Linux System Management for the Mainframe Systems Programmer

Mark Post
Novell, Inc.

Thursday, August 11, 2011
Session 9884 and 10085
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?
- Linux DASD Management
Agenda (2)

• 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
• Scripting Languages
• System Logs
Agenda (3)

- Basic Commands
- Advanced Commands
- Kernel Modules, Loading and Unloading
- System Startup and Shutdown
- Performance Management
- Creating Additional Images
- Questions?
- Additional Information Resources
  - Web sites, Redbooks/pieces/tips, O’Reilly Books
- Command Comparisons
Cultural Differences and Similarities

- Open Source Community
  - Think how many people you know from IBM-MAIN 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.
  - Where your email is, will be where you spend your most time.
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)

- 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.
  - 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 z/OS side.
  - There's no confusion about which TCP/IP parameters get used where.
  - Things that you learn about other UNIX systems are generally easier to apply to Linux than USS.
How Does Linux/390 Compare to...

- Unix System Services (cont.)
  - There's no "other side" (z/OS) to help bail you out when things get messed up.
  - Native ASCII. No EBCDIC <=> ASCII conversions.
  - Source code is typically available on Linux.
  - Default shell is more often bash than (t)csh.
  - No 3270 interface/limitations to work around. Which also means no ISPF or PDF.
  - PL/1, CLISTs are not available.
  - REXX is available, but called Regina.
  - OREXX is available, now in Open Source form.
How Does Linux/390 Compare to...

• Other Linux Platforms
  • Very similar, but it lacks a lot of common PC-type hardware
  • 3270 support included for consoles in LPAR mode
  • System z specific hardware
What are Distributions?

- VARs
- Packagers
- System Integrators
- Maintainers
- Developers
- Technical Support
- Why doesn’t IBM have one?
What Hardware Does Linux/390 Support?

- Any processor that supports the "Halfword Immediate and Relative Branch Feature" instructions added with the G2.
  - Not too many people have to worry about this any more ;)
- For decent performance, IEEE FPU is needed. (G5 and up, MP3000.)
- ECKD and FBA DASD
- 3480/3490/3590/3592 Tapes
  - Including 3592 hardware encryption
- SCSI over FCP, including tape drives and CD/DVD drives
- Crypto cards and Secure Key Entry
- 3270 and 3215 consoles
- Card reader/punch (2540) and printer (1403)
What Networking Connections Does Linux/390 Support?

- 3172
- OSA-2 (Token-Ring, Ethernet, Fast Ethernet)
- OSA-Express (Ethernet, Fast Ethernet)
- 2216 (Token Ring, Ethernet)
- QDIO OSA-Express (and -2 -3) (1 & 10Gb Ethernet, Fast Ethernet)
- ESCON / CTC (native and under z/VM)
- IUCV (z/VM only)
- HiperSockets (native and under z/VM)
- Guest LANs (z/VM only) – NOT the same as VLANs
- VSwitch (z/VM only, requires an OSA for external traffic)
- 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 fairly important ones:
  - /etc/passwd
  - /etc/shadow
  - /etc/group
  - /etc/inittab
  - /etc/inetd.conf
  - /etc/modules.conf
  - /etc/fstab
  - /etc/hosts
  - /etc/resolv.conf
  - /etc/rc.d/...
  - /etc/httpd/...
  - /etc/samba/...
  - /etc/pam.d/...
  - /etc/ssh/...
  - /etc/xinet.d/...
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 DASD Management

Quick Overview:

• Adding and removing DASD
• Preparing DASD for use
• Sample file system layout
• Backing up your DASD
Adding and Removing DASD for 2.4 Kernels (SLES 8, RHEL 3)

- Can be dynamic (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 will likely also be required.)
  - If you do this from YaST, it is done for you
Adding and Removing DASD for 2.6 Kernels (SLES 9 and up, RHEL 4 and up)

- Detection and control block building is dynamic and automatic.
  - Could be as a result of an HMC operation or z/VM ATTACH
- Bringing the device online or offline is manual
  - Use /sbin/chccwdev (change ccw device)
    - chccwdev --online 0.0.0b01
    - chccwdev -e 0.0.0b01
    - chccwdev --disable 0.0.0b01
    - chccwdev -d 0.0.0b01
  - Using YaST to activate is best
Formatting DASD - 2.4.x and 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!
  - `fdasd -a /dev/dasd`?
    - Automatically creates one partition using the entire disk in non-interactive mode
Creating File Systems

- 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, going away in SLES11)
Creating Swap

- mkswap /dev/dasd?1,2,3
  - mkswap /dev/dasda1
  - mkswap /dev/dasdb3
- For z/VM shops, VDISK is very good for Linux paging volumes
  - Make sure you have enough real storage for this
  - Make sure you have a good z/VM performance monitor to watch out for problems
  - Use SWAPGEN EXEC from Sine Nomine Associates to initialize the VDISK from CMS before booting into Linux
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
Sample File System Layout

```
# df -h

Filesystem      Size  Used  Avail  Use% Mounted on
/dev/dasda1     388M  119M  250M   33% /           
/dev/dasda2      97M   4.2M  88M    5% /home       
/dev/dasda3      74M   21M  50M    30% /opt        
/dev/dasdc1     1.2G  1.1G  100M   92% /srv        
/dev/dasdb1     291M  17M  260M    6% /tmp        
/dev/dasdb2     1.2G  915M  183M   84% /usr        
/dev/dasdb3     245M  69M  164M   30% /var        
```
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 are available, backup data will be sent over the network.
Software and 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.
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.

Part of the pain will be acting as the maintainer of that package, since you won't be getting support for it elsewhere.
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`
- For SUSE and Red Hat, the `rpmbuild` command does all this

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

Getting email notifications from trusted security industry sources (CERT, etc.) is recommended.
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
• pkgtools on Slack/390 systems
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)
  - OCFS2, GFS
- Under VM - sharing minidisks read-only between guests.
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
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, rcp, 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
- vmstat
- supportconfig/siga (SUSE)
- iostat
- sar
- 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)
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 Part 2)

- Perl
- ash / bash / csh / tcsh / ksh / ksh93 / zsh
- Regina (REXX)
- OREXX (31-bit only at the moment)
- Python (can also be compiled)
- Ruby
- Tcl
- The list goes on...
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`
- `grep`
- `cat`
- `less / more`
- `man`
- `info`
- `mount`
- `umount`
- `ps`
- `mkdir`
- `rmdir`
- `pushd`
- `popd`
- `which`
- `reboot`
- `shutdown`
- `rm -rf /`
Advanced Commands

• chmod
• chown
• df
• du
• file
• head
• tail

• ifconfig
• route
• host / nslookup
• gzip / bzip2
• find
• locate
• dmesg

• su
• sudo
• telinit
• lsmod
• modinfo
• mknitrd
• zipl
Kernel Modules

- The kernel, along with kernel modules, 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/modprobe` command
  • read from `/etc/modules.conf` or `/etc/modprobe.conf` by `modprobe` and the kernel.
Loading Kernel Modules (2)

• 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 | grep ext3
    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 the 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.
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, httpperf, sar, iostat, gkrellm, pload, statnet, Big Brother, Hobbit, iptraf, sysstat, vmstat
  • 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
Creating Additional Images

- AutoYaST (SUSE) and Kickstart (RHEL)
- Some commercial tools, none of them cheap
- Can be done with home grown tools
- IBM Redbooks:
  - Linux on IBM zSeries and S/390: ISP/ASP Solutions, SG24-6299
  - z/VM and Linux on IBM System z The Virtualization Cookbook for SLES 10 SP2, SG24-7493
  - z/VM and Linux on IBM System z The Virtualization Cookbook for Red Hat Enterprise Linux 5.2, SG24-7492
Questions?
Additional Information - Web Sites

- http://linuxvm.org/
  (Linux/390 mailing list subscription and archives)
- http://www.kernel.org/
- http://www.linux.org/
- http://www.tldp.org/
  (The Linux Documentation Project)
Additional Information - Redbooks

- 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
- Linux on zSeries and S/390: Large Scale Linux Deployment, SG24-6824
- Linux on zSeries and S/390: Performance Measurement and Tuning, SG24-6926
Additional Information – Rebooks (2)

• z/VM and Linux on IBM System z The Virtualization Cookbook for SLES 10 SP2, SG24-7493
• z/VM and Linux on IBM System z: The Virtualization Cookbook for Red Hat Enterprise Linux 5.2, SG24-7492
• Linux for IBM System z9 and IBM zSeries, SG24-6694
• Linux with zSeries and ESS: Essentials, SG24-7025
• Fibre Channel Protocol for Linux and z/VM on IBM System z, SG24-7266
• HiperSockets Implementation Guide, SG24-6816
• Problem Determination for Linux on System z, SG24-7599
• Achieving High Availability on Linux for System z with Linux-HA Release 2, SG24-7711
Additional Information – Rebooks (3)

- Experiences with Oracle for Linux on zSeries, SG24-6552
- Experiences with Oracle 10g Database for Linux on zSeries, SG24-6482
- Using Oracle Solutions on Linux for System z, SG24-7573
- Using Discontiguous Shared Segments and XIP2 Filesystems With Oracle Database 10g on Linux for IBM System z, SG24-7285
- 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 – Rebooks (4)

- SAP on DB2 UDB for OS/390 and z/OS: Implementing Application Servers on Linux for zSeries, SG24-6847
- Linux on IBM eServer zSeries and S/390: Best Security Practices, SG24-7023
- Introduction to the New Mainframe: Security, SG24-6776
- Introduction to the New Mainframe: z/VM Basics, SG24-7316
- Running Linux on IBM System z9 and zSeries under z/VM, SG24-6311
- z/VM and Linux Operations for z/OS System Programmers, SG24-7603
Additional Information - Redpieces

- Advanced LDAP User Authentication: Limiting Access to Linux Systems Using the Host Attribute, REDP-3863
- Linux on IBM zSeries and S/390: Securing Linux for zSeries with a Central z/OS LDAP Server (RACF), REDP-0221
- Linux on IBM eServer zSeries and S/390: Virtual Router Redundancy Protocol on VM Guest LANs, REDP-3657
- Linux on IBM eServer zSeries and S/390: VSWITCH and VLAN Features of z/VM 4.4, REDP-3719
- Networking Overview for Linux on zSeries, REDP-3901
- Sharing and maintaining Linux under z/VM, REDP-4322
Additional Information - Redpieces (2)

- A Shared WebSphere Application Server Installation for Linux on zSeries, REDP-3998
- WebSphere Portal Installation on Linux for zSeries, REDP-3699
- WebSphere Portal Server for Linux on zSeries and z9, REDP-4175
- Linux on IBM eServer zSeries and S/390: z/VM Configuration for WebSphere Deployments, REDP-3661
- e-commerce Patterns for Linux on zSeries Using WebSphere Commerce Suite V5.1 Patterns for e-business series, REDP-0411
Additional Information - Redpieces (3)

- Accounting and Monitoring for z/VM Linux guest machines, REDP-3818
- Linux Performance and Tuning Guidelines, REDP-4285
- Cloning FCP-attached SCSI Disks on SLES9 Linux for zSeries Systems, REDP-3871
- Introducing N_Port Identifier Virtualization for IBM System z9, REDP-4125
- Using Cryptographic Adapters for Web Servers with Linux on IBM System z9 and zSeries, REDP-4131
- Using the Oracle Grid Control Server with a Database on Linux for System z, REDP-4443
Additional Information - Hints & Tips

- 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
- Configuring Logical Volume Management (LVM) on Linux for zSeries, TIPS0128
- Creating a User Home directory when you use LDAP Authentication, TIPS0410
Additional Information - Hints & Tips (2)

• IBM eServer zSeries Linux Fibre Channel Adapter Device Driver Configuration for Tape Devices, TIPS0249
• The CP SET LOADDEV command, TIPS0440
• Capacity Test of IFL vs. CP, TIPS0479
• Securing VNC Network Traffic Using SSH Port Forwarding, TIPS0408
• Installing Oracle 10gR2 on SLES10 Linux on System z - July 10, 2007, TIPS0669
Additional Information - O'Reilly Books

- Apache: The Definitive Guide
- Building Internet Firewalls
- DNS and BIND
- Learning the bash Shell
- Learning the vi and Vim Editors
- Learning Perl
- Perl Cookbook
- Perl for System Administration
- Programming Perl
Additional Information - O'Reilly Books (2)

- Practical UNIX & Internet Security
- Running Linux
- sed & awk
- sendmail
- sendmail Desktop Reference
- SSH, The Secure Shell: The Definitive Guide
- TCP/IP Network Administration
- Using Samba
## Command Comparison

<table>
<thead>
<tr>
<th>MVS</th>
<th>VM</th>
<th>Linux</th>
<th>DOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISTC</td>
<td>L</td>
<td>ls / locate</td>
<td>dir</td>
</tr>
<tr>
<td>LISTD</td>
<td>L (L</td>
<td>file</td>
<td>attrib</td>
</tr>
<tr>
<td>LIST</td>
<td>TYPE</td>
<td>cat</td>
<td>type</td>
</tr>
<tr>
<td>COPY</td>
<td>COPY</td>
<td>cp</td>
<td>copy</td>
</tr>
<tr>
<td>MOVE</td>
<td>MOVE</td>
<td>mv</td>
<td>move</td>
</tr>
<tr>
<td>RENAME</td>
<td>RENAME</td>
<td>mv</td>
<td>ren</td>
</tr>
<tr>
<td>DELETE</td>
<td>ERASE</td>
<td>rm</td>
<td>del</td>
</tr>
<tr>
<td>HELP</td>
<td>HELP</td>
<td>man / info</td>
<td>help</td>
</tr>
<tr>
<td>ICKDSF</td>
<td>FORMAT</td>
<td>dasdfmt / mke2fs</td>
<td>format</td>
</tr>
</tbody>
</table>
## Command Comparison (2)

<table>
<thead>
<tr>
<th>MVS</th>
<th>VM</th>
<th>Linux</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEND</td>
<td>TELL / MSG / NOTE</td>
<td>write / talk / wall</td>
</tr>
<tr>
<td>LISTB</td>
<td>LOGOFF</td>
<td>/etc/motd</td>
</tr>
<tr>
<td>LOGOFF</td>
<td>PRINT</td>
<td>exit</td>
</tr>
<tr>
<td>PRINTDS</td>
<td>ATTACH</td>
<td>lpr</td>
</tr>
<tr>
<td>V ONLINE</td>
<td>DETACH</td>
<td>mount</td>
</tr>
<tr>
<td>V OFFLINE</td>
<td>Q U</td>
<td>umount</td>
</tr>
<tr>
<td>D TS</td>
<td>Q N</td>
<td>uptime</td>
</tr>
<tr>
<td>D TS,L</td>
<td>Q N</td>
<td>users / w</td>
</tr>
<tr>
<td>D A,L</td>
<td></td>
<td>ps -ax</td>
</tr>
</tbody>
</table>
## Command Comparison (3)

<table>
<thead>
<tr>
<th>Linux</th>
<th>DOS</th>
<th>Linux</th>
<th>DOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>• cd</td>
<td>• cd</td>
<td>• host</td>
<td>• nslookup</td>
</tr>
<tr>
<td>• mkdir</td>
<td>• mkdir</td>
<td>• nslookup</td>
<td>• nslookup</td>
</tr>
<tr>
<td>• rmdir</td>
<td>• rmdir</td>
<td>• netstat</td>
<td>• netstat</td>
</tr>
<tr>
<td>• less / more</td>
<td>• more</td>
<td>• route</td>
<td>• route print</td>
</tr>
<tr>
<td>• ping</td>
<td>• ping</td>
<td>• find</td>
<td>• find</td>
</tr>
<tr>
<td>• traceroute</td>
<td>• tracert</td>
<td>• set</td>
<td>• set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• export</td>
<td>• set</td>
</tr>
</tbody>
</table>