`	The current average CPU load
`onumcpus`	The number of CPUs that are online
`runable_proc`	The current number of processes that can be run
`idle`	The current idle percentage

These CPU hot plugging and unplugging values will be used in the next section. In the default setup, **cpuplugd** will only make changes to the virtual processor configuration. The auto adaptive adjustment of the memory using the `cmm` feature (module) is deactivated by default and also not available when running in a native LPAR environment.

20.7.2 Generating a workload to see cpuplugd work

You can now generate a workload to show how the **cpuplugd** will turn on CPUs.

> **Important:** Running the following command will generate significant CPU use. Verify there is not a mission-critical workload running on this z/VM LPAR, as this test may affect it. Also, be sure to kill the processes after seeing **cpuplugd** in action.

Perform the following steps:

- Put ten looping jobs in the background with the following **for** loop:

```
# for i in `seq 1 10`
> do
>   bash -c "cat /dev/zero > /dev/null" &
> done
[1] 2441
[2] 2442
```

```
[3] 2443
[4] 2444
[5] 2445
[6] 2446
[7] 2447
[8] 2448
[9] 2449
[10] 2453
```

- See that the jobs are running (you can also use the **top** command):

```
# pstree -G | grep cat
    +-sshd---sshd---bash---10*[bash---cat]
```

- Now run **lscpus** every so often. The following example shows that, after a minute or so, **cpuplugd** has started five of the nine spare processors.

```
# lscpus
/sys/devices/system/cpu/cpu0
1
/sys/devices/system/cpu/cpu1
1
/sys/devices/system/cpu/cpu2
1
/sys/devices/system/cpu/cpu3
1
/sys/devices/system/cpu/cpu4
1
/sys/devices/system/cpu/cpu5
1
/sys/devices/system/cpu/cpu6
0
/sys/devices/system/cpu/cpu7
0
/sys/devices/system/cpu/cpu8
0
/sys/devices/system/cpu/cpu9
0
```

After a few more minutes, all of the CPUs should be activated.

- Kill the processes with the **killall** command, then verify that the loops have stopped:

```
# killall cat
bash: line 1:  2450 Terminated              cat /dev/zero > /dev/null
bash: line 1:  2452 Terminated              cat /dev/zero > /dev/null
bash: line 1:  2451 Terminated              cat /dev/zero > /dev/null
bash: line 1:  2457 Terminated              cat /dev/zero > /dev/null
bash: line 1:  2456 Terminated              cat /dev/zero > /dev/null
[1]   Exit 143                bash -c "cat /dev/zero > /dev/null"
[2]   Exit 143                bash -c "cat /dev/zero > /dev/null"
...
# pstree -G | grep cat
```

No output shows that the processes to create a workload have been stopped.

20.7.3 Setting memory sizes with cpuplugd

Memory sizes can also be set by the **cpuplugd** service. However, unlike CPUs, there is no good generic default value. The following example is in the *Device Drivers* book:

```
MEMPLUG = "swaprate > freemem+10 & freemem+10 < apcr"
MEMUNPLUG = "swaprate < freemem + 10000"
```

However, this is just a starting point to explain the syntactical structure of a rule. Do not use this configuration in production. You should test any setting that you want to implement against a representative workload that your Linux systems will be running. Details are beyond the scope of this section.

20.8 Hardware cryptographic support for OpenSSH

This section shows how to copy a test file with OpenSSH, first without any crypto acceleration. Then crypto acceleration for OpenSSH is enabled and the same file is copied again. A much higher throughput rate should be observed. The prerequisite for using hardware cryptography is to have a firmware level of LIC 3863 installed on your System z CEC.

> **Important:** This section was completed successfully with RHEL 6.2. When trying it with SLES 11 SP2, an error was encountered whereby SSH sessions would no longer work. If you try this section with SLES 11 SP2, be prepared to possibly do some debugging or back out your changes to `/etc/ssl/openssl.cnf`.

This section is based on the white paper *First experiences with hardware cryptographic support for OpenSSH with Linux for System z*, by Manfred Gnirss, Winfried Münch, Klaus Werner and Arthur Winterling, on the Web at:

http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101690

This section only shows a single example of crypto acceleration. For a much more complete and detailed analysis, see the white paper.

To test copying a file with and without cryptographic acceleration, perform the following steps:

- Start an SSH session as root to any Linux.
- Create a 200 MB test file for copying in the `/tmp/` directory:

```
# cd /var
# dd if=/dev/zero of=testdata.txt bs=1048576 count=200
200+0 records in
200+0 records out
209715200 bytes (210 MB) copied, 17.87 s, 11.7 MB/s
# ls -lh testdata.txt
-rw-r--r-- 1 root root 200M Dec 28 07:57 testdata.txt
```

- Copy the file locally with the **scp** command, two times with specific encryption algorithms and once without, prefixing all with the **time** command:

```
# time scp -c 3des-cbc testdata.txt localhost:/dev/null
The authenticity of host 'localhost (::1)' can't be established.
RSA key fingerprint is 41:77:58:10:50:09:ba:2a:6a:7b:8b:56:95:1a:37:79.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (RSA) to the list of known hosts.
root@localhost's password:
testdata.txt                                  100%  200MB  13.3MB/s   00:15

real    0m15.943s
user    0m13.732s
sys     0m0.654s

# time scp -c aes128-cbc testdata.txt localhost:/dev/null
testdata.txt                                  100%  200MB  25.0MB/s   00:08
```

```
real    0m10.695s
user    0m5.142s
sys     0m0.470s

# time scp testdata.txt localhost:/dev/null
testdata.txt                                  100%  200MB  28.6MB/s   00:07

real    0m8.219s
user    0m3.700s
sys     0m0.907s
```

The output shows a throughputs of about 13.3, 25 and 28.6 MB/s and a user times of about 13.7, 5.1 and 3.7 seconds.

- Determine if the necessary cryptographic-related RPMs are installed:

```
# rpm -qa | grep openssl-ibmca
```

No output shows that they are not installed.

- **If you are using RHEL 6.2,** install the RPMs with the **yum install** command:

```
# yum -y install openssl-ibmca openssl-ibmca.s390
...
Installed:
  openssl-ibmca.s390 0:1.2.0-2.el6       openssl-ibmca.s390x 0:1.2.0-2.el6

Dependency Installed:
  glibc.s390 0:2.12-1.46.el6          keyutils-libs.s390 0:1.4-3.el6
  krb5-libs.s390 0:1.9-22.el6         libcom_err.s390 0:1.41.12-11.el6
  libselinux.s390 0:2.0.94-5.2.el6    nss-softokn-freebl.s390 0:3.12.9-11.el6
  openssl.s390 0:1.0.0-20.el6         zlib.s390 0:1.2.3-27.el6

Complete!
```

- **If you are using SLES 11 SP2**, install the RPMs with the **zypper install** command:

```
# zypper install openssl-ibmca libica
Loading repository data...
Reading installed packages...
'libica' not found in package names. Trying capabilities.
Resolving package dependencies...

The following NEW packages are going to be installed:
  libica-1_3_9 openssl-ibmca

2 new packages to install.
Overall download size: 179.0 KiB. After the operation, additional 779.0 KiB
will be used.
Continue? [y/n/?] (y): y
Retrieving package libica-1_3_9-1.3.9-6.6.s390x (1/2), 163.0 KiB (732.0 KiB unpacked)
Retrieving package openssl-ibmca-1.2.0-141.10.17.s390x (2/2), 16.0 KiB (47.0 KiB
unpacked)
Installing: libica-1_3_9-1.3.9-6.6 [done]
Additional rpm output:
Updating etc/sysconfig/z90crypt...

Installing: openssl-ibmca-1.2.0-141.10.17 [done]
```

- Verify that the RPMs are now installed:

```
# rpm -qa | egrep "libica|ibmca"           // for RHEL 6.2
libica-2.1.0-2.el6.s390x
openssl-ibmca-1.2.0-2.el6.s390x
```

```
openssl-ibmca-1.2.0-2.el6.s390
# rpm -qa | egrep "libica|ibmca"           // for SLES 11 SP2
libica-1_3_9-1.3.9-6.6
openssl-ibmca-1.2.0-141.10.17
```

- Verify that CP Assist for Cryptographic Function (CPACF) operations are supported:

```
# icainfo
The following CP Assist for Cryptographic Function (CPACF) operations are
supported by libica on this system:
SHA-1:        yes
SHA-256:      yes
SHA-512:      yes
DES:          yes
TDES-128:     yes
TDES-192:     yes
AES-128:      yes
AES-192:      yes
AES-256:      yes
PRNG:         yes
CCM-AES-128:  yes
CMAC-AES-128: yes
CMAC-AES-192: yes
CMAC-AES-256: yes
```

- Make a backup of the SSL configuration file, `/etc/ssl/openssl.cnf`:

```
# cd /etc/pki/tls                      // for RHEL 6.2
# cd /etc/ssl                          // for SLES 11 SP2
# cp openssl.cnf openssl.cnf.orig
```

- Append the sample SSL configuration file under `/usr/share/doc/openssl-ibmca-1.1/` to the actual SSL configuration file, `/etc/openssl.cnf`:

```
# // for RHEL 6.2:
# cat /usr/share/doc/openssl-ibmca-1.2.0/openssl.cnf.sample-s390x >> openssl.cnf
# // for SLES 11 SP2:
# cat /usr/share/doc/packages/openssl-ibmca/openssl.cnf.sample >> openssl.cnf
```

- Edit the appended file and search for the line with the **openssl_conf** variable. Move that line from the bottom to the top and save the file, as shown in the following example:

```
# vi openssl.cnf
/openssl_conf
#
# OpenSSL example configuration file.
# This is mostly being used for generation of certificate requests.
#

# This definition stops the following lines choking if HOME isn't
# defined.
HOME                   = .
RANDFILE               = $ENV::HOME/.rnd
openssl_conf = openssl_def
...
```

- Make a symbolic link to the file `/usr/lib64/openssl/engines/libibmca.so`:

```
# cd /usr/lib64
# ln -s openssl/engines/libibmca.so
# ls -l libibmca.so
lrwxrwxrwx. 1 root root 27 Oct 20 16:47 libibmca.so -> openssl/engines/libibmca.so
```

- Rerun the same **scp** commands:

```
# cd /var
```

```
# time scp -c 3des-cbc testdata.txt localhost:/dev/null
testdata.txt                                  100%  200MB  66.7MB/s   00:03

real    0m4.381s
user    0m1.671s
sys     0m0.579s
# time scp -c aes128-cbc testdata.txt localhost:/dev/null
testdata.txt                                  100%  200MB  66.7MB/s   00:03

real    0m3.357s
user    0m1.048s
sys     0m0.543s
# time scp testdata.txt localhost:/dev/null
Password:
testdata.txt                                  100%  200MB  66.7MB/s   00:03

real    0m4.839s
user    0m0.996s
sys     0m0.548s
```

- Delete the test file:

```
# rm /tmp/testdata.txt
```

You should see an improved throughput as a result of using the crypto hardware.

20.9 The X Window System

For many years UNIX-like operating systems have been using the X Window System (commonly just “X”). This system was designed to provide client/server, hardware-independent and network-enabled graphical environment. The current version is X11 which is widely used on UNIX and Linux platforms.

Confusion often arises among new X users regarding the concept of client and server, because client and server are defined from an application point of view where other protocols such as SSH, Telnet and FTP they are defined from an end user point of view. In X the server runs on the hardware with the mouse, keyboard and monitor (usually a workstation or a desktop), while the client runs on the UNIX or Linux server. Many Linux desktop users don’t recognize this difference because they often run both the server and client on their desktop.

It is a common practice to connect from a PC (SSH client) to remote Linux (SSH server) and then run an X application. It runs on remote Linux (X client) and displays on local PC (X server).

The X communication protocol by its nature is not secure at all. For this reason it is often used together with SSH protocol, which tunnels X11 traffic using encrypted (and thus secure) communications.

X11 itself provides the ability to display graphics on raster display, nothing more. If the user wants to be able to move, resize and otherwise manage windows, a *window manager* is needed. There are many window managers available; some are lightweight while some are more robust. So using a window manager is a good idea because it provides functionality which one expects from a GUI.

When you have Linux installed on your workstation, a window manager is probably not enough. Here you want a full desktop environment with menus, icons, task bars etc such as

Gnome and KDE. Installing GNOME or KDE on System z is discouraged as they are resource-intensive. Installing The X Window System is also not recommended.

20.9.1 VNC Server

As mentioned earlier, the X server is run where the mouse, keyboard and monitor are located - on the workstation. In a nutshell, VNC Server provides virtual workstation with all this peripherals (virtual). The VNC server starts an embedded X server. Then any X-based application can send its output to this X server, regardless of if the applications is local or remote to X server.

To interact with the X server, one uses VNC client on a workstation, as described in section 3.1.2, "Set up a VNC client" on page 29. The VNC server customization is described in section 8.2.4, "Configure the VNC server" on page 147. In our experience this is all you need if you want to run X applications from time to time.

One big advantage of VNC is that it is session oriented. If communication to VNC server is lost, a new connection is reestablished to the session as it was. Also, applications in a disconnected VNC session still continue to run.

20.9.2 X Server on workstation

If for some reason VNC is not acceptable, it is possible to use a standard X server on a workstation. Since Linux users usually know the X Window system, an X server running on Windows is described in this section.

There are many commercial and free X Window servers available for Windows. In the following examples XliveCD is used, which provides a free X server based on Cygwin. It can be run directly from a CD without requiring installation.

http://xlivecd.indiana.edu/

Any X application will send its output to an address defined with **-display** parameter or, if not provided, to an address specified in the `DISPLAY` environment variable. If neither is provided, the local computer is used for output. Following is an example that uses the **xclock** command (you may have to first install it with the command **yum -y install xclock**):

```
gpok224:~ # xclock
Error: Can't open display:
```

There is no display specified for **xclock** command and it will terminate.

Display is specified by setting DISPLAY environment variable.

```
gpok224:~ # export DISPLAY=9.145.177.158:0
gpok224:~ # xclock
No protocol specified
Error: Can't open display: 9.145.177.158:0
```

This command failed, because the XliveCD requires an explicit command to allow remote hosts to connect to it. When the command **xhost +** (plus means to add authorized hosts) is run, **xclock** can finally display on Windows as shown in Figure 20-2. Remember the program itself runs on a remote Linux.

```
gpok224:~ # xclock &
[1] 21915
```

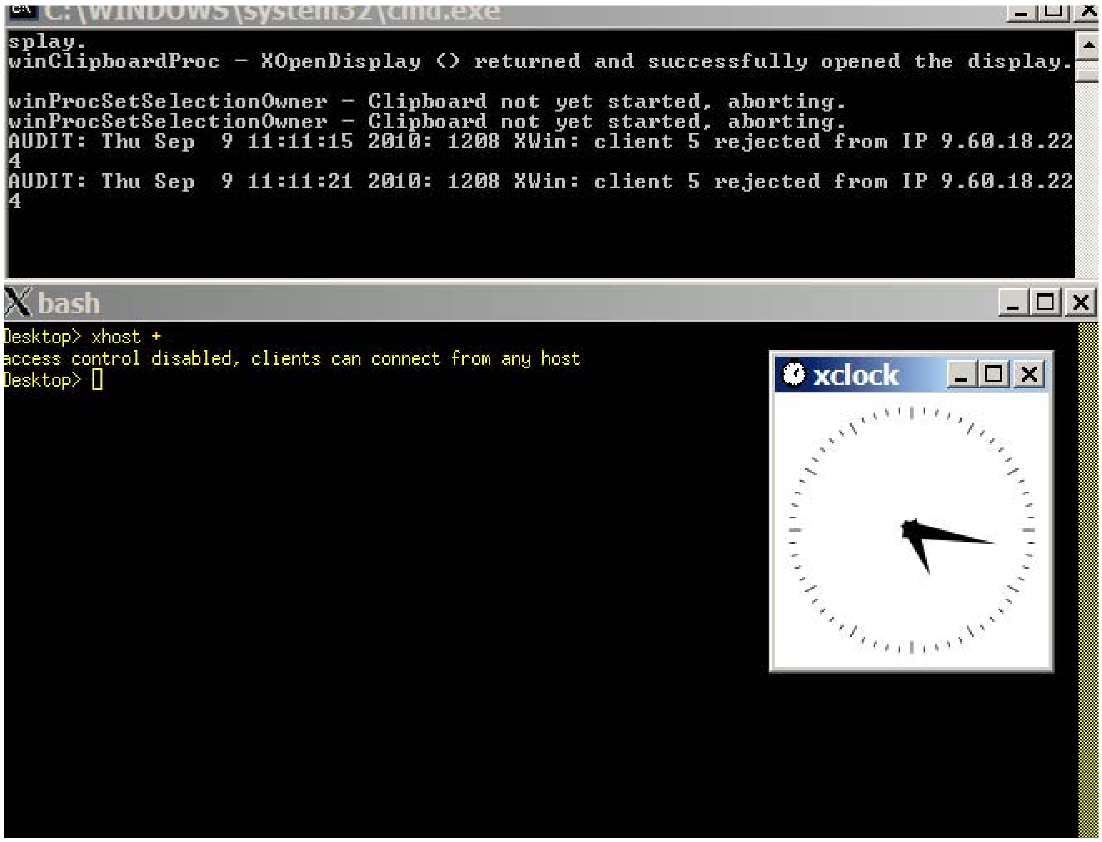


Figure 20-2 Manual setting of DISPLAY variable

The **xhost +** command allows any host to access the X Server. From a security point of view, this may not be a good idea. Even allowing just specific hosts is not enough, because X11 protocol itself is not secure. Using SSH tunneling removes this security exposure. SSH tunneling also prevents firewalls and NAT from breaking X11 communications.

It is possible to use an external SSH client which allows X11 forwarding, or SSH client embedded in XliveCD itself. Both options are shown.

Using PuTTY

To use PuTTY for X11 forwarding, select X11 forwarding as shown in Figure 20-3.

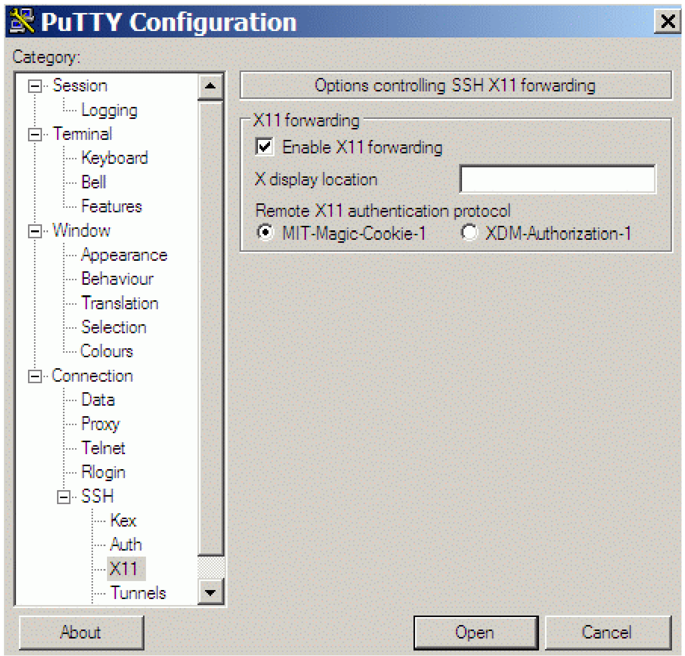


Figure 20-3 Allow X11 Forwarding in PuTTY

As you can see in Figure 20-4, the `DISPLAY` environment variable contains the special value of **`localhost:10.0`** which tells PuTTY to forward X11 protocol over SSH to SSH client address.In this case there is no need to enter **`xhost`** command because the connection appears to X Server as a local one.

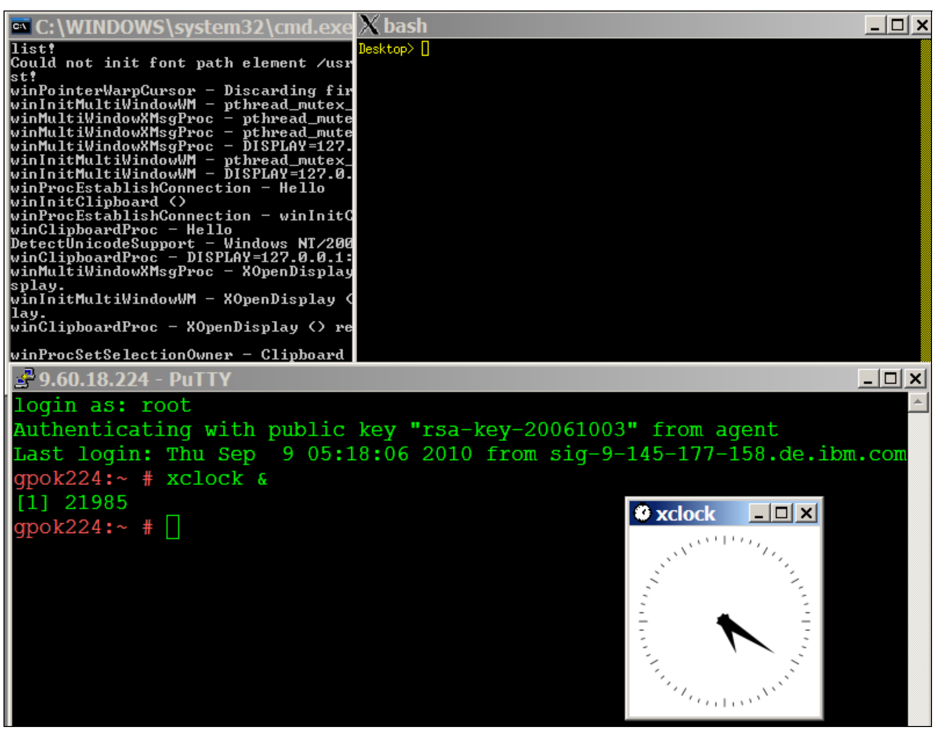


Figure 20-4 X11 forwarding with PuTTY

Using embedded SSH

It is also possible to achieve X11 forwarding with an embedded SSH client as shown below. Again, no **xhost** command is needed.

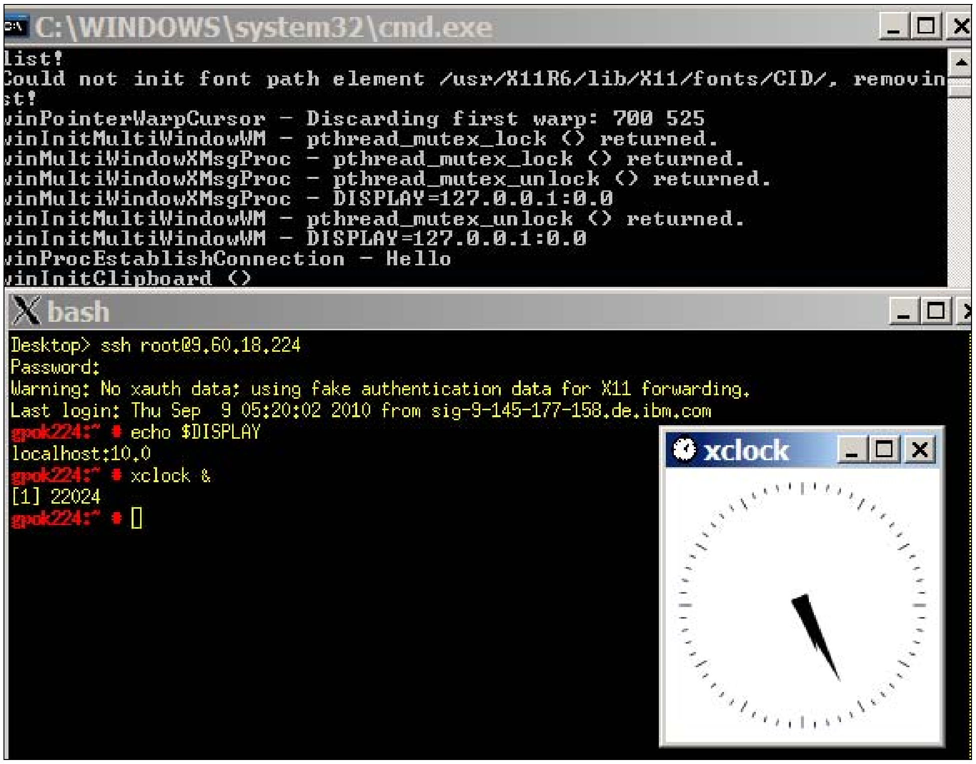


Figure 20-5 X11 forwarding with embedded SSH client

There are many ways how to achieve the same results. It is up to you to choose a solution which suits the purpose best.

20.10 Centralizing home directories for LDAP users

In previous versions of this book there was a section on a travelling `/home/` directory using LDAP, NFS and automount. In the interest of time, this section has been removed. See section 13.3 in the IBM Redbook *z/VM and Linux on IBM System z The Virtualization Cookbook for Red Hat Linux Enterprise Server 5.2*, SG24-7492, on the Web at:

http://www.redbooks.ibm.com/abstracts/sg247492.html

However, the following section has been added to this book. In December of 2009, the topic of how to set up a common home directory came up on the linux-390 list server. The following post by Patrick Spinler is copied, with permission, as it may be helpful to you:

20.10.1 Recommendations for centralizing home directories

“NFSv3 is not known for it's security. Consider the use of the NFS option `root_squash`, along with limiting the list of hosts who can connect to your home share. Only export home directories to hosts which you control, remember that anyone who has root on their box (e.g. a developer workstation) can impersonate any user to NFS. Here's the relevant `/etc/exports` line we use:

```
/export/unixdata/homedirs              \
  @hgrp_autohome_admin(rw,no_root_squash,insecure,sync) \
  @hgrp_autohome_hosts(rw,root_squash,insecure,sync)
```

I look forward to going to NFSv4 with kerberos authentication, but we're not there yet.

Regarding automount maps in LDAP, this works very well for us with one exception. The problem is that there's a significant number of automount map schemas out there, and different operating systems use different ones. As we are a fairly heterogeneous environment, I found it near impossible to keep a master map in LDAP. Right now we're just keeping a `/etc/auto.master` or `/etc/auto_master` on each host.

In order to make the individual map entries work heterogeneously, I had to add several object classes and a few redundant attributes to each entry. Here's what my home directory automount map entry looks like:

```
# ap00375, auto_home, unix.example.com
dn: automountKey=ap00375,automountMapName=auto_home,dc=unix,dc=example,dc=com
automountInformation: linux153.example.com:/vol/vol2/unixhomes-5gb/75/ap00375
cn: ap00375
automountKey: ap00375
objectClass: automount
objectClass: nisNetId
objectClass: top
```

Regarding heterogeneous clients, we found AIX in particular to be the hardest of our clients to configure, and Linux the easiest. Insure on AIX that you have the latest available LDAP client package from IBM. Also be aware that AIX wants to use it's extended LDAP schema rather than RFC2307, and wants full write access to the LDAP servers from every AIX client. Despite that, it will work with RFC2307 and read only access. Solaris, like Linux, has an option to not use an LDAP proxy account at all via anonymous binding, but I never got Solaris anonymous binding to work.

I recommend making LDAP use TLS or SSL on the wire, in order to keep clear-text passwords from flying about. Both AIX and Solaris require the server public SSL certificates to be loaded on every client to do LDAP over TLS or SSL. Linux can be configured to ignore authenticating the LDAP servers' certificates and proceed with TLS/SSL anyway - this is convenient, but does open the possibility of man in the middle attacks. In our environment this isn't a big deal, but it might be in yours.

We've found POSIX group membership management to be one of our more challenging issues overall. Some older systems (e.g. solaris <= 8 or 9) enforce the old POSIX limit of no more than 16 secondary groups. Further, the primary group concept is annoying - conceptually, in any organization with modest member mobility, which primary group do they get? If one assumes that the primary group is meaningful, e.g. reflective of someone's function, role, or job, what about people who do two or more things (E.g. student *and* employee) or people who transfer, but will have a transitional period?

Our not so great compromise was to first use NIS-style netgroups via LDAP for anything we can. In particular, we use a mutation of netgroups to control individual's authorization to log in via the use of service search descriptors, and also for sudo privileges. Second in our environment all meaningful POSIX groups are secondary groups. For primary groups we adopt the linux convention of creating a separate POSIX group for each individual: e.g. userA gets a group userA as her primary group. This has the problem of a huge proliferation of groups, though, and several LDAP clients, in particular AIX, have issues with that."

21

xCAT

"The only source of knowledge is experience."

--Albert Einstein

This chapter describes how to install, configure and use xCAT. It consists of the following sections:

- "Overview of xCAT"
- "Install the xCAT Management Node" on page 358
- "Install the xCAT User Interface" on page 362
- "Install the System z xCAT Hardware Control Point" on page 364

21.1 Overview of xCAT

xCAT stands for eXtreme Cloud Administration Toolkit (previously, the "C" stood for Cluster). It deploys virtual machines and operating systems. It performs hardware control to boot, power on, boot, power off and provide remote consoles. xCAT captures images and has support for diskless images and can build systems using RHEL **kickstart** or SLES **autoYast**. It performs software and hardware inventory. It allows shell commands to be sent in parallel to many virtual machines, as well as parallel copies and pings. It was designed to scale to thousands of nodes. As such, it is lightweight. It was "open-sourced" in 2007, licensed under the Eclipse Public License (EPL) and is available on sourceForge.

There is one xCAT Management Node (MN) on Linux per enterprise. There is one Hardware Control Point (HCP) on Linux per z/VM LPAR (or SSI member). The HCP uses SMAPI to communicate with the directory management product and invokes CP commands to query information about virtual machines and networks. Communication between the MN and the HCPs is over SSH and thus is secure.

This section describes installing two HCPs onto the two Linux administration servers on each SSI member and one MN also onto one of the Linux administration servers. On distributed servers, it is not common for the MN to be installed onto the same system as the HCP, but it is done here for simplicity.

The xCAT for z/VM documentation is on the Web at:

http://sourceforge.net/apps/mediawiki/xcat/index.php?title=XCAT_zVM

This chapter is similar, however, it is in the context of a larger book, not a wiki, and there are more details about configuring the entire system, especially DirMaint and SMAPI.

21.2 Install the xCAT Management Node

Because the LNXADMIN virtual machine on the SSI member 1 has a large file system mounted over /var/, this is the logical place to install the xCAT management node.

To install the xCAT Management Node, perform the following steps:

- "Turn off SELinux on RHEL 6.2" on page 358
- "Download xCAT management node install files" on page 359
- "Unwind the xCAT management node install files" on page 359
- "Create repositories for the xCAT code" on page 359
- "Install the xCAT management node" on page 360

21.2.1 Turn off SELinux on RHEL 6.2

If you are using RHEL 6.2, it is recommended that Security-Enhanced Linux (SELinux) be modified from a state of enforcing to one of permissive. To do so, perform the following steps:

- Start an SSH session as root with the Linux system running on LNXADMIN on SSI member 1.
- Change directory to /etc/selinux/:

```
# cd /etc/selinux
```

- View the state of SE Linux:

```
# cat config

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

The value of SELINUX=**enforcing** shows that SE Linux is working.

- Make a copy of the original configuration file:

```
# cp config config.orig
```

- Modify the config file by changing the variable SELINUX to **permissive**:

```
# vi config
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=permissive
```

```
...
```

This will turn off SE Linux at boot time.

- Turn off SE Linux for this session:

```
# echo 0 > /selinux/enforce
```

SE Linux should now be turned off across reboots and for this session.

21.2.2 Download xCAT management node install files

There are two xCAT tar files - the “core” file which is common for all architectures and the “dep” file which contains platform dependent code.

Download both files as follows:

- Point a browser to the following Web site:

  `http://xcat.sourceforge.net/#download`

- Download the “core” file which should be at the top of the page. In this example, the file `xcat-core-2.6.9.tar.bz2` was clicked on, then downloaded. Download the newest file possible.
- Follow the link *xCAT dep* which should take you to the following Web site:

  `http://sourceforge.net/projects/xcat/files/xcat-dep/2.x_Linux/`

- Download the “dep” file. In this example, the file `xcat-dep-201112270610.tar.bz2` was used. Download the newest file possible.
- Copy the files to the Linux administration system `/usr/local/src/` directory.
- List the files. In this example, the listing was as follows:

```
# cd /usr/local/src
# ls -lh
total 78M
-rw-r--r-- 1 root root 11M Dec 29 09:29 xcat-core-2.6.9.tar.bz2
-rw-r--r-- 1 root root 67M Dec 29 09:21 xcat-dep-201112270610.tar.bz2
```

You should now have the xCAT install files for System z downloaded.

21.2.3 Unwind the xCAT management node install files

Untar and decompress the two files as follows.

- Use the **tar** command with the **x** and **j** flags as follows:

```
# tar xjf xcat-core-2.6.9.tar.bz2
# tar xjf xcat-dep-201112270610.tar.bz
```

- List the contents - you should see two new directories:

```
# ls
xcat-core  xcat-core-2.6.6.tar.bz2  xcat-dep  xcat-dep-201109231753.tar.bz2
```

21.2.4 Create repositories for the xCAT code

To create repositories for the xCAT code, perform the following steps:

- **If you are installing onto RHEL 6.2**, run the following two scripts, then observe the two new repositories have been created in the `/etc/yum.repos.d/` directory

```
# xcat-dep/rh6/s390x/mklocalrepo.sh
/usr/local/src
# xcat-core/mklocalrepo.sh
/usr/local/src
# ls /etc/yum.repos.d/
redhat.repo  rhel-source.repo  rhel62.repo  xCAT-core.repo  xCAT-dep.repo
```

- **If you are installing onto SLES 11 SP2**, run the following **zypper ar** (add repository) commands, then observe the two new repositories have been created:

```
# zypper ar file:///usr/local/src/xcat-dep/sles11/s390x xCAT-dep
Adding repository 'xCAT-dep' [done]
Repository 'xCAT-dep' successfully added
Enabled: Yes
Autorefresh: No
GPG check: Yes
URI: file:/usr/local/src/xcat-dep/sles11/s390x
# zypper ar file:///usr/local/src/xcat-core xcat-core
Adding repository 'xcat-core' [done]
Repository 'xcat-core' successfully added
Enabled: Yes
Autorefresh: No
GPG check: Yes
URI: file:/usr/local/src/xcat-core
# zypper repos | cut -c 1-80
# | Alias                                        | Name
--+----------------------------------------------+-------------------------------
1 | SUSE-Linux-Enterprise-Server-11-SP2_11.2.2- | SUSE-Linux-Enterprise-Server-1
2 | xCAT-dep                                     | xCAT-dep
3 | xcat-core                                    | xcat-core
```

This shows that two new **zypper** repositories were created for xCAT.

21.2.5 Install the xCAT management node

To install the xCAT management node, perform the following steps:

Install the management node on RHEL 6.2

- **If you are installing onto RHEL 6.2**, use the following **yum** command to clean up metadata:

```
# yum clean metadata
Loaded plugins: product-id, security, subscription-manager
Updating certificate-based repositories.
Cleaning repos: RHEL62 xcat-2-core xcat-dep
4 metadata files removed
2 sqlite files removed
0 metadata files removed
```

- Use the following **yum** command to install xCAT:

```
# yum -y install xCAT
...
Installed:
  xCAT.s390x 0:2.6.6-snap201108100710

Dependency Installed:
  apr.s390x 0:1.3.9-3.el6_1.2
  apr-util.s390x 0:1.3.9-3.el6_0.1
  apr-util-ldap.s390x 0:1.3.9-3.el6_0.1
  bind.s390x 32:9.7.3-7.P3.el6
```

```
  dhcp.s390x 12:4.1.1-24.P1.el6
  expect.s390x 0:5.44.1.15-2.el6
  fping.s390x 0:2.4b2_to-2
  httpd.s390x 0:2.2.15-13.el6
  httpd-tools.s390x 0:2.2.15-13.el6
  ksh.s390x 0:20100621-9.el6
  net-snmp-perl.s390x 1:5.5-37.el6
  nmap.s390x 2:5.21-4.el6
  perl-Digest-HMAC.noarch 0:1.01-22.el6
  perl-Digest-SHA1.s390x 0:2.12-2.el6
  perl-Expect.noarch 0:1.21-1
  perl-IO-Socket-SSL.noarch 0:1.31-2.el6
  perl-IO-Tty.s390x 0:1.07-1
  perl-Net-DNS.s390x 0:0.65-4.el6
  perl-Net-LibIDN.s390x 0:0.12-3.el6
  perl-Net-SSLeay.s390x 0:1.35-9.el6
  perl-SOAP-Lite.noarch 0:0.710.08-1
  perl-XML-LibXML.s390x 1:1.70-5.el6
  perl-XML-NamespaceSupport.noarch 0:1.10-3.el6
  perl-XML-SAX.noarch 0:0.96-7.el6
  perl-XML-Simple.noarch 0:2.18-1
  perl-xCAT.noarch 4:2.6.6-snap201108160156
  tcl.s390x 1:8.5.7-6.el6
  vsftpd.s390x 0:2.2.2-6.el6_0.1
  xCAT-client.noarch 4:2.6.6-snap201108100709
  xCAT-server.noarch 4:2.6.6-snap201108120732

Complete!
```

The xCAT management node should now be installed on your RHEL 6.2 system.

Install the management node on SLES 11 SP2

If you are on SLES 11 SP2, perform the following steps to install the management node:

- Install the DHCP server with the following **zypper** command:

```
# zypper install dhcp-server
New repository or package signing key received:
Key ID: 20E475A8DA736C68
Key Name: Jarrod Johnson <jbjohnso@us.ibm.com>
Key Fingerprint: 47AE47C50C45D502AEA2A47520E475A8DA736C68
Key Created: Fri Jun 27 08:30:03 2008
Key Expires: (does not expire)
Repository: xCAT-dep

Do you want to reject the key, trust temporarily, or trust always? [r/t/a/?] (r): a
Building repository 'xCAT-dep' cache [done]
Building repository 'xcat-core' cache [done]
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW packages are going to be installed:
  dhcp dhcp-server

2 new packages to install.
Overall download size: 1.5 MiB. After the operation, additional 4.2 MiB will be
used.
Continue? [y/n/?] (y): y
Retrieving package dhcp-4.2.2-0.6.16.s390x (1/2), 610.0 KiB (1.6 MiB unpacked)
Retrieving package dhcp-server-4.2.2-0.6.16.s390x (2/2), 939.0 KiB (2.6 MiB unpacked)
```

```
Installing: dhcp-4.2.2-0.6.16 [done]
Installing: dhcp-server-4.2.2-0.6.16 [done]
Additional rpm output:
Updating etc/sysconfig/dhcpd...
Updating etc/sysconfig/syslog...
```

- Install the xCAT management node with the **zypper** command

```
# zypper install xCAT
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW packages are going to be installed:
  apache2 apache2-prefork apache2-utils bind bind-chrootenv libapr-util1
  libapr1 libdnet1 liblua5_1 net-snmp nfs-kernel-server nmap perl-Crypt-SSLeay
  perl-DBD-SQLite perl-DBI perl-Expect perl-HTML-Parser perl-HTML-Tagset
  perl-IO-Socket-SSL perl-IO-Stty perl-IO-Tty perl-Net-DNS perl-Net-Daemon
  perl-Net-IP perl-Net-SSLeay perl-PlRPC perl-SNMP perl-SOAP-Lite
  perl-TermReadKey perl-libwww-perl perl-xCAT vsftpd xCAT xCAT-client
  xCAT-server

The following recommended packages were automatically selected:
  apache2-prefork perl-IO-Socket-SSL

The following packages are not supported by their vendor:
  perl-SOAP-Lite perl-xCAT xCAT xCAT-client xCAT-server

35 new packages to install.
Overall download size: 15.5 MiB. After the operation, additional 36.4 MiB will
be used.
Continue? [y/n/?] (y): y
...
Created xCAT certificate.
Updated cluster site definition.
Restarting xCATd
 Shutting down vsftpd ..done
 Starting vsftpd ..done
..done
httpd has been restarted.
xCAT is now running, it is recommended to tabedit networks
and set a dynamic ip address range on any networks where nodes
are to be discovered. Then, run makedhcp -n to create a new dhcpd
configuration file, and /etc/init.d/dhcpd restart. Either examine sample
configuration templates, or write your own, or specify a value per
node with nodeadd or tabedit.
```

The xCAT management node should now be installed on the SLES 11 SP2 Linux administration system.

21.3 Install the xCAT User Interface

The xCAT user interface provides a user interface through a Web browser.

To install the xCAT management node user interface, perform the following steps:

Install the management node user interface on RHEL 6.2

If you are on RHEL 6.2, perform the following steps:

- Install the co-requisite packages:

```
# yum -y install php php-pear
...
Installed:
  php.s390x 0:5.3.3-3.el6            php-pear.noarch 1:1.9.0-2.el6

Dependency Installed:
  php-cli.s390x 0:5.3.3-3.el6          php-common.s390x 0:5.3.3-3.el6

Complete!
# yum -y install xCAT-UI
...
Installed:
  xCAT-UI.noarch 0:2.6.6-snap201108070844

Complete!
```

The xCAT management node user interface is now installed on RHEL 6.2.

Install the management node user interface on SLES 11 SP2

If you are on SLES 11 SP2, perform the following steps:

- Install the co-requisite packages:

```
# zypper install php5-openssl apache2-mod_php5
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW packages are going to be installed:
  apache2-mod_php5 libmm14 php5 php5-ctype php5-dom php5-hash php5-iconv
  php5-json php5-openssl php5-suhosin php5-tokenizer php5-xmlreader
  php5-xmlwriter

The following recommended packages were automatically selected:
  php5-ctype php5-dom php5-hash php5-iconv php5-json php5-suhosin
  php5-tokenizer php5-xmlreader php5-xmlwriter

13 new packages to install.
Overall download size: 2.5 MiB. After the operation, additional 7.1 MiB will be
used.
Continue? [y/n/?] (y): y
...
```

- Install the xCAT UI:

```
# zypper install xCAT-UI
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW packages are going to be installed:
  xCAT-UI xCAT-UI-deps

The following packages are not supported by their vendor:
  xCAT-UI xCAT-UI-deps

2 new packages to install.
Overall download size: 479.0 KiB. After the operation, additional 1.5 MiB will
be used.
Continue? [y/n/?] (y): y
```

```
Retrieving package xCAT-UI-deps-2.7-1.noarch (1/2), 380.0 KiB (1.2 MiB unpacked)
Retrieving package xCAT-UI-2.6.6-snap201108070844.noarch (2/2), 99.0 KiB (294.0 KiB
unpacked)
Installing: xCAT-UI-deps-2.7-1 [done]
Installing: xCAT-UI-2.6.6-snap201108070844 [done]
Additional rpm output:
To use xCAT-UI, point your browser to http://gpok152.endicott.ibm.com/xcat
Reload httpd2 (graceful restart)..done
```

The xCAT management node user interface is now installed on SLES 11 SP2.

21.4 Install the System z xCAT Hardware Control Point

The xCAT Hardware Control Point (HCP) needs to be installed on each member in the SSI cluster.

Be sure that DirMaint has been configured as described in Chapter 18, “Configure DirMaint, SMAPI and RACF” on page 273.

Install the hardware control point on RHEL 6.2

To install the xCAT HCP on RHEL 6.2, perform the following steps:

- For RHEL 6.2, install the GNU `gcc` and `gcc-c++` RPMs with the following **yum** command:

```
# yum -y install gcc gcc-c++
...
Installed:
  gcc.s390x 0:4.4.6-3.el6              gcc-c++.s390x 0:4.4.6-3.el6

Dependency Installed:
  cloog-ppl.s390x 0:0.15.7-1.2.el6        cpp.s390x 0:4.4.6-3.el6
  glibc-devel.s390x 0:2.12-1.43.el6       glibc-headers.s390x 0:2.12-1.43.el6
  kernel-headers.s390x 0:2.6.32-202.el6   libstdc++-devel.s390x 0:4.4.6-3.el6
  mpfr.s390x 0:2.4.1-6.el6                ppl.s390x 0:0.10.2-11.el6

Complete!
```

- Change to the following directory:

```
# cd /usr/local/src/xcat-dep/rh6/s390x/
```

- Install the xCAT hardware control point with the following RPM command:

```
# rpm -ivh zhcp-1.2-1.s390x.rpm
Preparing...                ########################################### [100%]
   1:zhcp                   ########################################### [100%]
```

Install the hardware control point on SLES 11 SP2

To install the xCAT HCP on SLES 11 SP2, perform the following steps:

- Install the GNU `gcc` and `gcc-c++` RPMs with the following **zypper** command:

```
# zypper install gcc gcc-c++
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following NEW packages are going to be installed:
  gcc gcc-c++ gcc43 gcc43-c++ glibc-devel libgomp46 libstdc++43-devel
  linux-kernel-headers
```

```
8 new packages to install.
Overall download size: 12.0 MiB. After the operation, additional 79.5 MiB will
be used.
Continue? [y/n/?] (y): y
...
```

- Change to the following directory:

```
# cd /usr/local/src/xcat-dep/sles11/s390x/
```

- Install the System z xCAT hardware control point with the following RPM command:

```
# rpm -ivh zhcp-1.2-1.s390x.rpm
Preparing...                ########################################### [100%]
   1:zhcp                   ########################################### [100%]
```

21.4.1 Add a privilege class to LNXADMIN

You may choose to add the `A` privilege class so xCAT can issue the `FORCE` command. The xCAT **`rpower`** command will force off specified virtual machines.

> **Important:** When xCAT forces off virtual machines, the Linux system is not shut down cleanly, so the following command is **not recommended** unless it is acceptable to crash your Linux servers. Rather you should use the **`rpower softoff`** command which will not require the **`CP FORCE`** command, and will shut down Linux systems cleanly.

```
# dirm for lnxadmin class abcdeg
DVHXMT1191I Your CLASS request has been sent for processing to
DVHXMT1191I DIRMAINT at POKDEV62.
Ready; T=0.01/0.01 10:14:13
DVHREQ2288I Your CLASS request for LNXADMIN at * has been accepted.
DVHBIU3450I The source for directory entry LNXADMIN has been updated.
DVHBIU3424I The next ONLINE will take place immediately.
DVHRLA3891I Your DSATCTL request has been relayed for processing.
DVHBIU3428I Changes made to directory entry LNXADMIN have been placed
DVHBIU3428I online.
DVHREQ2289I Your CLASS request for LNXADMIN at * has completed; with
DVHREQ2289I RC = 0.
```

21.4.2 Initialize the xCAT database

The xCAT database contains many relational database tables. Some of the more important are the following:

`nodehm`	Determines which platform specific “plugin” to use
`nodetype`	Determines which Operating System to install onto nodes
`passwd`	Determines how to login
`hosts`	Determines how to associate IP addresses and host names
`networks`	Determines the network infrastructure
`site`	Determines how to configure the management node
`zvm`	Determines System z specific attributes

To populate the database, perform the following steps:

- Start a new SSH session as `root` to the system where you just installed the Hardware Control Point. The new session will include xCAT directories in root’s `PATH`. Note that two new directories, `/opt/xcat/bin` and `/opt/xcat/sbin`, have been added to your `PATH`:

```
# echo $PATH
```

```
/opt/xcat/bin:/opt/xcat/sbin:/sbin:/usr/sbin:/usr/local/sbin:/root/bin:/usr/local/bin:/u
sr/bin:/bin:/usr/bin/X11:/usr/X11R6/bin:/usr/games:/usr/lib/mit/bin:/usr/lib/mit/sbin
```

- Add a passwd table entry to contain the default root password for new virtual machines.

```
# chtab key=system passwd.username=root passwd.password=lnx4vm
```

- Use the **tabdump passwd** command to show that the encrypted root password was added:

```
# tabdump passwd
#key,username,password,cryptmethod,comments,disable
"xcat","root","$6$akOYJKvG$F.QYS1QnyrKOUsj6zyGd4iin/PJ5u/1k3dR7t42LEnBP7ZyM/WQr1idwrrycj
aPtYABpJr3cm4nnkW5GOfnij1",,,
"system","root","lnx4vm",,,
```

- Populate the hosts table with the host names to be used for all Linux systems with the **chtab** command. In this example the values were obtained from Table 2-6 on page 18. If you populated the blank worksheets, consult Table 2-11 on page 23. A bash **for** loop is used:

```
# for i in 144 145 151 152 153 157
> do
>   chtab node=gpok$i hosts.ip="9.60.18.$i" hosts.hostnames="gpok$i.endicott.ibm.com"
> done
```

- Use the **tabdump** command to display the newly populated hosts table:

```
# tabdump hosts
#node,ip,hostnames,otherinterfaces,comments,disable
"gpok144","9.60.18.144","gpok144.endicott.ibm.com",,,
"gpok145","9.60.18.145","gpok145.endicott.ibm.com",,,
"gpok151","9.60.18.151","gpok151.endicott.ibm.com",,,
"gpok152","9.60.18.152","gpok152.endicott.ibm.com",,,
"gpok153","9.60.18.153","gpok153.endicott.ibm.com",,,
"gpok157","9.60.18.157","gpok157.endicott.ibm.com",,,
```

- Set up the network table with the common IP networking values. In this example the values were obtained from Table 2-6 on page 18. If you populated the blank worksheets, consult Table 2-11 on page 23. Note that the following three lines are one long command:

```
# chtab net=9.60.18.128 networks.mask=255.255.255.128 networks.mgtifname=eth1 \
networks.gateway=9.60.18.129 networks.dhcpserver=9.60.18.151 \
networks.tftpserver=9.60.18.151 networks.nameservers=9.60.18.151
```

- Use the **tabdump** command to display the newly populated networks table:

```
# tabdump networks
#netname,net,mask,mgtifname,gateway,dhcpserver,tftpserver,nameservers,ntpservers,logserv
ers,dynamicrange,nodehostname,ddnsdomain,vlanid,domain,comments,disable
"9_60_18_128-255_255_255_128","9.60.18.128","255.255.255.128","eth1","9.60.18.129","9.60
.18.151","9.60.18.151","9.60.18.151",,,,,,,,,
```

- Set up the site table:

```
# chtab key=dhcpinterfaces site.value='all|eth1'
# chtab key=nameservers site.value='9.60.18.151'
# chtab key=master site.value='9.60.18.151'
```

- Use the **tabdump** command to display the newly populated site table:

```
# tabdump site
#key,value,comments,disable
"blademaxp","64",,
"domain","endicott.ibm.com",,
"fsptimeout","0",,
"installdir","/install",,
"ipmimaxp","64",,
"ipmiretries","3",,
```

```
"ipmitimeout","2",,
"consoleondemand","no",,
"master","9.60.18.151",,
"forwarders","9.0.3.1",,
"nameservers","9.60.18.151",,
"maxssh","8",,
"ppcmaxp","64",,
"ppcretry","3",,
"ppctimeout","0",,
"sharedtftp","1",,
"SNsyncfiledir","/var/xcat/syncfiles",,
"tftpdir","/tftpboot",,
"xcatdport","3001",,
"xcatiport","3002",,
"xcatconfdir","/etc/xcat",,
"timezone","America/New_York",,
"useNmapfromMN","no",,
"enableASMI","no",,
"db2installloc","/mntdb2",,
"databaseloc","/var/lib",,
"sshbetweennodes","ALLGROUPS",,
"dnshandler","ddns",,
"vsftp","y",,
"cleanupxcatpost","no",,
"dhcpinterfaces","all|eth1",,
```

21.4.3 Define nodes

The next step is to add nodes to the xCAT database. The two golden images, RH62GOLD and S112GOLD, the two Linux administration systems, LNXADMIN, and the two target virtual machines, LINUX153 and LINUX157, are added with the following **mkdef** commands:

- Make definitions for six systems with the following **mkdef** command:

```
# mkdef -t node -o gpok144 userid=RH62GOLD hcp=gpok151.endicott.ibm.com mgt=zvm
groups=all
1 object definitions have been created or modified.
# mkdef -t node -o gpok145 userid=S112GOLD hcp=gpok151.endicott.ibm.com mgt=zvm
groups=all
1 object definitions have been created or modified.
# mkdef -t node -o gpok151 userid=LNXADMIN hcp=gpok151.endicott.ibm.com mgt=zvm
groups=all
1 object definitions have been created or modified.
# mkdef -t node -o gpok152 userid=LNXADMIN hcp=gpok152.endicott.ibm.com mgt=zvm
groups=all
1 object definitions have been created or modified.
# mkdef -t node -o gpok153 userid=LINUX153 hcp=gpok151.endicott.ibm.com mgt=zvm
groups=all
1 object definitions have been created or modified.
# mkdef -t node -o gpok157 userid=LINUX157 hcp=gpok151.endicott.ibm.com mgt=zvm
groups=all
1 object definitions have been created or modified.
```

- Use the **lsdef** command to show the new node object:

```
# lsdef all
Object name: gpok144
    groups=all
    hcp=gpok151.endicott.ibm.com
    mgt=zvm
    postbootscripts=otherpkgs
```

```
    postscripts=syslog,remoteshell,syncfiles
    userid=RH62GOLD
Object name: gpok145
    groups=all
    hcp=gpok151.endicott.ibm.com
    mgt=zvm
    postbootscripts=otherpkgs
    postscripts=syslog,remoteshell,syncfiles
    userid=S112GOLD
Object name: gpok151
    groups=all
    hcp=gpok151.endicott.ibm.com
    mgt=zvm
    postbootscripts=otherpkgs
    postscripts=syslog,remoteshell,syncfiles
    userid=LNXADMIN
Object name: gpok152
    groups=all
    hcp=gpok152.endicott.ibm.com
    mgt=zvm
    postbootscripts=otherpkgs
    postscripts=syslog,remoteshell,syncfiles
    userid=LNXADMIN
Object name: gpok153
    groups=all
    hcp=gpok151.endicott.ibm.com
    mgt=zvm
    postbootscripts=otherpkgs
    postscripts=syslog,remoteshell,syncfiles
    userid=LINUX153
Object name: gpok157
    groups=all
    hcp=gpok151.endicott.ibm.com
    mgt=zvm
    postbootscripts=otherpkgs
    postscripts=syslog,remoteshell,syncfiles
    userid=LINUX157
```

- Use the **makehosts** command to update the /etc/hosts file:

```
# makehosts
```

- View the updated /etc/hosts file:

```
# cat /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
9.60.18.144 gpok144 gpok144.endicott.ibm.com
9.60.18.145 gpok145 gpok145.endicott.ibm.com
9.60.18.151 gpok151 gpok151.endicott.ibm.com
9.60.18.157 gpok157 gpok157.endicott.ibm.com
9.60.18.152 gpok152 gpok152.endicott.ibm.com
9.60.18.153 gpok153 gpok153.endicott.ibm.com
```

- Set up key-based authentication from the management node to the hardware control point. Because both the MN and the HCP are the same system, this sets up key-based authentication to “itself”, but this step is required:

```
# xdsh gpok151 -K
Enter the password for the userid: root on the node where the ssh keys
will be updated:

/usr/bin/ssh setup is complete.
```

```
return code = 0
```

21.4.4 Configure networking servers

To configure the DHCP and DNS servers, perform the following steps:

- Add networks into the DHCP configuration:

```
# makedhcp -n
Renamed existing dhcp configuration file to  /etc/dhcp/dhcpd.conf.xcatbak

The dhcp server must be restarted for OMAPI function to work
Warning: No dynamic range specified for 9.60.18.128. If hardware discovery is being
used, a dynamic range is required.
```

  Don't worry about the warning message.

- Restart the DHCP server with the following **service** command:

```
# service dhcpd restart
Shutting down dhcpd: [  OK  ]
Starting dhcpd: [  OK  ]
```

- Add the relevant networks to DHCP with the **makedhcp -a** command:

```
# makedhcp -a
```

- Set up the DNS server with the **makedns** command:

```
# makedns
Getting reverse zones, this may take several minutes in scaling cluster.
Completed getting reverse zones.
Updating zones.
Completed updating zones.
Restarting named
Restarting named complete
Updating DNS records, this may take several minutes in scaling cluster.
Completed updating DNS records.
DNS setup is completed
```

- Start the DNS server with the **service** command:

```
# service named restart
Stopping named: .[  OK  ]
Starting named: [  OK  ]
```

- Set the DNS server to start in run levels 3 and 5 at boot time with the following **chkconfig** command:

```
# chkconfig --level 345 named on
insserv: Service network is missed in the runlevels 4 to use service named
insserv: Service syslog is missed in the runlevels 4 to use service named
insserv: Service sshd is missed in the runlevels 4 to use service xcatd
```

For more information, see the xCAT YouTube channel on the Web at:

http://www.youtube.com/user/xcatuser

Part 5

Appendices

This section consists of the following appendices:

A

References and cheat sheets

This book refers to additional material that can be downloaded from the Internet as described below.

A.1 Related books

The following publications can be used as information sources:

- *Documentation for System z Linux Development stream* - on the Web at:

  http://www.ibm.com/developerworks/linux/linux390/documentation_red_hat.html

- *RHEL 6: IBM System z Architecture - Installation and Booting*:

  http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/pt-install-info-s390.html

- z/VM documentation - start at:

  http://www.vm.ibm.com/library/
  – z/VM Guide for Automated Installation and Service
  – z/VM CP Messages and Codes
  – z/VM TCP/IP Messages and Codes
  – The Program Directory for Performance Toolkit for VM
  – z/VM CP Commands and Utilities Reference
  – z/VM CP Planning and Administration
  – z/VM Getting Started with Linux on System z9 and zSeries
  – z/VM TCP/IP Planning and Customization
  – *z/VM Performance Toolkit Guide*, SC24-6156-00
  – *z/VM Performance Toolkit Reference*, SC24-6157-00

- *Redbooks - start at:*

  http://www.redbooks.ibm.com/
  – *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
  – *IBM Lotus Domino 6.5 for Linux on zSeries Implementation*, SG24-7021
  – *Printing with Linux on zSeries Using CUPS and Samba*, REDP-3864

A.2 Online resources

These Web sites and URLs are also relevant as further information sources:

- ► The Linux for zSeries and S/390 portal:

  http://linuxvm.org/

- ► The IBMVM list server:

  http://www.lsoft.com/scripts/wl.exe?SL1=IBMVM&H=LISTSERV.UARK.EDU

- ► 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

- ► SUSE LINUX Enterprise Server 9 evaluation:

  http://www.novell.com/products/linuxenterpriseserver/eval.html

- ► z/VM publications:

  http://www.vm.ibm.com/pubs/

- ► z/VM performance tips:

  http://www.vm.ibm.com/perf/tips/

A.3 Important z/VM 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 21-1provides a summary and the location of important z/VM configuration files.

Table 21-1 Important z/VM configuration files

File	Location	Description
SYSTEM CONFIG	PMAINT CF0	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 virtual machines known to the system are defined here (assuming a directory maintenance product is not being used).
PROFILE 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.

A.4 Cheat sheets

This section contains quick references or “cheat sheets” for the XEDIT and vi editors

A.4.1 XEDIT cheat 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 screen.

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
```

A.4.2 vi cheat sheet

Following is a small subset of **vi** commands, but those most commonly used.The vi editor has three modes:

1. Input mode - the **Insert** key, **i**, **o** (add a line below), **O** (add a line above) and other commands put 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
```

A.4.3 DirMaint cheat sheet

```
Add - Add a new user or profile directory entry
AMDisk - Adds a new minidisk
DEDicate - Add or delete an existing dedicate statements
DMDisk - Removes a minidisk
FILE - Add or replace a DirMaint control file
RLDCode - Cause DirMaint to reload its resident operating
procedures
RLDExtn - Cause DirMaint to reload its CONFIG* DATADVH file
REView - Review a user or profile directory entry
MDisk - Change the access mode and passwords for minidisks
STorage - Change logon storage size
SEND - Request a copy of a DirMaint control file
SETOptn - Add, change or delete CP options
CLAss - Change the CP class for a directory entry
SPEcial - Add or delete an existing special statement
TMDisk - Transfer ownership of a minidisk from one userid to
another
```

B

Source code

This section lists source code associated with this book. The following sections are included:

- Appendix B.1, “Obtaining and using the Web material” on page 377
- Appendix B.2, “z/VM REXX EXECs and XEDIT macros” on page 377
- Appendix B.4, “Linux code” on page 398

B.1 Obtaining and using the Web material

The PDF of this book is on the Internet at:

http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.pdf

The files associated with this book are in a *GNU zip* tar file at:

http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.tgz

To use the files associated with this book, see section 4.2, “Download files associated with this book”.

B.2 z/VM REXX EXECs and XEDIT macros

This section lists the z/VM REXX EXECs and the XEDIT macro described in this book. All the code in this section was converted from *willy-nilly* REXX to something more reputable by Doug Breneman of IBM Endicott.

B.2.1 The CHPW620 XEDIT macro

Following is the code for the XEDIT macro that changes all passwords in the z/VM 6.2 `USER DIRECT` file.In It recommended to reside on the `MAINT 191` disk.

```
/***************************************************************/
/*                                                             */
/*  This 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.                                                 */
/*                                                           */
/*************************************************************/
/*                                                           */
/* Purpose:                                                  */
/*   Change all passwords in z/VM 6.2 USER DIRECT file to    */
/*   the specified password.                                 */
/*                                                           */
/* Inputs:                                                   */
/*   newpass - the new password                              */
/*                                                           */
/* Output:                                                   */
/*   USER DIRECT file with passwords changed.                */
/*                                                           */
/*************************************************************/
  Address XEDIT
  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 */
  'SUPERSET /STAY ON/NUM ON/NULLS ON/SERIAL OFF'||,
  '/CMDLINE BOTTOM/CURLINE ON 3/SCALE OFF/CASE MIXED'||,
  '/PREFIX OFF/VERIFY 1 80/ARBCHAR ON $/ZONE 1 72'
  'TOP'

/* change z/VM 6.2 passwords */
  ident_users = 'MAINT AVSVM TSAFVM GCS AUDITOR AUTOLOG1 CMSBATCH',
  'DISKACNT EREP LGLOPR MONWRITE OP1 OPERATNS OPERATOR OPERSYMP',
  'SYSDUMP1 SYSMON VMRMADMN VMRMSVM VMSERVR VMSERVS VMSERVU',
  'AUTOLOG2 CBDIODSP DHCPD DTCENS1 DTCENS2 DTCVSW1 DTCVSW2 FTPSERVE',
  'GSKADMIN IMAP IMAPAUTH LDAPSRV LPSERVE MPROUTE OSADMIN1 OSADMIN2',
  'OSADMIN3 OSAMAINT OSASF PERFSVM PORTMAP RACFSMF RACFVM',
  'RACMAINT REXECD RSCS RSCSAUTH RSCSDNS SMTP SNMPD SNMPQE SNMPSUBA',
  'TCPIP TCPMAINT UFTD VMNFS VMUTIL VSMEVSRV VSMGUARD VSMPROXY',
  'VSMREQIM VSMREQIN VSMREQIU VSMREQI6 VSMWORK1 VSMWORK2 VSMWORK3',
  'XCHANGE ZVMLXAPP'
  Do i = 1 To 'WORDS'(ident_users)
     this_user = 'WORD'(ident_users,i)
     'CHANGE \IDENTITY' this_user ||' $' || this_user || '\IDENTITY' 'LEFT'(this_user,8)
'LEFT'(newpass,8) || '\ *'
     If rc <> 1 & rc <> 3 Then Do
```

```
        Say 'Error: CHANGE \IDENTITY' this_user ||' $' || this_user || '\IDENTITY'
'LEFT'(this_user,8) 'LEFT'(newpass,8) || '\ *'
        End
     End
  users = 'BLDCMS BLDNUC BLDRACF BLDSEG CMS1 IBMUSER LNXMAINT MAINT620',
  'PMAINT SYSADMIN SYSMAINT VMSERVP 4OSASF40 5684042J',
  '6VMDIR20 6VMHCD20 6VMPTK20 6VMRAC20 6VMRSC20 6VMTCP20'
  Do i = 1 To 'WORDS'(users)
     this_user = 'WORD'(users,i)
     'CHANGE \USER' this_user ||' $' || this_user || '\USER' 'LEFT'(this_user,8)
'LEFT'(newpass,8) || '\ *'
     If rc <> 1 & rc <> 3 Then Do
        Say 'Error: CHANGE \USER' this_user ||' $' || this_user || '\USER'
'LEFT'(this_user,8) 'LEFT'(newpass,8) || '\ *'
        End
     End
/* change mindisk passwords */
  mdisks = 'AUDITOR AUTOLOG AVSOBJ BATCH DHCPD DTCVSW1 DTCVSW2',
  'FTPSERV GCS GSKADMN DTCENS1 DTCENS2 IMAP LDAPSRV LPSERVE MAINT',
  'MPROUTE PORTMAP REXECD SMTP SNMPD SNMPQE SNMPSUB SYSMON TCPIP',
  'TCPMAIN TSAFOBJ UFTD VMNFS 4TCPIP'
  Do i = 1 To 'WORDS'(mdisks)
     this_mdisk = 'WORD'(mdisks,i)
     old = 'LEFT'('R' || this_mdisk,8),
     'LEFT'('W' || this_mdisk,8),
     'LEFT'('M' || this_mdisk,8)
     new = 'COPIES'('LEFT'(newpass,8),3)
     'CHANGE \' || old || '\' || new || '\ *'
     If rc <> 1 & rc <> 3 Then Do
        Say 'Error: CHANGE \USER' this_user ||' $' || this_user || '\USER'
'LEFT'(this_user,8) 'LEFT'(newpass,8) || '\ *'
        End
     End

  'CHANGE /ALL      WRITE    MULTIPLE/ALL' newPass newPass'/ *'
  If rc <> 1 & rc <> 3 Then Do
     Say 'Error: CHANGE /ALL      WRITE    MULTIPLE/ALL' newPass newPass'/ *'
     End
  'CHANGE /ALL      WTCPMAIN MTCPMAIN/ALL' newPass newPass'/ *'
  If rc <> 1 & rc <> 3 Then Do
     Say 'Error: CHANGE /ALL      WTCPMAIN MTCPMAIN/ALL' newPass newPass'/ *'
     End
  'CHANGE /RCATALOG WCATALOG/'newPass newPass'/ *'
  If rc <> 1 & rc <> 3 Then Do
     Say 'Error: CHANGE /RCATALOG WCATALOG/'newPass newPass'/ *'
     End
  'CHANGE /RCONTROL WCONTROL/'newPass newPass'/ *'
  If rc <> 1 & rc <> 3 Then Do
     Say 'Error: CHANGE /RCONTROL WCONTROL/'newPass newPass'/ *'
     End
  'CHANGE /RCRRLOG1 WCRRLOG1/'newPass newPass'/ *'
  If rc <> 1 & rc <> 3 Then Do
     Say 'Error: CHANGE /RCRRLOG1 WCRRLOG1/'newPass newPass'/ *'
     End
  'CHANGE /RCRRLOG2 WCRRLOG2/'newPass newPass'/ *'
  If rc <> 1 & rc <> 3 Then Do
     Say 'Error: CHANGE /RCRRLOG2 WCRRLOG2/'newPass newPass'/ *'
     End
  'CHANGE /RDATA    WDATA/'newPass newPass'/ *'
  If rc <> 1 & rc <> 3 Then Do
```

```
   Say 'Error: CHANGE /RDATA    WDATA/'newPass newPass'/ *'
   End
'CHANGE /RDVF     WDVF     MDVF/'newPass newPass newPass'/ *'
If rc <> 1 & rc <> 3 Then Do
   Say 'Error: CHANGE /RDVF     WDVF     MDVF/'newPass newPass newPass'/ *'
   End
'CHANGE /READ     WRITE    MULTIPLE/'newPass newPass newPass'/ *'
If rc <> 1 & rc <> 3 Then Do
   Say 'Error: CHANGE /READ     WRITE    MULTIPLE/'newPass newPass newPass'/ *'
   End
'CHANGE /READ     WRITE/'newPass newPass'/ *'
If rc <> 1 & rc <> 3 Then Do
   Say 'Error: CHANGE /READ     WRITE/'newPass newPass'/ *'
   End
'CHANGE /MR READ/'MR newPass'/ *'
If rc <> 1 & rc <> 3 Then Do
   Say 'Error: CHANGE /MR READ/'MR newPass'/ *'
   End
'CHANGE /RLOG1    WLOG1/'newPass newPass'/ *'
If rc <> 1 & rc <> 3 Then Do
   Say 'Error: CHANGE /RLOG1    WLOG1/'newPass newPass'/ *'
   End
'CHANGE /RLOG2    WLOG2/'newPass newPass'/ *'
If rc <> 1 & rc <> 3 Then Do
   Say 'Error: CHANGE /RLOG2    WLOG2/'newPass newPass'/ *'
   End
'CHANGE /RSERVER  WSERVER/'newPass newPass'/ *'
If rc <> 1 & rc <> 3 Then Do
   Say 'Error: CHANGE /RSERVER  WSERVER/'newPass newPass'/ *'
   End
Exit
```

B.2.2 The CPFORMAT EXEC

Following is the code for the EXEC that formats multiple disks using **CPFMTXA** (described in section 5.8, “Add page and perm volumes”):

```
/************************************************************/
/*                                                          */
/*  This 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.                                                */
/*                                                          */
/************************************************************/
/*                                                          */
/* Purpose:                                                 */
/*   CP format one, a range or multiple ranges of DASD.     */
/*   and label these DASDs.                                 */
```

```
/*                                                            */
/* Inputs:                                                    */
/*   dasds - address(es) of the DASD to format.               */
/*   type  - type of formatting to be done: PERM, PAGE, SPOL */
/*           or TEMP.                                         */
/*                                                            */
/* Output:                                                    */
/*   Virtual DASD that is CP formatted and labeled.           */
/*                                                            */
/* Return codes:                                              */
/*   0 - success                                              */
/*   1 - help was asked for or given                          */
/*   2 - user did not respond Y to confirm formatting         */
/*   3 - DASD (minidisk) range is not valid                   */
/*   4 - at least one DASD (minidisk) is reserved to MAINT    */
/*                                                            */
/* References:                                                */
/*  The Cloud Computing Cookbook for z/VM 6.2, RHEL 6.2 and  */
/*    SLES 11 SP2                                             */
/*  URL: http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.pdf */
/*                                                            */
/************************************************************/
  Address COMMAND
  firstchar = 'J'
  Arg dasds 'AS ' type .
  If dasds = '' | dasds = '?' Then Call help
  labelPrefix = firstchar || getLabelPrefix(type)
  numDasd = parseDasd(dasds)
  answer = areYouSure(type)
  If answer = 'Y' Then Do
     /* the user is sure */
     formatted = ''
     retVal = doFormat(labelPrefix numDasd type)
     Call doReport retVal
     End
  Else retVal = 2
  Exit retVal

/*+------------------------------------------------------------------+*/
help:
  Procedure Expose firstchar
/*+------------------------------------------------------------------+*/
  Parse Source . . fn .
  Say
  Say 'Synopsis:'
  Say
  Say '  Format and label DASD as page, perm, spool or temp disk space'
  Say '  The label written to each DASD is' firstchar || '<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 "                <---------------<              "
  Say "   >>--CPFORMAT--.-vdev--------.--AS---.-PERM-.---------><"
  Say "                 '-vdev1-vdev2-'       '-PAGE-'"
  Say "                                       '-SPOL-'"
  Say "                                       '-TEMP-'"
  Say
  Exit 1
```

```
/*+------------------------------------------------------------------+*/
areYouSure:
  Procedure
/*| Warn the user of possible data loss and ask if it is okay to     |*/
/*|  format the DASD.                                                |*/
/*|  parm 1: format type for the virtual DASD                        |*/
/*|  retVal: first character of response. continue if 'Y'.          |*/
/*+------------------------------------------------------------------+*/
  Arg type
  Say
  Say 'WARNING - this will destroy data!'
  Say 'Are you sure you want to format the DASD as' type 'space (y/n)?'
  Pull answer .
  Return  'LEFT'(answer,1) /* from areYouSure */

/*+------------------------------------------------------------------+*/
getLabelPrefix:
  Procedure expose firstchar
/*| Return the second character of the virtual DASD label            |*/
/*|  parm 1: format type for the virtual DASD                        |*/
/*+------------------------------------------------------------------+*/
  Arg type .
  firstchar. = 0
  firstchar.PERM = 'M'
  firstchar.PAGE = 'P'
  firstchar.SPOL = 'S'
  firstchar.TEMP = 'T'
  If firstchar.type = 0 Then Do
  /* Incorrect formatting type specified. Provide help and quit. */
     Say 'Error: "AS" must be present, type must be PERM, PAGE, SPOL or TEMP'
     Call help
     End
  Return firstchar.type

/*+------------------------------------------------------------------+*/
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
  dropheader = ''
  Say
  Say 'Format the following DASD:'
  Do While dasds <> ''
     Parse Upper Var dasds dasd dasds
     dashPos = 'POS'('-',dasd)
     If dashPos = 0 Then Do
     /* There is a singleton DASD specified. */
     /* start and end of range are the same. */
        startrange = dasd
        endrange = dasd
        End
     /* process the range of DASD */
     Else Parse Var dasd startrange  '-' endrange
     Do i = 'X2D'(startrange) To 'X2D'(endrange)
        numDasd = numDasd + 1
        dasdList.numDasd = 'D2X'(i)
```

```
        'PIPE CP QUERY MDISK' dasdList.numDasd 'LOCATION',
        dropheader,
        '|CONS'
        If rc <> 0 Then Do
           Say 'Return code from QUERY MDISK =' rc
           /* If RC=40, then HCPxxx40E has been issued and msg below */
           If rc = 40 Then Say 'DASD' dasdList.numDasd 'is not attached.'
           Exit 3
           End
        Call checkReserved(dasdList.numDasd)
        dropheader = '|DROP 1'
        End
     End
  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
  /* Save the current settings for MORE */
  Parse Value 'DIAG'('08','CP QUERY TERM') With ' MORE' morevalues ','
  'CP TERM MORE 1 1' /* Make MORE brief */

  /* Save system identifier and SSI name */
  'PIPE CP QUERY USERID | SPEC W3 | VAR systemID'
  'PIPE CP QUERY SSI | LOCATE /SSI Name/ | SPEC W3 | VAR SSIname'
  If (SSIname = "SSINAME") Then /* variable not set */
    inSSI = 'no'
  Else
    inSSI = 'yes'

  /* Iterate through all DASD in list */
  Do i = 1 to numDasd
     label = labelPrefix  || 'RIGHT'(dasdList.i,4,'0')
     retVal = formatOne(dasdList.i type label)
     If retVal <> 0 Then Do
        Say 'Error from CPFMTXA on DASD' label 'rc =' retVal
        Leave /* error - abort this format */
        End

     /* add owner info for CP owned devices */
     If (type != 'PERM') Then /* CP owned => owner info is needed */
       If (inSSI = 'yes') Then /* add owner info */
         call addOwnerInfo(dasdList.i label SSIname systemID)
       Else
         call addOwnerInfo(dasdList.i label "NOSSI" systemID)
     formatted = formatted label
     End /* Do i = */
  'CP TERM MORE' morevalues
  Return retVal /* from doFormat */

/*+------------------------------------------------------------------+*/
checkReserved:
  Procedure
```

```
/*| Try copying an already formatted DASD Then relabelling it        |*/
/*|  parm 1: dasd - the virtual address of the DASD                   |*/
/*+------------------------------------------------------------------+*/
  Arg dasd
  /* Create a list of reserved virtual DASD addresses. */
  /* Ensure that a system minidisk is not formatted.   */
  resvd = '122 123 124 190 191 193 19D 19E 2CC 401 402 990 CF1 CF3 CFD'
  If 'POS'(resvd,dasd) <> 0 Then Do
     /* MAINT minidisk - ABORT! */
     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
  'CP DETACH' dasds
  'CP ATTACH' dasds '*'
  Say
  Say 'DASD status after:'
  'CP QUERY 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 - PERM, PAGE, SPOL or TEMP                          |*/
/*|  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'
  'EXEC CPFMTXA'
  retVal = rc
  Return retVal /* from formatOne */

/*+------------------------------------------------------------------+*/
AddOwnerInfo:
  Procedure
/*| Tag PAGE, SPOL and TDSK volumes with SSI                         |*/
```

```
/*|  parm 1: disk - the vaddr to be formatted                          |*/
/*|  parm 2: type - PERM, PAGE, SPOL or TEMP                           |*/
/*|  parm 3: label - the six character label                           |*/
/*+--------------------------------------------------------------------+*/
  Arg disk label SSIname systemID
  Queue 'OWNER'
  Queue disk
  Queue label
  Queue SSIname
  Queue systemID
  'EXEC CPFMTXA'
  retVal = rc
  Return retVal /* from addOwnerInfo */
```

B.2.3 The SSICMD EXEC

Following is the code for the EXEC that issues CP commands on all joined members of an SSI cluster. It recommended to reside on the `MAINT 191` disk.

```
/************************************************************/
/*                                                          */
/*  This 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.                                                 */
/*                                                          */
/************************************************************/
/*                                                          */
/* Purpose:                                                 */
/*   Issue a command on all members of a cluster using the   */
/*   response from QUERY SSI to find the member names.       */
/*                                                          */
/* Inputs:                                                  */
/*   cmd - the CP command to issue on each member.           */
/*                                                          */
/* Output:                                                  */
/*   The results from issuing the AT command.                */
/*                                                          */
/* References:                                              */
/*  The Cloud Computing Cookbook for z/VM 6.2, RHEL 6.2 and  */
/*    SLES 11 SP2                                            */
/*  URL: http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.pdf */
/*                                                          */
/************************************************************/
  Address COMMAND
/* The command is passed by the caller */
  Arg cmd
  /* Provide help if requested or if no command is specified */
  If cmd = '' | cmd = '?' Then Call Help
```

```
  /* Determine the members of the SSI cluster */
  'PIPE CP QUERY SSI',
  '| STEM MSG.',           /* Save the response if error */
  '| XLATE',               /* Make all output upper case */
  '| FRTARGET ALL /SLOT/', /* Just look after 'SLOT' */
  '| LOCATE /JOINED/',     /* JOINED members can do a command */
  '| SPEC W2',             /* Get the member names */
  '| STEM SSI.'            /* Save the member names */
  /* If nonzero return code, show error message and exit */
  If rc <> 0 | ssi.0 = 0 Then Do
     Say 'Error: QUERY SSI return code =' rc
     Say msg.1
     End
  Else Do
  /* Send the command to each member of the SSI cluster */
     Do i = 1 To ssi.0
        Say ssi.i||":"
        'CP AT' ssi.i 'CMD' cmd
        Say
        End
     End
  Exit

help:
  /* Provide syntax information to the user */
  Say 'SSICMD cmd'
  Say
  Say 'cmd is a command to be issued on each of the members'
  Say '  in the SSI cluster using the AT command.'
  Exit
```

B.2.4 The SSISHUTD EXEC

Following is the code for the EXEC that shuts down or shuts and and re-IPLs an SSI cluster. It recommended to reside on the the `MAINT 191` disk.

```
/**************************************************************/
/*                                                            */
/*  This 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.                                                 */
/*                                                            */
/**************************************************************/
/*                                                            */
/* Purpose:                                                   */
/*   Issue a shutdown or shutdown reipl command to each      */
/*   member of an SSI cluster. It uses the response from     */
/*   QUERY SSI to find the member names.                     */
```

```
/*                                                            */
/* Inputs:                                                    */
/*   REIPL - option to specify that a REIPL should be done.  */
/*                                                            */
/* Output:                                                    */
/*   Possible return codes:                                   */
/*      0 - success                                           */
/*      1 - help was asked for or given                       */
/*      3 - user did not confirm the IPL                      */
/*      4 - one of the SHUTDOWN commands returned an error   */
/*      n - QUERY SSI or QUERY IPLPARMS returned an error    */
/*                                                            */
/* References:                                                */
/*  The Cloud Computing Cookbook for z/VM 6.2, RHEL 6.2 and  */
/*    SLES 11 SP2                                             */
/*  URL: http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.pdf */
/*                                                            */
/************************************************************/
  Address COMMAND
  Arg reipl .
  exitrc = 0
  If reipl = "?" Then Call help
  /* Determine the members of the SSI cluster */
  'PIPE CP QUERY SSI',
  '| STEM MSG.',           /* Save the response if error */
  '| XLATE',               /* Make all output upper case */
  '| FRTARGET ALL /SLOT/', /* Just look after 'SLOT' */
  '| LOCATE /JOINED/',     /* JOINED members can do a command */
  '| SPEC W2',             /* Get the member names */
  '| STEM SSI.'            /* Save the member names */
  /* If nonzero return code, show error message and exit */
  If rc <> 0 | ssi.0 = 0 Then Do
     Say 'Error: QUERY SSI return code =' rc
     Say msg.1
     exitrc = rc
     End
  Else Do
     reiplparms = ''
     If 'LENGTH'(reipl) <> 0 Then Do
        If reipl <> "REIPL" Then Do
           Say "Unrecognized parameter:" reipl
           Call help
           End
        Else Do
           'PIPE CP QUERY IPLPARMS | STEM MSG. | DROP 1 | VAR IPLP'
           If rc <> 0 Then Do
              Say 'QUERY IPLPARMS return code,' rc
              Say msg.1
              Exit rc
              End
           reiplparms = 'REIPL IPLPARMS' iplp 'CONS=SYSC'
           End
        End
     Call shutdown
     End
  Exit exitrc

/************************************************************/
shutdown:
  procedure expose ssi. reipl reiplparms
```

```
  Say "Are you sure you want to SHUTDOWN" reipl "the SSI cluster? (y/n)"
  Pull answer
  If 'LEFT'(answer, 1) <> "Y" Then Exit 3
  Do i = 1 to ssi.0
     'CP AT' ssi.i 'CMD SHUTDOWN SYSTEM' ssi.i reiplparms
     If rc <> 0 Then Exit 4
     End
  Return

/**********************************************************/
help:
  Say
  Say 'Synopsis:'
  Say '  SHUTDOWN or SHUTDOWN REIPL an SSI cluster'
  Say
  Say 'Syntax is:'
  Say "   >>--SSISHUTD-----.---------.-----------><"
  Say "                    '--REIPL--'"
  Say
  Exit 1
```

B.2.5 PROFILE EXEC for Linux virtual machines

This section lists the code for the `PROFILE EXEC` that is shared among Linux virtual machines 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
```

B.2.6 The RHEL62 EXEC

This section lists the code for the `RHEL62 EXEC` that starts a RHEL 6.2 installation. It recommended to reside on the `LNXMAINT 192` disk.

```
/**********************************************************/
/*                                                        */
/*  This 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.                                                 */
/*                                                           */
/*************************************************************/
/*                                                           */
/* Purpose:                                                  */
/*   Punch a RHEL6.2 install system to the user's reader     */
/*   and IPL it.                                             */
/*                                                           */
/* Inputs:                                                   */
/*   Files: RHEL62 KERNEL, userid PARM-RH6, RHEL62 INITRD    */
/*                                                           */
/* Output:                                                   */
/*   none                                                    */
/*                                                           */
/* WARNING:                                                  */
/*   This exec purges all reader files for this userid.      */
/*                                                           */
/*************************************************************/
  Address 'COMMAND'
  'CP SPOOL PUN *'
  'CP CLOSE RDR'
  'CP PURGE RDR ALL'
  'PUNCH RHEL62 KERNEL * (NOHEADER'
  'PUNCH' 'USERID'() 'PARM-RH6 * (NOHEADER'
  'PUNCH RHEL62 INITRD * (NOHEADER'
  'CP CHANGE RDR ALL KEEP'
  'CP IPL 00C CLEAR'
  Exit
```

B.2.7 The SLES11S2 EXEC

This section lists the code for the `SLES11S2 EXEC` that starts a SLES 11 SP2 installation. It recommended to reside on the `LNXMAINT 192` disk.

```
/* EXEC to punch SLES-11 SP2 install system to reader and IPL from it */
Address 'COMMAND'
'CP SPOOL PUN *'
'CP CLOSE RDR'
'CP PURGE RDR ALL'
'PUNCH SLES11S2   KERNEL * (NOHEADER'
'PUNCH' Userid() 'PARM-S11 * (NOHEADER'
'PUNCH SLES11S2   INITRD * (NOHEADER'
'CP CHANGE RDR ALL KEEP'
'CP IPL 00C CLEAR'
```

B.2.8 The SWAPGEN EXEC

Following is the code for the EXEC that creates Linux swap spaces from z/VM VDISKs:

```
/*******************************************************************
* Program: SWAPGEN EXEC
*
* Original Author: Dave Jones (djones@sinenomine.net)
```

```
*
* Description/Purpose:
*
* Generate VDISK swap for Linux on System z guest virtual
* machines
*
* Syntax:
*
* Issue: SWAPGEN ? for syntax etc.
*
* Version History:
*
* Versions 1-4 are globbed into the base code as the time the
* tool was placed under CMS update control. The descriptive
* text associated with these changes is preserved below. Version
* 5 begins normal CMS update maintenance.
*
* Begin globbed text included from old headers:
*
* 1 -- FBA only.  June 2002.
*
*           Initial release.
*
* 2 -- FBA/DIAG.  16 May 2003.  DIAG handling added.
*
* Swap file size bug for DIAG DASD (Adam Thornton, May 2003)
*
* 3 -- FBA/DIAG.  3 July 2003.  Error handling, comments, and license
*
* Rewritten by Phil Smith III, June 2003; added error handling etc.
* Various twiddlings by Adam Thornton and Dave Jones, July 2003
*                  added.
*
* 4 -- FBA/DIAG.  12 Mar 2004.  Put under source control, fixed
*                  label bug.
*
* 5 -- FBA/DIAG.  21 Jun 2004.  Add (REUSE option.
* More fixes by Adam Thornton, March 2004 and June 2004.
*
* End globbed text included from old headers:
*
* New updates are formatted as follows:
*
*   Update   Date      Maintainer
*
*           Version:  <versionid>
*
*           Descriptive text regarding update
*
* Update Log:
*
*   DB080001 2008-04-16 DB
*
*           Version: SNA080401
*
*           Clean up and structure module comments to standard.
*
*           Create private stack buffer for formatting and other
*           processing performed by SWAPGEN to avoid disturbing
*           the contents of the console stack
```

```
*           Reported by: Scott Rohling (scott.rohling@gmail.com)
*
*           Add version display message if called with VERSION
*           option. Future updates should ensure that the Version:
*           routine is updated to reflect current release info.
*
*           Add debugging messages if called with DEBUG
*           option.
*
*********************************************************************
*********************************************************************
* License:
*
* Distribution of this package is governed by the Artistic License
* as follows:

Preamble

The intent of this document is to state the conditions under which a
Package may be copied, such that the Copyright Holder maintains some
semblance of artistic control over the development of the package, while
giving the users of the package the right to use and distribute the
Package in a more-or-less customary fashion, plus the right to make
reasonable modifications.

Definitions

    "Package" refers to the collection of files distributed by the
    Copyright Holder, and derivatives of that collection of files
    created through textual modification.

    "Standard Version" refers to such a Package if it has not been
    modified, or has been modified in accordance with the wishes of the
    Copyright Holder as specified below.

    "Copyright Holder" is whoever is named in the copyright or
    copyrights for the package.

    "You" is you, if you're thinking about copying or distributing this
    Package.

    "Reasonable copying fee" is whatever you can justify on the basis of
    media cost, duplication charges, time of people involved, and so
    on. (You will not be required to justify it to the Copyright Holder,
    but only to the computing community at large as a market that must
    bear the fee.)

    "Freely Available" means that no fee is charged for the item itself,
    though there may be fees involved in handling the item. It also
    means that recipients of the item may redistribute it under the same
    conditions they received it.

   1. You may make and give away verbatim copies of the source form of
      the Standard Version of this Package without restriction, provided
      that you duplicate all of the original copyright notices and
      associated disclaimers.

   2. You may apply bug fixes, portability fixes and other modifications
      derived from the Public Domain or from the Copyright Holder. A
      Package modified in such a way shall still be considered the
```

```
   Standard Version.

3. You may otherwise modify your copy of this Package in any way,
   provided that you insert a prominent notice in each changed file
   stating how and when you changed that file, and provided that you
   do at least ONE of the following:

      1. place your modifications in the Public Domain or otherwise
      make them Freely Available, such as by posting said
      modifications to Usenet or an equivalent medium, or placing the
      modifications on a major archive site such as uunet.uu.net, or
      by allowing the Copyright Holder to include your modifications
      in the Standard Version of the Package.
      2. use the modified Package only within your corporation or
      organization.
      3. rename any non-standard executables so the names do not
      conflict with standard executables, which must also be
      provided, and provide a separate manual page for each
      non-standard executable that clearly documents how it differs
      from the Standard Version.
      4. make other distribution arrangements with the Copyright
      Holder.

4. You may distribute the programs of this Package in object code or
   executable form, provided that you do at least ONE of the
   following:

      1. distribute a Standard Version of the executables and library
      files, together with instructions (in the manual page or
      equivalent) on where to get the Standard Version.
      2. accompany the distribution with the machine-readable source
      of the Package with your modifications.
      3. give non-standard executables non-standard names, and
      clearly document the differences in manual pages (or
      equivalent), together with instructions on where to get the
      Standard Version.
      4. make other distribution arrangements with the Copyright
      Holder.

5. You may charge a reasonable copying fee for any distribution of
   this Package. You may charge any fee you choose for support of
   this Package. You may not charge a fee for this Package
   itself. However, you may distribute this Package in aggregate with
   other (possibly commercial) programs as part of a larger (possibly
   commercial) software distribution provided that you do not
   advertise this Package as a product of your own. You may embed
   this Package's interpreter within an executable of yours (by
   linking); this shall be construed as a mere form of aggregation,
   provided that the complete Standard Version of the interpreter is
   so embedded.

6. The scripts and library files supplied as input to or produced as
   output from the programs of this Package do not automatically fall
   under the copyright of this Package, but belong to whomever
   generated them, and may be sold commercially, and may be
   aggregated with this Package. If such scripts or library files are
   aggregated with this Package via the so-called "undump" or
   "unexec" methods of producing a binary executable image, then
   distribution of such an image shall neither be construed as a
   distribution of this Package nor shall it fall under the
```

```
      restrictions of Paragraphs 3 and 4, provided that you do not
      represent such an executable image as a Standard Version of this
      Package.

   7. C subroutines (or comparably compiled subroutines in other
      languages) supplied by you and linked into this Package in order
      to emulate subroutines and variables of the language defined by
      this Package shall not be considered part of this Package, but are
      the equivalent of input as in Paragraph 6, provided these
      subroutines do not change the language in any way that would cause
      it to fail the regression tests for the language.

   8. Aggregation of this Package with a commercial distribution is
      always permitted provided that the use of this Package is
      embedded; that is, when no overt attempt is made to make this
      Package's interfaces visible to the end user of the commercial
      distribution. Such use shall not be construed as a distribution of
      this Package.

   9. The name of the Copyright Holder may not be used to endorse or
      promote products derived from this software without specific prior
      written permission.

  10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR
      IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED
      WARRANTIES OF MERCHANTIBILITY AND FITNESS FOR A PARTICULAR
      PURPOSE.

The End

*/

   address command

   arg vdev blks . '(' options ')'

   debug = 0                                    /* Default to quiet */
   fba = 0                                     /* No FBA option yet */
   reuse = 0                                 /* No reuse option yet */
   do while options <> ''                      /* Parse the options */
      parse var options option options              /* Get an option */
      select
         when option = 'DIAG' then fba = 0        /* Use DIAG driver */
         when option = 'FBA' then fba = 1          /* Use FBA driver */
         when option = 'REUSE' then reuse = 1          /* Reuse DASD */
         when option = 'DEBUG' then debug = 1    /* Wants debug chat */
         when option = 'VERSION' then signal Version /* version query*/
         otherwise
         say 'Invalid option "'option'"'             /* Else unknown */
      end
   end

   minblks = 32 - 8 * fba  /* Minimum number of blocks that can work */

   if reuse = 1 then do
      parse value diagrc(8, 'Q V 'vdev) ,     /* Get blocks from ... */
        with rc . 17 msg                   /* ... actual device size */
      if rc <> 0 then signal BadDev
      parse var msg . . . . . newblks .
      if blks <> newblks then signal WrongBlks    /* Mismatch, error */
```

```
end

if vdev = '?' then signal Help            /* Wants Help, give it */
if vdev = '' then signal NoVdev               /* Missing, error */
if blks = '' then signal NoBlks               /* Missing, error */
if datatype(blks, 'W') = 0 | blks < minblks then /* Bad/too small */
signal BadBlks                                      /* So error */
if datatype(vdev, 'X') = 0 | length(vdev) > 4 then     /* Invalid */
signal BadVdev                                      /* So error */

if fba then do    /* If FBA driver, make sure we have the package */
   'NUCEXT RXDASD'                           /* Already got it?? */
   if rc <> 0 then 'NUCXLOAD RXDASD'        /* No, try to load it */
   if debug then say 'SWAPGEN: Loading RXDASD got rc=' rc
   if rc <> 0 then signal NoRXDASD       /* That failed, so error */
end

if reuse = 0 then do
   call diag 8, 'DETACH' vdev       /* DETACH any existing device */
   parse value diagrc(8, 'DEFINE VFB-512 AS' vdev 'BLK' blks) ,
     with rc . 17 msg '15'x                  /* Define the V-DISK */
   if debug then say 'SWAPGEN: DEFINE VDEV got rc=' rc
   if rc <> 0 then signal BadDefine      /* That failed, so error */
end
call csl 'DMSGETFM rc reacode  fm'        /* find a free filemode */
if debug then say 'SWAPGEN: Got filemode' fm 'from DMSGETFM'
if rc <> 0 then signal NoFreeModes /* Weren't any, strange, error */

if fba then do
   pages = trunc((blks * 512)/4096) - 1               /* FBA case */
   writeit = 'stem swap.'                            /* Pipe stage */
   if debug then say 'SWAPGEN: Computed' pages 'for FBA disk'
end
else do                 /* Not FBA, we must FORMAT and RESERVE it */
   'MAKEBUF'         /* Guard stack contents if something's there */
   buf = rc    /* Remember buffer number so we drop the right one */
   if debug then say 'SWAPGEN: Acquired buffer' buf ,
     'before non-FBA format.'
   writeit = 'specs number 1 1-* next' ,
     '| mdskupdate LINUX SWAP' fm 'F 512'         /* Pipes stages */
   queue '1'                       /* Yes to the format? question */
   queue 'LXSWAP'                              /* Disk volume name */
   queue '1'                       /* Yes to the reserve question */
   'PIPE (name SWPFORMAT)' ,       /* FORMAT and RESERVE the disk */
     '| cms FORMAT' vdev fm '(BLKSIZE 512 NOERASE' ,    /* FORMAT */
     '| var rs1' ,                      /* Remember how that went */
     '| hole' ,                         /* And otherwise pitch it */
     '| cms RESERVE LINUX SWAP' fm ,                     /* Do it */
     '| var rs2' ,                      /* Remember how that went */
     '| hole' ,                         /* And otherwise pitch it */
     '| state LINUX SWAP' fm  , /* Look at the reserved swap file */
     '| var reserveok' ,                 /* Keep that information */
     '| specs w6 1'          ,  /* Word 6 is the number of blocks */
     '| specs w1 1' ,                    /* Calculate it modulo 8 */
     'a: word 1 .' ,                     /* Get the token we want */
     '   set #0:=a%8-1' ,        /* Calculate it modulo 8 minus 1 */
     '   print #0 20' ,                               /* Write it */
     '| specs w2 1' ,      /* Just get the number of usable pages */
     '| var pages'                         /* And remember that */
   if debug then say 'SWAPGEN: Formatted' pages 'pages on disk' ,
```

```
     fm 'in PIPE'
    'DROPBUF' buf          /* Not nice to leave trash lying around */
    if debug then say 'SWAPGEN: Dropped buffer' buf
 end

 if debug then say 'SWAPGEN: About to write non-FBA swap signature'

/* Must use separate Pipe to write since mdskupdate commits to 0 */
 'PIPE (name SWPWrite)' ,
   '| var pages' ,                          /* Get number of pages */
   '| specs pad 00 w1 d2c 1.4 right' ,                /* Format it */
   '| append strliteral x'c2x(copies('00'x, 4086-1033+1) || ,
   '53574150535041434532'x) ,             /* "SWAPSPACE2" in ASCII */
   '| join' ,                        /* Build that into a nice chunk */
   '| preface strliteral x'c2x(copies('00'x, 1027)'01'x) ,   /* 0s */
   '| join' ,                        /* Build that into a nice chunk */
   '| deblock 512',                          /* Break into records */
   '|' writeit   /* And write to disk or variable, per driver type */
 if debug then say 'SWAPGEN: Wrote non-FBA swap signature with rc=' rc
 if rc <> 0 then signal BadWrite

/* If FBA, we have the values, need to use RXDASD to write them */
 if fba then do i = 1 to swap.0     /* If FBA, we didn't write yet */
    if debug then say 'SWAPGEN: About to write FBA signature' i
    rc = DASD('WRITED', vdev, i-1, swap.i)           /* Write one */
    if debug then say 'SWAPGEN: Wrote FBA signature' i 'with rc='rc
    if rc <> 0 then signal BadWrite            /* Failed, so error */
 end

 if fba then type ='FBA'
 else type = 'DIAG'
 say type 'swap disk defined at virtual address' vdev , /* Success! */
   '('pages-1' 4K pages of swap space)'
 call Quit 0

Quit:
 arg rc
 if rc <> 0 then say 'No Swap disk was created.'
 exit rc

NoVdev:              /* User didn't give us a virtual device address */
 say 'A virtual device address must be specified!'
 signal Help

NoBlks:                    /* User didn't give us a number of blocks */
 say 'Number of blocks must be specified!'
 signal Help

NoFreeModes:                      /* No free disk modes can be found */
 say 'No free disk modes are available!'
 say 'Please release a minidisk and try again.'
 call Quit 1      /* They invoked it correctly, so don't show help */

BadDev:                             /* REUSE tried to use bad device */
 say 'The device at 'vdev' cannot be used:'
 say msg
 call Quit 24

BadBlks:                 /* User gave us an invalid number of blocks */
 say 'Invalid number of blocks "'blks'" specified; must be'
```

```
   say 'at least 'minblks' 512-byte blocks.'
   call Quit 24

WrongBlks:               /* Supplied number of blocks does not match */
   say 'REUSE requested with' blks ,
     'and existing disk block count is' newblks'.'
   call Quit 24

BadVdev:           /* User gave us an invalid virtual device address */
   say 'Invalid virtual device address "'vdev'" specified;'
   say 'must be a 1- to 4-digit hexadecimal value.'
   call Quit 24

NoRXDASD:                  /* We don't have the required FBA utility */
   say 'Unable to NUXCLOAD RXDASD MODULE; this is available from:'
   say ' http://www.vm.ibm.com/download/packages'
   call Quit rc

BadDefine:                              /* Error DEFINE-ing the VDISK */
   say 'Error' rc 'from CP DEFINE VFB-512 AS' vdev 'BLK' blks':'
   say msg                                   /* Display error from CP */
   call Quit rc

BadFBA:                            /* Error writing FBA block on disk */
   say 'Error' rc 'from RXDASD'
   call Quit rc

BadWrite:                        /* Error on FORMAT or RESERVE steps */
   select                          /* Figure out where it went wrong */
      when symbol('RESERVEOK') <> 'VAR' then do
         say 'Error' rc 'from CMS RESERVE LINUX SWAP' fm':'
         say rs2
      end
      when symbol('RS2') <> 'VAR' then do
         say 'Error' rc 'from CMS FORMAT' vaddr fm '(BLKSIZE 512:'
         say rs1
      end
      otherwise
      say 'Error' rc 'calculating swap size, contact support'
   end
   call Quit rc

Help:
   parse source . . fn .

   say 'Syntax is:'
   say ''
   say fn 'vdev #blocks <( <options> <)> >'
   say ''
   say 'where:'
   say ''
   say 'vdev     -- is a virtual device address'
   say '#blocks  -- is a decimal number of 512-byte blocks;'
   say '            minimum 24 (FBA) or 32 (DIAG)'
   say ''
   say 'Options are:'
   say 'DIAG     -- (Default) Use DIAG I/O (requires Linux DIAG driver)'
   say 'FBA      -- use FBA driver instead of DIAG; requires RXDASD'
   say '            package, downloadable from the IBM VM download'
   say '            page at: http://www.vm.ibm.com/download/packages'
```

```
   say 'REUSE    -- use existing device at vdev.  WARNING: This will'
   say '            destroy any data on device vdev.'
   say 'VERSION  -- display current version number string and date'
   say '            of last module update.'
   say 'DEBUG    -- display progress messages and debugging'
   say '            information about the program logic. '
   say ''
   say fn 'will DETACH any existing virtual device at that address,'
   say 'DEFINE a new VDISK, format it, and write the Linux swap'
   say 'signature on the disk so Linux will recognize it.'
   say ''
   say 'If using FBA mode, SWAPGEN prepares the whole device:'
   say '    /dev/dasdb or /dev/dasd/0151/device'
   say 'so the whole device must then specified in the Linux fstab.'
   say ''
   say 'If using DIAG mode, because the V-DISK is CMS FORMATted,'
   say 'SWAPGEN prepares the partition:'
   say '    /dev/dasdb1 or /dev/dasd/0151/part1'
   say 'so the partition must be specified in the fstab on Linux.'
   call Quit 1

Version:
   parse source . . fn .

/* These variables should be updated with each release */
   version = 'SNA080403'                /* Release string: SNAyymmvv */
   last_update = '2008-04-18 (yyyy-mm-dd)'       /* Last update date */

   say 'SWAPGEN: Version' version', last updated:' last_update'.'
   call Quit 4

/* UPDATES APPLIED AT 09:38:57 ON 18 APR 2008
AT124501  Initialize
*         SWAPGEN  AT124501 Z1 -       06/21/04   13:57:22
AT124502  Add more error checking and (REUSE option
*         SWAPGEN  AT124502 Z1 -       06/21/04   14:40:30
DB080001  ADD MAKEBUF/DROPBUF PROCESSING TO TOLERATE STUFF IN STACK
*         SWAPGEN  DB080001 Z1 -       04/16/08   11:43:21
PS080002  Use NOERASE to speed FORMAT as appropriate
*         SWAPGEN  PS080002 Z2 -       04/17/08   06:44:54
PS080003  Improve error handling to make things clearer to users
*         SWAPGEN  PS080003 Z2 -       04/18/08   09:38:48
*/
```

B.3 Sample files

This section lists sample files described in the book.

B.3.1 The SAMPLE CONF-RH6 file

This section lists the sample RHEL 6 configuration file:

```
DASD=100-103,300-301
HOSTNAME=hostName.DNSname.com
NETTYPE=qeth
IPADDR=n.n.n.n
SUBCHANNELS=0.0.0700,0.0.0701,0.0.0702
```

```
NETMASK=255.255.255.0
SEARCHDNS=DNSname.com
GATEWAY=n.n.n.n
DNS=n.n.n.n
MTU=1500
PORTNAME=DONTCARE
PORTNO=0
LAYER2=1
```

B.3.2 The SAMPLE PARM-RH6 file

This section lists the sample RHEL 6 configuration file:

```
root=/dev/ram0 ro ip=off ramdisk_size=40000
CMSDASD=191 CMSCONFFILE=userid.CONF-RH6
vnc vncpassword=12345678
```

B.3.3 The SAMPLE PARM-S11 file

This section lists the sample SLES 11 SP2 configuration file:

```
ramdisk_size=65536 root=/dev/ram1 ro init=/linuxrc TERM=dumb
HostIP=n.n.n.n Hostname=yourhost.example..com
Gateway=n.n.n.n Netmask=255.255.255.0
Broadcast=n.n.n.n Layer2=1
ReadChannel=0.0.0700  WriteChannel=0.0.0701  DataChannel=0.0.0702
Nameserver=n.n.n.n
portname=whatever
portno=0
Install=nfs://n.n.n.n/var/nfs/sles11sp2/SLES-11-SP2-DVD-s390x-GM-DVD1.iso
UseVNC=1  VNCPassword=12345678
InstNetDev=osa OsaInterface=qdio OsaMedium=eth Manual=0
```

B.4 Linux code

Following are two sections listing the clone scripts for RHEL and SLES.

B.4.1 RHEL clone script

This section lists the code for the **/usr/sbin/clone** script that clones from a RHEL golden Linux image to a target virtual machine.

```
#!/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] sourceID targetID [rootMinidisk [minidisk1 minidisk2..]]"
  echo "    Switches"
  echo "       -v Verbose output"
  echo "    Required"
  echo "       sourceID the z/VM user id you want to clone from"
  echo "       targetID  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/${target_linux_id}.conf ] && . /etc/clone/${target_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 $target_linux_id's value for: " && first=1
    echo -n "$v: "
    read in
    eval $(echo $v=\"$in\")
    export $v
    echo "$v=$in" >> /etc/clone/${target_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/$target_dev_node
  ret_val=$?

  if [ $ret_val -eq 0 ] ; then
    [ -n "$VERBOSE" ] && echo "Invoking Linux command: dasdfmt -p -b 4096 -y -F -f
/dev/$target_dev_node"
    [ -n "$VERBOSE" ] && progress="-p"
    dasdfmt $progress -b 4096 -y -F -f /dev/$target_dev_node
    [ $? -ne 0 ] && echo "Error: dasdfmt failed" && ret_val=1
  fi

  if [ $ret_val -eq 0 ] ; then
    wait_for_device /dev/$source_dev_node
    ret_val=$?
  fi
```

```
  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 [ "$CLONE_METHOD" == "AUTO" -o "$CLONE_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 $source_linux_id $1 $MASTER_LINK RR
    link_one $target_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 $target_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 $source_linux_id to $target_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/${target_linux_id}.conf -a "$PROMPT" != "y" ]; then
   echo "Error: /etc/clone/${target_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

  sleep 2
  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 $target_linux_id
  rc=$?
  if [ $? != 0 ]; then
    echo "xautolog $target_linux_id failed with $rc"
    return 0
  fi
  echo "Booting $target_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

# Allow UPPER or lower case source, target, blacklist entries.
# Convert all to lower case for consistency.
source_linux_id=$(echo $1 | tr "[:upper:]" "[:lower:]")
target_linux_id=$(echo $2 | tr "[:upper:]" "[:lower:]")

# Check the blacklist, which prevents using the master image as a target.
if [ -f /etc/clone/blacklist.conf ]; then
  . /etc/clone/blacklist.conf
  BlackList=$(echo ${BLACKLIST} | tr "[:upper:]" "[:lower:]")
  for Target in ${BlackList}
  do
    if [ "${Target}" == "${target_linux_id}" ]; then
```

```
      echo "${target_linux_id} is blacklisted!  Exiting!"
      exit
    fi
  done
fi

# Check that the master and clone z/VM IDs are logged off.
check_logged_off $source_linux_id
check_logged_off $target_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 $source_linux_id to $target_linux_id ..."
[ -z "$DASD" ] && echo "Error: no DASD defined in /etc/clone/${target_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 $target_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 $source_linux_id to $target_linux_id"
```

B.4.2 SLES clone script

This section lists the code for the **/usr/local/sbin/clone.sh** script that clones from a SLES golden Linux image to a target virtual machine.

```
#!/bin/sh
#
# clone.sh <LinuxUserID> - clone a Linux server running under z/VM
#
# For details on how this script works see the book:
#  "z/VM and Linux on IBM System z: The Cloud Computing Cookbook
#    for z/VM 6.2 RHEL 6.2 and SLES 11 SP2"
#  on the Web at: http://www.vm.ibm.com/devpages/mikemac/CKB-VM62.pdf
#
# ---------------------------------------------------------------------------
# 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
# ---------------------------------------------------------------------------

#+--------------------------------------------------------------------------+
function help()
# give help
#+--------------------------------------------------------------------------+
 {
  echo "Usage: clone [options] from <sourceID> to <targetID>"
  echo ""
  echo "  Clone Linux from sourceID 100 and 101 minidisks to targetID"
  echo "  options:"
  echo "    -v or --verbose: verbose"
  echo ""
  echo "Example: clone.sh from s11gold to linux01"
  exit 1
 }

#+--------------------------------------------------------------------------+
function processArguments()
# Parse command line arguments
# Args: The arguments passed in to the script
#+--------------------------------------------------------------------------+
 {
  verbose="off"
  sourceID="none"
  targetID="none"
  while (( "$#" )); do
    case $1 in
      -v|--verbose)
         verbose="on"
         ;;
      from)
        shift
        sourceID=`echo $1 | tr '[a-z]' '[A-Z]'` # fold source ID to upper case
```

```
        ;;
      to)
        shift
        targetID=`echo $1 | tr '[a-z]' '[A-Z]'` # fold target ID to upper case
        ;;
    esac
    shift
  done
  if [ $sourceID = "none" ]; then # source user ID was not passed
    echo "Error: Source Linux user ID not supplied"
    help
  fi
  if [ $targetID = "none" ]; then # target user ID was not passed
    echo "Error: Target Linux user ID not supplied"
    help
  fi
 }

#+--------------------------------------------------------------------------+
function CPcmd()
# echo a CP command and invoke it via the vmcp module/command
#   Arg1-n: the command to issue
#   Return: the command's return code
#+--------------------------------------------------------------------------+
 {
  echo "Invoking CP command: $@"
# parse output to get return code: awk -F# splits line at '#' with rc at end
  output=`vmcp $@ 2>&1`
  echo "$output"
  retVal=0
  retVal=`echo $output | grep "Error: non-zero CP response" | awk -F# '{print $2}'`
  return $retVal
 }

#+--------------------------------------------------------------------------+
function checkID()
# Verify user ID exists and is logged off
#    Arg 1: The user ID to check
#+--------------------------------------------------------------------------+
 {
  userID=$1
  echo "Checking that $userID exists and is not logged on ..."
  CPcmd QUERY $userID
  rc=$?
  case $rc in
    0)  # user ID is logged on or disconnected
      echo "$userID user ID must be logged off"
      exit 2
      ;;
    3)  # user ID does not exist
      echo "$userID user ID does not exist"
      exit 3
      ;;
   45) # user ID is logged off - this is correct
      ;;
   *) # unexpected
      echo "Return code of $rc unexpected from QUERY $userID"
      echo "User ID must exist and be logged off"
      exit 4
  esac
```

```
 }

#+--------------------------------------------------------------------------+
function prepareIPaddr()
# Set the variable "newIPaddr" by adding a backslash before any "."s
#    Arg 1: The IP address to be modified
#+--------------------------------------------------------------------------+
 {
  newIPaddr=`echo $1 | sed -e 's:\.:\\\.:g'`
 }

#+--------------------------------------------------------------------------+
function prepareVaddr()
# Prepare an address by folding to lower case and prepending leading zeros
# to make it 4 digits
#    Arg 1: The vaddr to be modified
# Return:
#    The new value is written to the global variable newVaddr
#+--------------------------------------------------------------------------+
 {
  newVaddr=`echo $1 | tr '[A-Z]' '[a-z]'`  # fold to lower case
  let leadingZeros=4-${#1}                 # determine number of zeros to add
  let i=0
  while [ $i -lt $leadingZeros ]; do
    newVaddr="0$newVaddr"
    i=$[$i+1]
  done
 }

#+--------------------------------------------------------------------------+
function copyDisk()
# Use FLASHCOPY to copy a disk, if it fails, fall back to dasdfmt then dd
#   Arg 1: Source vaddr
#   Arg 2: Target vaddr
#+--------------------------------------------------------------------------+
 {
  source=$1
  target=$2
  echo ""
  echo "FLASHCOPYing $source to $target ..."
  CPcmd FLASHCOPY $source 0 end to $target 0 end
  if [ $? != 0 ]; then
    echo "FLASHCOPY failed, falling back to dasdfmt and dd ..."
    chccwdev -e $source
    if [ $? != 0 ]; then exit 7; fi
    chccwdev -e $target
    if [ $? != 0 ]; then exit 8; fi
    sleep 1
    srcDev=/dev/$(egrep ^0.0.$source /proc/dasd/devices | awk '{ print $7 }')
    if [ "$?" != 0 ]; then exit 5; fi
    tgtDev=/dev/$(egrep ^0.0.$target /proc/dasd/devices | awk '{ print $7 }')
    if [ "$?" != 0 ]; then exit 6; fi
    echo "dasdfmt-ing $tgtDev ..."
    dasdfmt -y -b 4096 -f $tgtDev
    if [ "$?" != 0 ]; then exit 9; fi
    echo "dd-ing $srcDev to $tgtDev ..."
    dd bs=1M if=$srcDev of=$tgtDev oflag=sync
    if [ "$?" != 0 ]; then exit 10; fi
    sync
    echo "disabling and re-enabling $target ..."
```

```
    chccwdev -d $target
    if [ $? != 0 ]; then exit 11; fi
    chccwdev -e $target
    if [ $? != 0 ]; then exit 12; fi
    sync
  fi
 }

#+--------------------------------------------------------------------------+
function getNetworkInfo()
# Bring 191 minidisk online to check for two parameter files
#+--------------------------------------------------------------------------+
 {
  # recycle 191 to pick up latest changes
  chccwdev -d 191
  chccwdev -e 191
  rc=$?
  if [ $rc != 0 ]; then # unable to enable 191 disk
    echo "unable to enable 191, rc from chccwdev = $rc"
    exit 13
  fi
  udevadm settle
  CMSdisk=`lsdasd | grep 0191 | awk '{ print $3 }'`
  cmsfslst -d /dev/$CMSdisk | grep -i $sourceID | grep PARM-S11
  rc=$?
  if [ $rc != 0 ]; then
    echo "Error: $sourceID PARM-S11 not found on 191 minidisk. Exiting"
    exit 14
  fi
  cmsfslst -d /dev/$CMSdisk | grep -i $targetID | grep PARM-S11
  rc=$?
  if [ $rc != 0 ]; then
    echo "Error: $targetID PARM-S11 not found on 191 minidisk. Exiting"
    exit 15
  fi

# get informaton about target
  export local $(cmsfscat -a -d /dev/$CMSdisk $targetID.PARM-S11)
  tagetHostname=$Hostname
  targetIP=$HostIP
  targetDNS=$Nameserver
  targetGW=$Gateway
  targetMask=$Netmask
  targetBroadcast=$Broadcast
  prepareVaddr $ReadChannel
  targetReaddev=$newVaddr
  prepareVaddr $WriteChannel
  targetWritedev=$newVaddr
  prepareVaddr $DataChannel
  targetDatadev=$newVaddr

# get information about source
  export local $(cmsfscat -a -d /dev/$CMSdisk $sourceID.PARM-S11)
  sourceHostname=$Hostname
  prepareIPaddr $HostIP
  sourceIP=$newIPaddr
  prepareIPaddr $Nameserver
  sourceDNS=$newIPaddr
  prepareIPaddr $Gateway
  sourceGW=$newIPaddr
```

```
  prepareIPaddr $Netmask
  sourceMask=$newIPaddr
  prepareIPaddr $Broadcast
  sourceBroadcast=$newIPaddr
  prepareVaddr $ReadChannel
  sourceReaddev=$newVaddr
  prepareVaddr $WriteChannel
  sourceWritedev=$newVaddr
  prepareVaddr $DataChannel
  sourceDatadev=$newVaddr
 }

#+--------------------------------------------------------------------------+
function askAreYouSure()
# Ask "Are you sure?" - if not, then exit
#+--------------------------------------------------------------------------+
 {
  echo ""
  echo "WARNING!!: Mindisks 100 and 101 will be copied to $targetID"
  echo "New host name will be:  $tagetHostname"
  echo "New IP address will be: $targetIP"
  echo "Other network data is retrieved from $targetID PARM-S11 on 191 disk"
  echo -n "Are you sure you want to overwrite these disks (y/n): "
  read ans
  if [ $ans != "y" ]; then
    echo "Aborting clone per user input"
    exit 16
  fi
 }

#+--------------------------------------------------------------------------+
function copySystem()
# For each of two mindisks 100 and 101:
#   -) Link disk
#   -) Enable disk
#   -) Copy disk
#+--------------------------------------------------------------------------+
 {
  echo "Linking source and target 100 disks ..."
  CPcmd detach 1100
  CPcmd link $sourceID 100 1100 rr
  if [ $? != 0 ]; then exit 17; fi
  CPcmd detach 2100
  CPcmd link $targetID 100 2100 mr
  if [ $? != 0 ]; then exit 18; fi
  echo "Copying 100 disks ..."
  copyDisk 1100 2100
  echo "Take 1100 Offline...."
  chccwdev -d 1100
  CPcmd det 1100
  # don't detach 2100 yet because it has to be modified

  echo " "
  echo "--------------------------------------"
  echo "Linking source and target 101 disks ..."
  CPcmd detach 1101
  CPcmd link $sourceID 101 1101 rr
  if [ $? != 0 ]; then exit 19; fi
  CPcmd detach 2101
  CPcmd link $targetID 101 2101 mr
```

```
  if [ $? != 0 ]; then exit 20; fi
  echo "Copying 101 disks ..."
  copyDisk 1101 2101
  echo "Taking 1101 Offline..."
  chccwdev -d 1101
  CPcmd det 1101
  echo "Taking 2101 Offline..."
  chccwdev -d 2101
  CPcmd det 2101
 }

#+--------------------------------------------------------------------------+
function modifyClone()
# Mount newly copied system and modify networking info
#+--------------------------------------------------------------------------+
 {
  targetVaddr=2100
  targetDir="/mnt/targetLinux"         # directory of target Linux system
  echo "Mounting newly cloned image over $targetDir ..."
  if [ ! -d $targetDir ]; then
    mkdir $targetDir
    rc=$?
    if [ $rc != 0 ]; then
      echo "Error: mkdir $targetDir failed with $rc"
      exit 21
    fi
  fi
  # recycle target mount point
#  chccwdev -d $targetVaddr
  chccwdev -e $targetVaddr
  rc=$?
  if [ $rc != 0 ]; then
    echo "Fatal error: chccwdev -e $targetVaddr failed with $rc"
    CPcmd DET $targetVaddr
    exit 22
  fi
  sleep 2
  tDev=/dev/$(egrep ^0.0.$targetVaddr /proc/dasd/devices | awk '{ print $7 }')1
    if [ "$?" != 0 ]; then exit 23; fi
  echo "Mounting $tDev over $targetDir ..."
  mount $tDev $targetDir
  rc=$?
  if [ $rc != 0 ]; then
    echo "Error: 'mount $tDev $targetDir' failed with $rc"
    lsdasd
    CPcmd DET $targetVaddr
    exit 24
  fi
  echo "Modifying cloned image under $targetDir ..."
  sed --in-place -e "s/$sourceHostname/$tagetHostname/g" \
    $targetDir/etc/HOSTNAME
  sed --in-place -e "s/$sourceIP/$targetIP/g" \
    -e "s/${sourceHostname%%.*}/${tagetHostname%%.*}/g" $targetDir/etc/hosts
  sed --in-place -e "s/$sourceGW/$targetGW/g" \
    $targetDir/etc/sysconfig/network/routes
  sed --in-place -e "s/$sourceIP/$targetIP/g" \
    -e "s/$sourceBroadcast/$targetBroadcast/g" \
    $targetDir/etc/sysconfig/network/ifcfg-eth0
  sed --in-place -e "s/$sourceDNS/$targetDNS/g" $targetDir/etc/resolv.conf
```

```
# Delete SSH keys - sshd will recreate them at first boot
  echo "Removing SSH keys"
  rm $targetDir/etc/ssh/ssh_host*

# Remove any old entry, then copy clone's public key to known_hosts file
  echo "Removing $targetIP from known_hosts file"
  cd /root/.ssh
  grep -v $targetIP known_hosts > known_hosts.temp
  mv known_hosts.temp known_hosts

# clean up
  sleep 1
  sync # sync disks
  umount $targetDir
  sync
  chccwdev -d $targetVaddr
  CPcmd DETACH $targetVaddr
  return 0
 } # modifyClone()

# main()
processArguments $@                      # process arguments passed by user
if [ $verbose = "on" ]; then set -vx; fi  # turn on debug
checkID $sourceID                        # user ID must exist and be logged off
checkID $targetID                        # user ID must exist and be logged off
getNetworkInfo                           # get info from parm files
askAreYouSure                            # confirm disks will be overwritten
copySystem                               # copy source disks to target
modifyClone                              # modify newly copied system
echo "sleeping 10 seconds"
sleep 10
CPcmd XAUTOLOG $targetID                 # bring new clone to life
if [ $verbose = "on" ]; then set +vx; fi  # turn off debug
echo "Successfully cloned $sourceID to $targetID"
echo "You should be able to ping $targetIP within one minute"
exit 0
```