Linux/390 System Management for the Mainframe System Programmer

I know how to do “abc” with my usual OS. How do I do that with Linux/390?

Session 9224/9225
About me

• Mark Post (mark.post@eds.com)
• Senior Infrastructure Specialist (Systems Programmer, “plus”)
• EDS, Auburn Hills, Michigan
• Worked with IBM mainframes since entering college in 1973.
• Supported MVS and VM for GM and EDS
• Linux and Linux/390 Technical Lead since 01/2003
My Linux background

- First installed Slackware Linux on a PC at home in 1998.
- Involved with Linux/390 since May of 2000.
- Co-author of IBM Redbook: “Linux for zSeries and S/390: Distributions,” SG24-6264
- Active in the Linux-390 mailing list hosted by Marist College.
- Webmaster for the linuxvm.org web site.
Agenda

• Cultural differences
• 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?
Agenda (2)

- What networking connections does Linux/390 support?
- Where’s SYS1.PARMLIB kept?
- Do I really have to know all this stuff?
- Linux/390 DASD management
- Backup and restore
- Software and maintenance management
- Data sharing with Linux/390 and other OS
Agenda (3)

- Job scheduling
- Security and user management
- Diagnostic information available (or not)
- Editors
- Basic vi concepts
- Scripting languages
- System logs
- Basic commands
Agenda (4)

• Advanced commands
• Kernel modules, insmod and modprobe
• System startup and shutdown
• Performance management
• Creating additional images
• Additional information resources
  – Web sites, Redbooks/pieces/tips, Usenet (news), O’Reilly Books
• Command comparisons
Cultural differences

- Open Source Community
- Open Source software
- Software quality
- Where and how you get support
- Expectations of self-reliance
- 3270 terminals vs VTxxx.
- Install Linux on a PC and use it.
- Is rebooting unnecessarily a sin?
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
- diff
- diff
- diff
Terminology (2)

- paging space
- multi-processor
- systems Programmer
- GUI
- pattern matching
- help files
- STC / VSM
- address space
- task (tcb)
- swap 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 -jcf backup.tar.bz2 file1 file2 file3 file4`
  - `tar -zxf backup.tgz`
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 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.
How does Linux/390 compare to...

• Unix System Services (cont.)
  – 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.
How does Linux/390 compare to...

- **Unix System Services (cont.)**
  - Native ASCII. No EBCDIC <> ASCII conversions.
  - 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.
  - HLASM is now available from Tachyon Software
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
  - S/390 specific hardware
What are distributions?

- VARs
- System Integrators
- Packagers
- Maintainers
- Developers
- Support
- Why doesn’t IBM have one?
Linux/390 hardware support

- 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 Tapes
- SCSI over FCP now available
Linux/390 networking connections

- 3172
- OSA-2 (Token-Ring, Ethernet, Fast Ethernet)
- OSA-Express (Ethernet, Fast Ethernet)
- 2216 (Token Ring, Ethernet)
- QDIO OSA-Express (Gb Ethernet, Fast Ethernet)
- ESCON / CTC (native and under VM)
- IUCV (only under VM)
- HiperSockets (native and under VM)
- Guest LANs (only under VM)
- Cisco CLAW (CIP) - driver by UTS Global
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/gshadow
  - /etc/inittab
  - /etc/inetd.conf
  - /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?
- If you really don’t want to know what’s going on or have a large virtual farm:
  - YaST2 (SuSE)
  - Nautilus (Red Hat)
  - Webmin (completely perl-based)
  - AdminUX (Green Light Advantage)
  - linuxconf (no longer recommended by RH)
  - Others
Linux/390 DASD management

- 2 kinds of DASD layouts, cdl and ldl.
- Adding/removing DASD
- Preparing DASD for use
  - Formatting
  - Partitioning (2.4.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 dynamic:
  – 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` and re-run “zipl” to make the change permanent.

• For 2.2.x kernels, requires updating `/boot/parmfile`, re-running “silo,” and rebooting.
Formatting DASD - 2.4.x

- **dasdfmt** -b 4096 [ -l volser ] [ -d layout ]
  -f /dev/dasd?
  -n 0d18 (only if devfs is in use)
    - dasdfmt -b 4096 -d cdl -f /dev/dasda
    - dasdfmt -b 4096 -d ldl -n 0cf3
    - dasdfmt -b 4096 -d cdl -n 0d2f

- **fdasd** /dev/dasd?
  - *Must* create one, two, or three partitions
Formatting DASD - 2.2.x

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

- `dasdfmt -b 4096 [ -l volser ] -n fc23`
Creating file systems and Swap

• 2.4.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

• 2.2.x Kernels
  – mke2fs -b 4096 /dev/dasd?1
    ▪ mke2fs -b 4096 /dev/dasda1
    ▪ mke2fs -b 4096 /dev/dasdb1

• mkswap /dev/dasd?1 (2.2 kernels)
• mkswap /dev/dasd?1,2,3 (2.4 kernels)
## File systems vs. directories

<table>
<thead>
<tr>
<th>File System</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ (root)</td>
<td>Root directory</td>
</tr>
<tr>
<td>/bin</td>
<td>Binary files for executable programs</td>
</tr>
<tr>
<td>/boot *</td>
<td>Contains system startup files</td>
</tr>
<tr>
<td>/dev</td>
<td>Device files for hardware devices</td>
</tr>
<tr>
<td>/etc</td>
<td>Configuration files for the system</td>
</tr>
<tr>
<td>/home</td>
<td>Home directories for users</td>
</tr>
<tr>
<td>/lib</td>
<td>Library files for programs</td>
</tr>
<tr>
<td>/mnt</td>
<td>Mounted file systems</td>
</tr>
<tr>
<td>/opt</td>
<td>Optional files</td>
</tr>
<tr>
<td>/proc</td>
<td>Process control files</td>
</tr>
<tr>
<td>/root (not to be confused with / root)</td>
<td>Root directory</td>
</tr>
<tr>
<td>/sbin</td>
<td>System binary files</td>
</tr>
<tr>
<td>/tmp</td>
<td>Temporary files</td>
</tr>
<tr>
<td>/usr</td>
<td>User files for programs and libraries</td>
</tr>
<tr>
<td>/var</td>
<td>Variable files for programs and libraries</td>
</tr>
</tbody>
</table>
Backup and restore

- Native Linux facilities
  - afio/cpio/tar

- Software packages (Open Source and proprietary)
  - Amanda / offlindr
  - DFSMSdss / DDR
  - Tivoli TSM/ADSM
  - Innovation FDRINSTANT/UPSTREAM
  - CA BrightStor
  - SecureAgent SecureBackup
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.
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.
Software & maintenance management (3)

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

• Getting email notifications from security sources (CERT, etc.) is recommended.

• If you have to install 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`
  - `rpm -e package.name`
  - `rpm -q package.name`
  - `rpm -ql package.name`
  - `rpm -qlp package.name.rpm`
  - `rpm -qa`
  - `rpm -qf /path/to/file/name`

- **dpkg on Debian-based systems**
Data sharing with Linux/390 and other OS

- No direct, hard-wired sharing
- z/OS and z/VM don’t “know” ext2
- Linux doesn’t “know” VTOCS, etc.
  - Except now it does. But, no security!
- Various network-based methods
  - NFS
  - GFS
  - AFS
  - Samba (SMB / CIFS / MS Networking)
- Under VM - sharing minidisks *read-only* between guests.
Job scheduling

- Linux native facilities
  - cron
  - at
- Open Source:
  - DQS
  - queue
  - OpenPBS
  - generic NQS
- Proprietary
  - CA-7 Agent
  - Jobtrac
  - CA Scheduler
  - PBSPro
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 shadow password utilities.

• Use TCP Wrappers (/etc/hosts.allow, etc.)

• 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”).
- 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
- Webmin is popular
- linuxconf is/was popular
- useradd, userdel, usermod, groupadd, groupdel, groupmod are common
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/suauth to allow designated people to “su” using their own password.
• Consider using /etc/sudoers to grant some selected command authority to designated people.
Diagnostic information

- strace
- ulimit (to enable core dumps)
- gdb
- uptime
- top
- ksymoops
- netstat
- ping
- traceroute
- system logs
- dmesg
- standalone dump (2.4.x kernel only)
Editors (Holy War fodder)

- vi / vim / elvis
- emacs / xemacs
- joe
- jed
- jove
- ed (sed)

- nano
- pico
- ne
- Nedit
- THE (The Hessling Editor)

No native free 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 to visual mode.
Basic vi concepts (2)

- :set smd  or :set showmode
  - gives visual indicator 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
- Tcl
System logs

• Most of what you want will be in /var/log, or in a subdirectory of it.
• Names and contents vary by distribution
• 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`
Advanced commands

- ifconfig
- netstat
- route
- ping
- host / nslookup
- traceroute
- su
- sudo
- gzip / bzip2
- last

- chmod
- chown
- dmesg
- du
- df
- locate
- top
- sed
- head
- tail

- silo / zipl
- wget
- export
- file
- insmod
- modprobe
- lsmod
- rmmod
- telinit
Kernel modules, insmod and modprobe

• 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.
Kernel modules, insmod and modprobe (2)

• 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 has the right data in it.

• How does the module get it’s parameters?
  – provided on insmod command
  – read from /etc/modules.conf by modprobe and the kernel.
• How does the kernel find the module?
  – Usually in `/lib/modules/kernelver/something`...
  – `depmod -a` command creates a cross reference of the modules and where they are located, and writes it into `/lib/modules/kernelver/modules.dep`.
  – `modprobe` figures out the dependencies and loads the modules in the right order. `insmod` does not do this.
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.
Performance management

- **Some “standalone” native Linux tools:**
  - top, ntop, httperf, sar, iostat, gkrellm, pload, statnet, Big Brother, iptraf
- **Proprietary products**
  - Velocity Software ESALPS (under z/VM)
  - BMC MAINVIEW
  - Candle OMEGAMON XE
Creating additional images

- Some commercial tools
- Can be done with home grown tools
- For large “penguin farms,” knowing what to share between images and how is key.
- IBM Redbook “Linux on IBM zSeries and S/390: ISP/ASP Solutions,” SG24-6299
Additional information - web sites

- http://linuxvm.org/
  (Largely Linux/390 specific)
  (Linux/390 mailing list)
- 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
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
Additional information – Redpieces (2)

• Linux on zSeries and S/390: Cloning Linux Images in z/VM, redp0301
• 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
Additional information – Redpieces (3)

- Linux on zSeries and S/390: z/VM Configuration for WebSphere Deployments, redp3661
- 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
Additional information – Redpieces (4)

- 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 - Usenet (“news”)

- alt.os.linux.dial-up
- alt.os.linux.redhat
- alt.os.linux.slackware
- alt.os.linux.suse
- alt.os.linux.turbolinux
- comp.os.linux.admin
- comp.os.linux.advocacy
- comp.os.linux.announce
- comp.os.linux.development.apps
- comp.os.linux.development.system
- comp.os.linux.hardware
- comp.os.linux.help
- comp.os.linux.misc
- comp.os.linux.networking
- comp.os.linux.questions
- comp.os.linux.redhat
- comp.os.linux.security
- comp.os.linux.setup
- comp.os.linux.x
- comp.protocols.smb (Samba, mainly)
- linux.debian.devel.mentors
- linux.debian.devel.qa
- linux.debian.devel.release
- linux.debian.ports.s390
- linux.debian.project
- linux.debian.project
- linux.debian.security
- linux.dev.c-programming
- linux.dev.kernel
- linux.dev.laptop
- linux.dev.newbie
- linux.help
- linux.kernel
- linux.net.masquerade
- linux.redhat.announce
- linux.redhat.devel
- linux.redhat.install
- linux.redhat.pam
- linux.redhat.rpm
- linux.samba
- linux.sources.kernel
Additional information - O’Reilly books

  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
* 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 (3)

* 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 (4)

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

MVS
- VM
- Linux

**NOTE**
- LOGOFF
- PRINT
- ATTACH
- Q U
- Q N

**wall**
- /etc/motd
- exit
- lpr
- mount
- umount
- uptime
- users / w
- ps -ax
### 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>
</tbody>
</table>
Questions?