Linux/390 System Management for the Mainframe System Programmer

Mark Post
Novell, Inc.

Thursday, August 16, 2007
Session 9224/9225

Agenda

- Cultural Differences & Similarities

- Terminology

- How Does Linux/390 Compare to Unix System Services?

- How Does Linux/390 Compare to Other Linux platforms?

- What is a Distribution, and why doesn't IBM have one?

- What Hardware Does Linux/390 Support?

- What Networking Connections Does Linux/390 Support?

- Where's SYS1.PARMLIB Kept?

- Do I *Really* Have to Know All This Stuff?

Agenda (2)

- Linux/390 DASD Management

- Backup and Restore

- Software and Maintenance Management

- Data Sharing with Linux/390 and Other OS

- Job Scheduling

- Security and User Management

- Diagnostic Information Available (or not)

- Editors

- Basic vi Concepts

Agenda (3)

- Scripting Languages

- System Logs

- Basic Commands

- Advanced Commands

- Kernel Modules, Loading and Unloading

- System Startup and Shutdown

- Performance Management

- Creating Additional Images

- Questions?

Agenda (4)

- Additional Information Resources
  - Web sites, Redbooks/pieces/tips, Usenet (news), O'Reilly Books

- Command Comparisons

Cultural Differences & Similarities

- Open Source Community
  - Think of how many people you know from IBMMAIN or IBMVM

- Open Source software

- Software quality

- Where and how you get support

- Expectations of self-reliance

- 3270 terminals vs. VTxxx.

- Is rebooting unnecessarily a sin?

- Install Linux on a PC and use it.

Terminology

- IPL
- nucleus
- PLPA / LINKLIST
- TSO / CMS
- OSA
- PTF / APAR
- IEBUPDTE
- IEBUPDTE input
- SuperC / COMPARE
- xedit update mode

- boot
- kernel
- kernel modules
- shell
- NIC
- patch
- patch
- diff
- diff
- diff

Terminology (2)

- paging space

- multi-processor

- systems programmer

- GUI

- pattern matching

- help files

- STC / SVM

- address space

- task (tcb)

- swap partition/volume/file

- SMP

- system administrator

- X / X-Window

- regular expression

- man pages

- daemon

- process

- thread

Terminology (3)

- runlevel

- device driver

- tarball (tar - tape archive)
  - tar -cf backup.tar file1 file2 file3 file4
  - tar -xf backup.tar
  - tar -zcf backup.tar.gz file1 file2 file3 file4
    - Equivalent to tar command followed by gzip command
  - tar -zcf backup.tgz file1 file2 file3 file4
  - tar -zxf backup.tgz
  - tar -jcf backup.tar.bz2 file1 file2 file3 file4
    - Uses bzip2 instead of gzip for compression
  - tar –jxf backup.tar.bz2

How Does Linux/390 Compare to...

- Unix System Services
  - Linux/390 is "real" UNIX "under the covers." Things that run on most other Unix systems can be run on Linux/390, usually without change (assuming source is available).
  - There is no such thing as SMP/E or SES in UNIX / Linux. (Not that USS really uses it either, per se.)
  - Option "switches" on various commands are different, due to different shells, or compliance to different standards.
  - The Linux man pages work without extra effort on OS/390 side.
  - There's no confusion about which TCP/IP parameters are put where.
  - Things that you learn about other UNIX systems are generally easier to apply to Linux than USS.
  - There's no "other side" (OS/390) to help bail you out when things get messed up.
  - Native ASCII. No EBCDIC <=> ASCII conversions.

How Does Linux/390 Compare to...

- Unix System Services (cont.)
  - Source code is available.
  - Default shell is more often bash than (t)csh.
  - No 3270 interface/limitations to work around. Which also means no real ISPF. (You can buy a clone of it, though.)
  - PL/1, CLISTs are not available.
  - REXX is available, but called Regina.
  - OREXX is available, now in Open Source form.
  - HLASM is now available from Tachyon Software.

- Other Linux Platforms
  - Very similar, but it lacks a lot of common PC-type hardware
  - 3270 support included for consoles in LPAR mode
  - S/390 specific hardware

What are Distributions?

- VARs

- Packagers

- System Integrators

- Maintainers

- Developers

- Support

- Why doesn't IBM have one?

Linux/390 Hardware Suppport

- Any processor that supports the "Halfword Immediate and Relative Branch Feature" instructions added with the G2.

- For decent performance, IEEE FPU is needed. (G5 and up, MP3000.)

- ECKD DASD

- FBA DASD

- 3480/3490/3590/3592 Tapes
  - Including 3592 hardware encryption

- SCSI over FCP, including tape drives

- Crypto cards and Secure Key Entry

Linux/390 Network Connections

- 3172

- OSA-2 (Token-Ring, Ethernet, Fast Ethernet)

- OSA-Express (Ethernet, Fast Ethernet)

- 2216 (Token Ring, Ethernet)

- QDIO OSA-Express (1 & 10Gb Ethernet, Fast Ethernet)

- ESCON / CTC (native and under VM)

- IUCV (only under VM)

- HiperSockets (native and under VM)

- Guest LANs (only under VM) – NOT the same as VLANs

- VSwitch (only under VM, requires an OSA)

- Cisco CLAW (CIP) - original driver by UTS Global, redone by IBM

Where's SYS1.PARMLIB Kept?

- Just about everything you need is kept under /etc (at some level of hierarchy).

- Individual text files (or groups of them), since no concept of a PDS in Linux.

- Some *really* important ones:

  - /etc/passwd
  - /etc/shadow
  - /etc/group
  - /etc/inittab

  - /etc/inetd.com
  - /etc/modules.conf
  - /etc/fstab
  - /etc/hosts
  - /etc/resolv.conf

  - /etc/rc.d/...
  - /etc/httpd/...
  - /etc/sysconfig/...
  - /etc/samba/...
  - /etc/pam.d/...
  - /etc/ssh/...

Do I *Really* Have to Know All This Stuff?

- No, but shouldn't you?
  - Would you let a junior systems programmer or system operator loose on SYS1.PARMLIB or SYSTEM CONFIG via a GUI?

- If you really don't want to know what's going on or have a large virtual farm:
  - YaST   (SUSE)
  - Nautilus   (Red Hat)
  - GNOME Control Center (SUSE and Red Hat)
  - Webmin   (completely Perl-based)
  - Others

Linux/390 DASD Management

Quick Overview:

- Adding and removing DASD
- Preparing DASD for use
  - Formatting
    - 2 kinds of DASD layouts
  - Partitioning (2.4.x & 2.6.x kernels only)
  - Creating file systems and Swap
  - File systems vs. directories
- Backing up your DASD

Adding & Removing DASD

- For 2.4.x kernels, can be non-disruptive, but manual:
    - Adding a device
    - **echo "add device range=*devno-range* " > /proc/dasd/devices**
    - Disabling a device
    - **echo "set device range=*devno-range* off " > /proc/dasd/devices**
    - Enabling a device
    - **echo "set device range=*devno-range* on " > /proc/dasd/devices**
    - Still want to update /boot/parmfile or /etc/zipl.conf and re-run "zipl" to make the change permanent. (mkinitrd may also be required.)

- The 2.2.x kernels, required updating /boot/parmfile, re-running "silo," and rebooting.

Adding and Removing DASD in 2.6.x

- **New disks detected automatically**

- **/sys/bus/ccw/devices/0.0.0b01@**

- **/sys/bus/ccw/drivers/dasd-eckd/0.0.0b01@**

- **/sys/bus/css/devices/0.0.0016@**

- **/sys/bus/css/drivers/io_subchannel/0.0.0016@**

- **/sys/devices/css0/0.0.0016/**
    - **0.0.0b01/**
      bus@
      chpids
      driver@
      pimpampom

- **/sys/devices/css0/0.0.0016/0.0.0b01/**
    - **Availability**
    - **bus@**
    - **cmb_enable**
    - **cutype**
    - **devtype**
    - **discipline**
    - **driver@**
    - **online**
    - **readonly**
    - **use_diag**

Use /sbin/chccwdev instead

chccwdev --online 0.0.0b01

chccwdev -e 0.0.0b01

chccwdev -d 0.0.0b01

Formatting DASD - 2.4.x, 2.6.x

- dasdfmt -b 4096 [-l volser] [-d layout] -f /dev/dasd?

  - dasdfmt -b 4096   -d cdl -f /dev/dasda
  - dasdfmt -b 4096   -f /dev/dasda
  - dasdfmt -b 4096   -d ldl -f /dev/dasda

- fdasd /dev/dasd?
  - **Must** create one, two, or three partitions for CDL-formatted volumes
  - Similar to the fdisk command
    - fdisk is still used for SCSI disks!

Creating File Systems and Swap

2.4.x, 2.6.x Kernels

- mke2fs -b 4096 /dev/dasd?1,2,3
  - mke2fs -b 4096 /dev/dasda1
  - mke2fs -b 4096 /dev/dasda2
  - mke2fs -b 4096 /dev/dasda3
  - mke2fs -b 4096 /dev/dasdb1
  - mke2fs -b 4096 /dev/dasdc1

- Other file system types possible (Red Hat only supports EXT2 & EXT3)
  - mkfs.reiserfs
  - mkfs.ext3
  - mkfs.xfs
  - mkfs.jfs (deprecated in SLES10)

- mkswap /dev/dasd?1,2,3

File Systems and Directories

/   (root)

/bin

/boot *

/dev

/etc

/home

/lib, /lib64

/mnt

/opt

/proc

/root (not to be confused
        with / root)

/sbin

/srv

/sys

/tmp

/usr

/var

22

Backup and Restore

- Native Linux facilities
  - afio/cpio/tar

- Software packages (Open Source and proprietary)
  - Bacula, Amanda
  - DFSMSdss / DDR
  - Tivoli TSM/ADSM
  - Innovation FDRINSTANT/UPSTREAM
  - CA BrightStor
  - SecureAgent SecureBackup
  - UTS Global TSS-BAR
  - Veritas NetBackup

- Note that unless you have the backup *server* on the local system and tape drives available, backup data will be sent over the network.

Software & Maintenance Management

- Since there is no such thing as SMP/E or SES, you have to learn a new mindset.

- There are "binary" packages for several platforms, primarily Intel. Different distributions use different methods to manage those packages: RPM, dpkg.

- Source packages (RPM, tar.gz, etc.) are always available for Open Source software. This is *not* an invitation to compile from source and install without using the same packaging tool as your distribution provider uses.

Software & Maintenance Management (2)

- In a number of cases, there is no binary available for Linux/390. Compiling from source is the only option. This can be **very** easy or **very** difficult, depending on the package.

- You will become very familiar with the 'tar,' 'gzip,' and 'make' commands. Most likely 'patch' and 'diff' as well.

- Once you start down that road, you are no longer a Linux user, but a Linux developer, with all the pain that can entail.

Software & Maintenance Management (3)

- Keeping track of security patches is **very** important, and people-intensive.

- Getting email notifications from trusted security industry sources (CERT, etc.) is recommended.

- If you *have* to compile a package from source, it probably won't be too difficult.
  - tar -zxf package.name.tar.gz
  - cd package-directory
  - ./configure
  - make, and then make install

Using RPM

- VERY high-level!
  - rpm -i package.name.rpm   (install)
  - rpm -e package.name   (remove)
  - rpm -q package.name   (query)
  - rpm -ql package.name
  - rpm -qlp package.name.rpm
  - rpm -qa
  - rpm -qf /path/to/file/name

- dpkg on Debian-based systems

27

Data Sharing With Linux and Other OSes

- No direct, hard-wired sharing

- z/OS doesn't "know" ext2
  - z/VM does, but only usable when the Linux system is down

- Linux does "know" VTOCS, etc.
  - But, no security! (your audit compliance folks won't like you)

- Various network-based methods and clustering file systems
  - NFS
  - AFS
  - Samba (SMB / CIFS / MS Networking)
  - GFS, OCFS2 (OCFS2 is not quite ready for production use)

- Under VM - sharing minidisks *read-only* between guests.

28

Job Scheduling

- Linux native facilities
  - cron
  - at

- Open Source:
  - DQS
  - queue
  - OpenPBS
  - generic NQS

- Proprietary
  - Avatar
  - Computer Associates
  - ESP
  - Jobtrac
  - PBSPro
  - UC4:global
  - Xi-Batch

29

Security and User Management

- Security in an ongoing process, not a status. It must be constantly attended to for you to have any chance at all.

- Most successful security breaches come from employees of a company, not outsiders.

- In contrast to typical mainframe security, Linux security is more network oriented.

- If possible, have a UNIX security person handle your Linux security needs.

Security and User Management (2)

- Turn off *ALL* unnecessary services: telnet, ftp, smtp, time, finger, http, pop3, imap, login, shell, printer, nfs, etc., etc.

- Use OpenSSH instead of telnet, ftp, rlogin, rsh, rexec, rlogin, etc..

- Use TCP Wrappers (/etc/hosts.allow, etc.) if you have to use an insecure protocol.

- Review your system logs regularly.

- Monitor security alerts from your suppliers, and from various security organizations.

Security and User Management (3)

- Don't lump all your users into one group (typically "users" - SUSE default).

- Don't create a separate group for each user (Red Hat's approach).

- Try to have reasonable groups defined so that people can share data appropriately, and put the proper users into them.

- Don't give *anyone* a UID of zero unless it's absolutely necessary (and even then think about other ways to avoid it).

Security and User Management (4)

- Various tools are available for adding, deleting and changing user and group definitions. All information about users and groups are in plain text files.

- SUSE has YaST

- Red Hat has Nautilus and redhat-config-* tools

- Webmin is popular

- useradd, userdel, usermod, groupadd, groupdel, groupmod are the common command-line tools

Security and User Management (5)

- Protect the password of "root" very carefully.

- Login as "yourself" and su to root only when really needed.

- Consider using /etc/sudoers to grant some selected command authority to designated people.

- If you're running a Linux that doesn't use PAM (pluggable authentication modules), consider using /etc/suauth to allow designated people to "su" using their own password.

Diagnostic Information

- strace

- ulimit (to enable core dumps)

- gdb

- uptime

- top

- ksymoops

- siga (SUSE)

- netstat

- ping

- traceroute

- system logs

- dmesg

- standalone dump

- sysreport (Red Hat)

Editors (Holy War fodder)

- vi / vim / elvis

- emacs / xemacs

- joe

- jed

- jove

- ed (sed)

- nano

- pico

- ne

- ned (3270 enabled)

- Nedit

- THE (The Hessling Editor)

No native free (no cost) ISPF/PDF clones.
(2 proprietary ones)

Basic vi Concepts

- Cursor keys work as expected (or h-j-k-l), as do Page up and Page Down, Delete and Backspace (when ssh client is properly configured).

- Two important modes: command, insert.

- I'm pretty unfamiliar with vi, so I basically use insert mode and command mode.

- Insert button = insert mode (twice = replace)

- ESC = exit insert/command mode.

Basic vi Concepts (2)

- :set smd   or :set showmode
  - gives visual indicator of what mode you're in

- :d = delete a line

- :w = write updated file to disk

- :x = write updated file to disk and exit

- :q = quit if no updates have been made since the last save (:w)

- :q! = quit regardless

- :help = help me!

Scripting Languages (Holy War Cont.)

- Perl

- ash / bash / csh / tcsh / ksh / ksh93 / zsh

- Regina (REXX)

- OREXX

- Python (can also be compiled)

- Ruby

- Tcl

- The list goes on...

39

System Logs

- Most of what you want will be in /var/log/, or in a subdirectory of it.

- Names and contents vary by distribution
  - Look at /etc/syslog.conf if you're interested

- Reviewing them *frequently* is important

- Samples:

  - /var/log/messages
  - /var/log/syslog
  - /var/log/debug
  - /var/log/boot.log
  - /var/log/dmesg

  - /var/log/proftpd.log
  - /var/log/maillog
  - /var/log/warn
  - /var/log/httpd/...
  - /var/log/samba/...

Basic Commands

- **rm -rf /**
- cd
- cp
- mv
- rm
- ls
- find

- grep
- cat
- less / more
- man
- info
- mount
- umount

- mkdir
- rmdir
- ps
- pushd
- popd
- which
- zipl

Advanced Commands

- ifconfig
- route
- host / nslookup
- su
- sudo
- gzip / bzip2
- last

- chmod
- chown
- dmesg
- du
- df
- locate
- top

- sed
- head
- tail
- wget
- export
- file
- telinit

Kernel Modules

- The kernel contains all the code necessary to run as an operating system. This includes device drivers, file system drivers, networking code, etc.

- Kernel modules are parts of the kernel that don't have to be compiled into the kernel itself (but can be if you want to).

- This can allow you to update these components on the fly, without rebooting.

Loading Kernel Modules

- Since modules are *not* in the kernel, how do they get loaded into kernel storage?
    - insmod
    - modprobe
    - dynamically by the kernel if /etc/modules.conf or /etc/modprobe.conf has the right data in it.

- How does the module get it's parameters?
    - provided on insmod command
    - read from /etc/modules.conf or /etc/modprobe.conf by modprobe and the kernel.

Loading Kernel Modules

- How does the kernel find the module?
  - Usually in /lib/modules/kernelver/...
    - /lib/modules/2.6.16.27-0.9-smp/
  - depmod -a command creates a cross reference of the modules and where they are located, and writes it into /lib/modules/kernelver/modules.??? files.
  - modprobe figures out the dependencies and loads the modules in the right order. insmod does not do this.

Unloading Kernel Modules

- The rmmod command is used to do the unloading

- Kernel modules must be unused to be unloaded
  - The jbd module is in use by the ext3 module and cannot be unloaded
    # lsmod
    jbd   62880 1 ext3

- There is a -f (force) parameter to rmmod, but under normal circumstances it would be unwise to use it.
  - CONFIG_MODULE_FORCE_UNLOAD=Y must have been specified in he kernel config for it to work.

System Startup and Shutdown

- From the HMC, just like any other OS, except no loadparms are needed/used.

- From VM, there are usually some CMS/CP commands issued in PROFILE EXEC, followed by a
'CP IPL devno CLEAR'

- From VM, you can still IPL from the reader, if desired.

- snIPL (simple network IPL)

- How the rest of the system is brought up by init is a very complex process, beyond the scope of this talk.
  - See if there are any handouts left from "9274 – The Linux IPL Procedure" which was given this morning.

System Startup and Shutdown (2)

- When shutting down, it is important to do it *properly*. File system corruption and data loss can result otherwise.

- Use the halt, reboot or shutdown command.

- You can use shutdown to warn any logged on users, and/or set the shutdown to some time in the future.

- The shutdown command has options to reboot, halt, or cancel a previous shutdown command.

- The ctrlaltdel entry in /etc/inittab can be used to automate this by listening for LPAR or z/VM shutdown signals:
  ```
  ca::ctrlaltdel:/sbin/shutdown -h -t 4 now
  ```
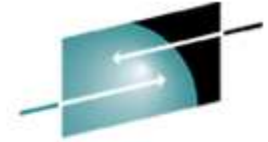
Performance Management

- Some "standalone" native Linux tools:
  - top, ntop, httperf, sar, iostat, gkrellm, pload, statnet, Big Brother, Hobbit, iptraf
  - Understand that in a shared environment, such as z/VM (or even LPAR), these will *all* be lying to you in certain ways. SLES10 and RHEL5 are the first versions that use a kernel that has the code necessary to correct this.

- Proprietary products
  - Velocity Software ESALPS
  - BMC MAINVIEW
  - IBM Performance Tookit
  - OMEGAMON XE

49

Creating Additional Images

- Some commercial tools, none of them cheap

- Can be done with home grown tools

- IBM Redbook "Linux on IBM zSeries and S/390: ISP/ASP Solutions," SG24-6299

- z/VM and Linux on IBM System z: The Virtualization Cookbook for SLES10

- z/VM and Linux on IBM System z: The Virtualization Cookbook for RHEL5

Questions?

Additional Information - Web Sites

- http://linuxvm.org/

- http://www.marist.edu/htbin/wlvindex?linux-390
  (Linux/390 mailing list subscription and archives)

- http://www.slack390.org/

- http://www.kernel.org/

- http://www.linux.org/

- http://www.tldp.org/
  (The Linux Documentation Project)

Additional Information - Redbooks

- http://www.redbooks.ibm.com/

- Linux for S/390, SG24-4987

- Linux for zSeries and S/390: Distributions,
    SG24-6264

- Linux on zSeries and S/390: ISP/ASP Solutions,
    SG24-6299

- Linux on zSeries and S/390: Application Development,
    SG24-6807

- Linux on zSeries and S/390: System Management,
    SG24-6820

Additional Information - Redbooks (2)

- Linux on zSeries and S/390: Large Scale Linux Deployment, SG24-6824

- Linux on zSeries and S/390: Performance Measurement and Tuning, SG24-6926

- Linux with zSeries and ESS: Essentials, SG24-7025

- Experiences with Oracle for Linux on zSeries, SG24-6552

- SAP on DB2 UDB for OS/390 and z/OS: Implementing Application Servers on Linux for zSeries, SG24-6847

Additional Information - Redbooks (3)

- e-Business Intelligence: Leveraging DB2 for Linux on S/390, SG24-5687

- e-Business Intelligence: Data Mart Solutions with DB2 for Linux on zSeries, SG24-6294

Additional Information - Redpieces

- Implementing Linux in your Network using Samba, redp0023

- Building Linux Systems Under IBM VM, redp0120

- Linux on zSeries and S/390: High Availability for z/VM and Linux, redp0220

- Linux on zSeries and S/390: Securing Linux for zSeries with a Central z/OS LDAP Server (RACF), redp0221

- Linux on zSeries and S/390: Server Consolidation with Linux for zSeries, redp0222

- Linux on zSeries and S/390: Cloning Linux Images in z/VM, redp0301

Additional Information - Redpieces (2)

- Linux on zSeries and S/390: TCP/IP Broadcast on z/VM Guest LAN, redp3596

- Linux on zSeries and S/390: Managing a Samba Server from z/VM, redp3604

- Linux on zSeries and S/390: Porting LEAF to Linux on zSeries, redp3627

- Linux on zSeries and S/390: Virtual Router Redundancy Protocol on VM Guest LANs, redp3657

- Linux on zSeries and S/390: z/VM Configuration for WebSphere Deployments, redp3661

Additional Information - Redpieces (3)

- Linux on zSeries and S/390: Building SuSE SLES8 Systems under z/VM, redp3687

- Linux on zSeries and S/390: VSWITCH and VLAN Features of z/VM 4.4, redp3719

- e-commerce Patterns for Linux on zSeries Using WebSphere Commerce Suite V5.1 Patterns for e-business series, redp0411

- Getting Started with zSeries Fibre Channel Protocol, redp0205

- WebSphere Portal Installation on Linux for zSeries, redp3699

- Open Your Windows with Samba on Linux, redp3780

Additional Information - Hints & Tips

- Linux on zSeries: Configuring gcc as a cross-compiler, tips0005

- Dynamic management of DASD devices in Linux running on zSeries, tips0023

- Formatting and Labeling a DASD Volume for Linux Guests Running Under z/VM, tips0275

- Partitioning DASD for Linux Guests Running under z/VM, tips0277

Additional Information - O'Reilly Books

* Apache: The Definitive Guide, 2nd Edition
   Apache Pocket Reference
* Building Internet Firewalls, 2nd Ed
* DNS and BIND, 4th Edition
* Learning Perl, 3rd Edition
* Learning the bash Shell, 2nd Edition
* Learning the vi Editor, 6th Edition
   MySQL & mSQL
* Perl Cookbook
   Perl for System Administration
   Perl for Web Site Management
   Perl in a Nutshell

Additional Information - O'Reilly Books

* Practical UNIX & Internet Security, 2nd Edition
* Programming Perl, 3rd Edition
* Running Linux, 3rd Edition
  Samba Pocket Reference
  sed & awk Pocket Reference
* sed & awk, 2nd Edition
* sendmail, 2nd Edition
  sendmail Desktop Reference
* SSH, The Secure Shell: The Definitive Guide
* TCP/IP Network Administration, 2nd Edition
* Using Samba - comes with the software
  vi Editor Pocket Reference

Additional Information - O'Reilly Books

* Networking CD Bookshelf
    TCP/IP Network Administration, 2nd Edition
    sendmail, 2nd Edition
    sendmail Desktop Reference
    DNS and BIND, 3rd Edition
    Practical UNIX & Internet Security, 2nd Edition
    Building Internet Firewalls

* The Perl CD Bookshelf, Version 2.0
    Programming Perl, 3rd Edition
    Perl for System Administration
    Perl in a Nutshell
    Perl Cookbook
    Advanced Perl Programming

Additional Information - O'Reilly Books

* The Linux Web Server CD Bookshelf
    Running Linux, 3rd Edition
    Linux in a Nutshell, 3rd Edition
    Apache: The Definitive Guide, 2nd Edition
    MySQL & mSQL
    Programming the Perl DBI
    CGI Programming with Perl, 2nd Edition

63

Command Comparison

MVS	VM	Linux	DOS
LISTC	L	ls / locate	dir
LISTD	L (L	file	attrib
LIST	TYPE	cat	type
COPY	COPY	cp	copy
MOVE	MOVE	mv	move
RENAME	RENAME	mv	ren
DELETE	ERASE	rm	del
HELP	HELP	man / info	help
ICKDSF	FORMAT	dasdfmt / mke2fs	format

Command Comparison (2)

MVS	VM	Linux
SEND	TELL / MSG / NOTE	write / talk / wall
LISTB		/etc/motd
LOGOFF	LOGOFF	exit
PRINTDS	PRINT	lpr
V ONLINE	ATTACH	mount
V OFFLINE	DETACH	umount
D TS	Q U	uptime
D TS,L	Q N	users / w
D A,L	Q N	ps -ax

Command Comparison (3)

Linux	DOS	Linux	DOS
• cd	• cd	• host	• nslookup
• mkdir	• mkdir	• nslookup	• nslookup
• rmdir	• rmdir	• netstat	• netstat
• less / more	• more	• route	• route print
• ping	• ping	• find	• find
• traceroute	• tracert	• set	• set
		• export	• set