

Unix System Administration Handbook - 7.13.98

Notes typed and organized by Scot Corrie

1	Where to Start	12	12
2	Booting and Shutting down	48	36
3	Rootly Powers	55	7
4	The Filesystem	65	10
5	Controlling Processes	82	17
6	Adding New Users	95	13
7	Devices and Drivers	106	11
8	Serial Devices	132	26
9	Adding a Disk	166	34
10	Periodic Processes	178	12
11	Backups	198	20
12	Syslog and Log Files	216	18
13	Configuring the Kernel	239	23
14	TCP/IP and Routing	287	48
15	Network Hardware	312	25
16	The Domain Name System	366	54
17	The Network File System	384	18
18	Sharing System Files	403	19
19	SLIP and PPP	419	16
20	The Internet	438	19
21	Electronic Mail	517	79
22	Network Management	537	20
23	Security	559	22
24	Usenet News	579	20
25	Printing and Imaging	617	38
26	Disk Space Management	630	13
27	Hardware Maintenance	637	7
28	Accounting	652	15
29	Performance Analysis	671	19
30	UUCP	704	33
31	Daemons	721	17
32	Policy and Politics	750	29

1.9 **[Essential Tasks of the System Administrator]**

- Adding and removing users
- Adding and Removing Hardware
- Backups
- Installing New Software
- Monitoring the System
- Troubleshooting
- Maintaining Local Documentation
- Auditing Security
- Helping Users

2.15 **[Steps in the Boot Process]**

- Kernel Initialization
- Hardware Configuration
- System Processes
 - sched - 0
 - init - 1
 - Various memory handlers (except Solaris)
- Operator Intervention (Manual Boot Only) - Only the root partition is mounted, fsck must be run by hand
- Startup Scripts - The scripts are kept in `::/etc/init.d` with links made into the directories `/etc/rc#.d`
- Multi-User Operation - In order for login to be accepted, there must be a `::getty` process running. (Except Solaris). Init spawns the getty processes.

2.31 **[Startup Scripts]**

::/etc/inittab

The startup scripts are broken into small, manageable pieces that are invoked as init moves among run levels. The exact definition of each run level is governed by `::/etc/inittab`. If this file does not exist or is corrupted, only single-user operation is possible.

=> id:rstate:action:process

::sysinit

There are several special action categories. The most important is `::sysinit`, which identifies actions needed to bring the system to a state where init can query the operator for information.

2.33 **::lvinit** – Initializes the logical volume system, which allows you to spread filesystems across multiple disks.

2.37 **[:/etc/rc#.d]**

Contains all the scripts associated with run level #. These are symbolic links that point back to files in `/etc/init.d`. Some typical startup scripts from `/etc/init.d`

<u>Script</u>	<u>Function</u>
acct	Starts accounting
announce	Displays startup and shutdown announcements
audio	Enables audio functionality
autoconfig	Rebuilds the kernel to match current configuration
bsdldr	Starts the BSD-style line printer daemons
chkdev	Checks for the existence of devices on the system
configmsg	Notifies operator if config files need to be checked
cron	Start the cron daemon
filesystems	Runs fsck and mounts filesystems
fontserver	Starts the X Windows font server
lp	Starts the ATT-style line printer daemons
mail	Starts the sendmail daemon
netls	Starts a network licensing daemon
network	Configures network interfaces
quotas	Starts quotas and runs quotacheck
rmtmpfiles	Cleans out files from /tmp
savecore	Saves a copy of a kernel core dump from swap space
swap	Starts swapping on filesystems listed in <code>/etc/fstab</code>
syssetup	Starts the system logging daemon (syslogd)
usr	Mounts the /usr filesystem
uucp	Enables the UUCP system
xdm	Starts the X Windows display manager

Major Boot Scripts in Solaris

<code>/sbin/autopush</code>	Loads STREAMS modules	<code>/etc/iu.ap</code>
<code>/sbin/rcS</code>	Single-user startup	<code>/etc/rcS.d/*</code>
<code>/sbin/rc2</code>	Multi-user startup	<code>/etc/rc2.d/*</code>
<code>/sbin/rc3</code>	Exports network resources	<code>/etc/rc3.d/*</code>
<code>/sbin/rc0</code>	Brings the system down	<code>/etc/rc0.d/*</code>

2.40 **Boot-time configuration files in Solaris**

<code>/etc/defaultrouter</code>	Contains the address of the default IP gateway
<code>/etc/defaultdomain</code>	Contains the machine's NIS domain name
<code>/etc/nodename</code>	Contains the machine's primary hostname
<code>/etc/hostname.X</code>	Contains the hostname for interface X

HP-UX Startup Scripts

<code>/etc/bcheckrc</code>	Checks and mounts local filesystems
<code>/etc/recoverst</code>	Cleans up shared libraries
<code>/etc/brc</code>	Configures the console
<code>/etc/rc</code>	The main rc script, calls many other scripts
<code>/etc/netlinkrc</code>	Sets up networking
<code>/etc/netnfsrc</code>	Sets up NFS and NIS
<code>/etc/netbsdrc</code>	Sets up BSD services (named, gated, sendmail, etc.)
<code>/etc/vuerc</code>	Starts the Visual User Environment (X plus Motif)

2.42 **[System Won't boot]**

Some problems that can keep a UNIX system from booting are:

- Hardware Problems
- Defective Boot Blocks
- Damaged Filesystems – If the root partition cannot be read, the kernel won't load, and the system may

behave as if a hardware problem were at fault. If you suspect filesystem corruption, try to boot into single-user mode. If this is impossible, the system is toast.

- Improperly Configured Kernels
- Errors in Startup Scripts – Script errors are the most common impediment to booting.

2.45 [Rebooting and Shutting Down]

There are a number of ways to shutdown or reboot:

- Turning off the power
- Using the shutdown command
- Using the halt and reboot commands (BSD)
- Sending init a TERM signal
- Using telinit to change init's run level (ATT)
- Killing init

::shutdown

System	Pathname	Time	Reboot	Halt	Single
Solaris	/usr/sbin/shutdown	-gsecs	-i6	-i0	-iS
HP-UX	/etc/shutdown	secs	-r	-h	

::halt – also called shutdown –h

::reboot – also called shutdown –r

[kill init]

> sync

> kill –TERM 1

::telinit – Change init's run level

> telinit S

-q Makes init re-read inittab

4.59 [Standard directories and their contents]

Pathname	Contents
/	The root directory
/bin or /sbin	Commands needed for minimal system operability
/dev	Device entries for terminals, disks, modems, etc.
/etc	Critical startup and configuration files
/kernel	(Solaris) Files needed to load the kernel
/lib	Libraries for the C compiler
/proc	Images of all running processes (some newer systems)
/stand	Stand-alone utilities, disk formatters, etc.
/sys	(BSD) Kernel-building work are, configuration files
/tmp	Temporary files that disappear between reboot
/usr/5bin	(BSD) System V compatibility commands
/usr/bin	Executable files
/usr/etc	System maintenance commands
/usr/games	Games and diversions
/usr/include	Header files for C programs
/usr/lib	Support files for standard UNIX programs
/usr/local	Local software (stuff you install)
/usr/local/adm	Local accounting and log files
/usr/local/bin	Local executables
/usr/local/etc	Local system configuration files and commands
/usr/local/lib	Local support files
/usr/local/sbin	Local system maintenance commands
/usr/local/src	Source code for /usr/local/*
/usr/man	On-line manual pages
/usr/sbin	More system maintenance commands
/usr/ucb	Berkeley utilities and programs
/var/adm	Accounting files, records of resource usage
/var/spool	Spooling directories for printers, UUCP, mail, etc.
/var/tmp	Temporary space (files don't disappear after reboot)

(Where /sbin is present, /bin is usually a symbolic link to /usr/bin.)

4.59 [Eight types of files in UNIX]

- Regular files
- Directories
- Character and Block Device Files – see 4.60

Device files can be created with **::mknod**, and removed with **::rm**.

- UNIX Domain Sockets (BSD)
- Named Pipes
- Hard Links
Links are created with **::ln** and removed with **::rm**
- Symbolic Links
Symbolic links are created with **::ln -s** and removed with **::rm**

4.62 **[File Permissions]**

The SetUID and SetGID Bits

The Sticky Bit

The Permission Bits

Changing Permissions

4.64 Table 4.2 - Permission encoding for chmod

::umask – Assigning Default Permissions

4.65 Table 4.3 – Permission encoding for umask

[inode]

Table 4.4 – File-type encoding used by ls

5.68 **[Components of Process]**

5.74 Table 5.1 – UNIX Signals

5.75 **[Process states]**

State	Meaning
Runnable	The process can be executed.
Sleeping	The process is waiting for some resource.
Swapped	The process is not in memory.
Zombie	The process is trying to die.
Stopped	The process is suspended (not allowed to execute).

Processes generally get stopped in three situations:

- When a Control-Z (or the “suspend” character declared with stty) is typed to an interactive process
- At the specific request of a user or program
- When a process that has been put in the background tries to access its control terminal

5.77 **[nice and renice]**

5.80 Table 5.4 – Explanation of ps -ef output

5.81 Table 5.5 – Explanation of ps -elf output (shows more detail)

::top

5.82 **::nohup**

=> nohup *command* &

6.85 **[Adding Users]**

::adduser (on CD)

Required:

- Edit the /etc/passwd file
- Set an initial password
- Create the user’s home directory

For the User:

- Copy default startup files to the user’s home directory
- Setup the user’s mail home and establish mail aliases

For you:

- Add the user to the /etc/group file
- Record accounting information
- Enter the user in the site-wide user database
- Enter contact information in the local phone book
- Configure disk quotas
- Verify that the account is set up correctly

6.88 Sun suggests that you use the **useradd**, **usermod**, and **userdel** commands.

6.91 **[Startup files and their uses]**

Command	Filename	Typical uses
cs	.login	Sets the terminal type Sets up environment variables Sets biff and mesg switches
	.cshrc	Sets command aliases Sets path for command searches Sets umask value to control permissions Sets <i>cdpath</i> for filename searches Sets variables <i>prompt</i> , <i>history</i> , <i>savelist</i>
	.logout	Prints "to do" reminders Runs the fortune command Clears the screen
sh	.profile	Similar to .login and .cshrc for sh
vi	.exrc	Sets vi editor options
emacs	.emacs_pro	Sets emacs editor options Sets emacs key bindings
mail	.mailrc	(BSD) Defines personal mail aliases (BSD) Sets emacs key bindings
rn	.newrc	Specifies newsgroups of interest
xrdb	.Xdefaults	Specifies X11 configuration: fonts, color, etc.
startx	.xinitrc	Specifies initial X11 environment

6.94 **::edquota** – Use if you set quotas on disk usage.

7.98 **[Device Numbers and Jump Tables]** - ::Study

7.99 **[Driver Configuration Files]**

System	Device Table	Included Files	Man
Solaris	/usr/kernel/drv/*conf	/usr/kernel/drv/*	driver.conf, prtconf
HP-UX	/etc/master	/etc/conf/dfile	master, config, uxgen

7.105 **[Device Files]**

in **::/dev** (Solaris: **::/devices**, but links to **/dev** are automatically maintained.)

::mknod

=> *mknod filename type major minor*

[Naming Conventions for Devices]

Devices that have both block and character identities usually preface the character device name with the letter **r** for "raw" (Ex:/rdsk).

Serial devices are usually named **tty_L**

7.106 Table 7.3 – Sample naming conventions for disks and tapes

[Loadable Kernel Modules] - ::Study

7.107 In Solaris, virtually everything is a loadable module.

::modinfo

::add_drv – Add a Driver

::rem_drv – Unload a Driver

::modload

::modunload

8.109 **[Serial Devices]**

::/etc/remote – Configure to use a dial-out modem.

8.119 **[The Login Process]**

8.120 Table 8.5 – Terminal configuration files

::/etc/ttytype – Terminal type information is stored here.

=> *termtype device*

::/etc/gettytab – Associates symbolic names with port configuration information.

8.121 **::/etc/inittab**

=> *_ id:run-levels:action:process*

8.122 Table 8.6 – Possible values for the **/etc/inittab** file

8.123 **::/etc/gettydefs**

Solaris uses SAF.

::sacadm

::/etc/ttydefs – similar to **gettydefs**

8.124 **::/etc/terminfo** – Database of terminal capabilities

- 8.125 **::stty** – Directly change options.
- 8.126 Table 8.7 – Special characters for the terminal driver
- 8.126 **::tset** – Set options automatically.
- 9.139 **[Disk Installation Procedure]**
Overview of the process
 - Create device files – in /dev
 - Format the disk > format
 - Partition and label the disk
 - Create a UNIX filesystem > newfs
 - Check the integrity > fsck
 - Setup automatic mounting > vi /etc/fstab
 - Setup swapping on swap partitions
- 9.144 **[::/etc/fstab]** – Lists the device names and mount points of all the systems disks. The following are different:
Solaris = **::/etc/vfstab** (swap partitions and mount points are listed here)
HP-UX = **::/etc/checklist**
- 9.148 **[Specifics for Solaris Disk Installation]** – See example starting on pg. 148.
Entries for devices are in /etc/dsk and /etc/rdisk.
=> /dev/[r]dsk/cCtAd0sP
Options
C – Controller; A – SCSI target number; P – partition number
::drvconfig and **::disks** – After installing a disk, run either command to create device files for it.
::prtvtoc – Displays disk label.
- 9.149 **::newfs** or **::mkfs** can be used to install a filesystem
- 10.172 **::cron** – Daemon that handles periodic execution.
::crontab – Command used to submit files to \$/crontab directory to schedule cron jobs
::/var/cron/log or **::/usr/lib/cron/log** – location of cron logs, don't use cron logging.
- 10.173 **[Format of crontab files]** - ::Study
- 10.178 **[Specifics for crontab]**
Solaris
::/var/spool/cron/crontabs – location of crontab files.
::/etc/cron.d – location of config files such as cron.allow
::/etc/default/cron – CRONLOG=YES to turn on logging information to be written to /var/cron/log
HP-UX
::/usr/spool/cron/crontabs – location of crontab files
::/usr/lib – other config files
- 11.181 **[Backups]** - ::Study
- 12.204 Table 12.1 - **[Log Files]**
Don't touch /usr/adm/lastlog and /etc/utmp.
- 12.205 **[Specifics for Log Files]**
Solaris – (totally disorganized)
/var/log/*
/var/cron/log
/var/lp/logs/*
/var/saf/_log
/var/saf/zsmon/log
/var/adm/{messages,aculog,sulog,vold.log,wtmp}
/var/adm/log/asppp.log
HP-UX
/usr/adm
/usr/adm/nettl.LOG00 – is a network control and statistics file. Don't touch ~/diag logs (they are in binary form).
- 12.206 **::syslog** – System event logger (on CD) - ::Study
::/etc/syslog.conf – configuration file for syslog

13.218 [Subdirectories of /kernel in Solaris]

Subdir What it contains

drv	Loadable objects for device drivers and configuration files listing probe addresses for each device
exec	Loadable objects that understand object files (a.out, ELF, etc.)
fs	Loadable objects that manage filesystems (UFS,NFS,RFS, etc.)
misc	ELF loadable objects for miscellaneous kernel routines
sched	Directory containing process scheduling modules
strmod	Directory containing STREAMS modules
sys	Directory containing various optional kernel systems
unix	The base kernel itself

:/etc/system – use to configure the kernel

14.243 [TCP/IP] - ::Study

[IP] Internet Protocol. Transports raw data from one machine to another.

[ICMP] Internet Control Message Protocol. Provides several kinds of low-level support for IP, including error messages, routing assistance, and echo requests.

[ARP] Address Resolution Protocol. Translates logical network addresses to hardware addresses.

[UDP, TCP] User Datagram Protocol. Send data from one program to another using IP. UDP provides unverified transportation for individual messages, while TCP is reliable and connection-based.

14.244 [TCP/IP Network Model]

<i>Layer</i>	<i>Function</i>
Link Layer	Network hardware and device drivers
Network layer	Basic communication, addressing, and routing
Transport layer	Communication among programs on a net
Application layer	End-user application programs

14.262 **::ifconfig** – Used to enable or disable a network interface.

=> ifconfig *interface [family] address up option*

(In Solaris, you must run > ifconfig plumb before you can run > netstat -i)

14.265 **::route** – Configure static routes

14.266 **::routed** – The standard routing daemon

:/etc/gateways – Consulted by routed when it starts up.

14.267 **::gated** – A better routing daemon (on CD)

[Vendor-supported routing daemons]

System routed gated gated protocols

Solaris	Yes	No	-
HP-UX	No	Yes	HELLO, RIP, EGP, BGP

:/etc/gated.conf – configuration file

14.269 [Solaris, Configure a Network]

1. > vi /etc/hostname.le0
2. > vi /etc/netmasks
3. > vi /etc/defaultrouter

14.270 [HP-UX, Configure a Network]

1. vi /etc/netlinkrc
2. vi /etc/hosts (lan0, lan1, etc.)

::netstat – Displays various network-related status information

14.281 **::arp** – Inspect and Manage Address Mappings

[Find out an IP address]

> arp *hostname*

15.290 [Network Hardware, Ethernet] - ::Study

Variety of information on cables, networks, and hardware.

16.316 [::DNS] – Table 16.1 and more - ::Study

16.319 [::BIND] - ::Study

There are three components of the BIND system:

- A daemon called **named** that answers queries
- Library routines that resolve host queries using DNS
- Command-line interfaces to DNS: nslookup, dig, and host

[Name Server and Resolver]

In DNS parlance, a daemon like **named** (or the machine on which it runs) is called a “name server,” and the client code that contacts it is called a “resolver.”

::named – The BIND Name Server

named answers queries about hostnames and IP addresses. If named doesn't know the answer, it asks other

servers and caches their responses. **named** is also responsible for “zone transfers”.

Name servers are either: primary, secondary or caching-only. A machine can be a primary server for one domain, and a secondary server for other domains.

16.321 **::nslookup** – The shell interface to DNS. It can be used to query the database. (see Table 16.8 pg.357)

[How DNS Works]

gethostbyname – Programs call the gethostbyname routine to map hostnames to IP addresses. When a host is configured to use DNS, gethostbyname uses the resolver to query a name server for the address.

16.324 **[BIND Client Issues]** - ::Study

Table 16.3 – BIND installation and maintenance chores

[Configuring the resolver]

::/etc/resolv.conf

> vi /etc/resolv.conf

16.327 **[Configuration files]**

::/etc/exports – Controls NFS file sharing. (Solaris is **::/etc/dfs/dfstab**). Must contain the fully qualified names of the hosts allowed to mount a filesystem.

Users' **::~/rhosts** files need fully qualified names and short names.

sendmail must recognize its own fully qualified name as a local host.

[Setting up a Name Server]

The complete configuration for named consists of the boot file, the cache file, and, for primary server, the data file or files containing address mappings for each zone.

::/etc/named.boot – named starts at boot time and runs continuously.

16.328 **[Kill named]**

> kill -SIGNAL 'cat /etc/named.pid'

Note: named uses syslog, so **syslogd** should be started before named.

::/etc/named.boot – Specifies the role of a host relative to each zone, and the way it should get its copy of the resource records that make up the local part of the database.

::xfrnets - Edit this section to specify an access list of networks and hosts that are allowed to receive zone transfers of your entire database.

16.331 **[The DNS Database]** - ::Study

Items stored in the database are called **resource records**.

=> [name] [ttl] [class] type data

16.354 Table 16.6 – Syslog messages from named

16.361 **[Solaris, Specifics]**

Solaris uses a service order file called: **::/etc/nsswitch.conf** to specify how BIND, NIS, NIS+ and the /etc/hosts file interact.

16.362 **[BIND files in Solaris]**

File	Directory	Description
resolv.conf	/etc	Resolver library configuration
in.named	/etc	Name server daemon
named-xfer	/usr/sbin	Zone transfer code
named.boot	/etc	Boot files for name servers
named.pid	/etc	Process ID
named.run	/var/tmp	Output from debug mode
named.stats	/var/tmp	Statistics output
named_dump.db	/var/tmp	Dump of entire database

[HP-UX]

There is no service order file. DNS, if configured, is tried first. If that fails, NIS is tried, and if that fails or is not configured, then /etc/hosts is tried.

[BIND files in HP-UX]

File	Directory	Description
resolv.conf	/etc	Resolver library configuration
in.named	/etc	Name server daemon
named-xfer	/etc	Zone transfer code
named.boot	/etc	Boot files for name servers
named.pid	/etc	Process ID
named.run	/usr/tmp	Output from debug mode
named.stats	/usr/tmp	Statistics output
named_dump.db	/usr/tmp	Dump of entire database
Zone files	/etc/newconfig	Default location for zone files

(See **::/etc/newconfig/bind** for some sample files.)

17.369 **[NFS, Network File System]**
::mountd – Daemon that runs on the server to handle mount requests (also called `rpc.mountd`). It reads the `/etc/exports` (Solaris = `/etc/dfs/dfstab`) file to see which hosts should be allowed access and what restrictions. Solaris uses the **share** command to export filesystems instead of the `/etc/exports` file.

17.370 Table 17.1 – Common export attributes

[What to do after editing /etc/exports]

Solaris – Not applicable (see below)

HP-UX - > `/usr/etc/exportfs -a`

17.371 **::/etc/dfs/dfstab** – This file is actually a shell script that executes the **share** command once for each exported filesystem.

Table 17.4 – Options for the share command

::nfsd – Once `mountd` validates a client's request, `nfsd` handles requests for various filesystem operations.

These requests are handled on the server side by `nfsd`, the *NFS operations daemon*. `nfsd` takes one argument specifying the number of copies of itself to fork. Run at least four `nfsds`. It is tricky to configure, but closely tied to the maximum hardware contexts supported by your computer's CPU. Formula $N - 2$. Where N is the number of contexts.

Run `> netstat -s`. Determine how many UDP sockets overflow. Add `nfsds` until UDP socket overflows drop to 0, or you reach the maximum permitted level.

Maximum number of nfsds to use on a busy server

Solaris SPARC 10

SuperSPARC 12

HP-UX PA-RISC 8

17.373 **::biode** – run this daemon on all clients

17.374 Table 17.6 – NFS mount flags

17.377 **::automount** – Sun's Automounter - ::Study
`automount` understands three different kinds of configuration files (called maps):
 direct maps, indirect maps, and master maps

17.378 **::amd** – A better automounter (on CD) - ::Study

17.383 **::nfsstat** – Displays various statistics kept by the NFS system. Run this regularly.

Options

-s servers

-c clients

18.386 **[System files that are commonly shared]**

Filename	Function
<code>/etc/passwd</code>	User account information database
<code>/etc/group</code>	UNIX group definitions
<code>/etc/hosts</code>	Maps between hostnames and IP addresses
<code>/etc/networks</code>	Associates text names with IP network numbers
<code>/etc/services</code>	Lists port numbers for well-known network services
<code>/etc/protocols</code>	Maps text names to protocol numbers
<code>/etc/ethers</code>	Maps between hostnames and Ethernet addresses
<code>/etc/aliases</code>	Electronic mail aliases
<code>/etc/rpc</code>	Lists ID numbers for RPC services
<code>/etc/netgroup</code>	Defines collections of hosts, users, and networks

Most server-based systems are setup to work with only these and perhaps a few additional files.

18.387 **::rdist** – In most cases, this command is the best way to distribute files from a central server. Similar to `make`.

18.390 **::expect** – Pulls files

18.392 **::NIS** – The Network Information Service

18.400 **::NIS+**

19.406 **[SLIP and PPP]** - ::Study as needed

20.429 **[Setup an Anonymous ftp Server]**

21.439 **[Electronic Mail]** - ::Study

22.530 **[SNMP]** - ::Study

See also the tables on pgs. 532-533

23.541 **[Seven Rules of Security]**

23.542 **[Check for null passwords]**

`> awk -F: '{ if ($2 == "") print $1 }' /etc/passwd`

23.545 **[SetUID Programs]**

`> /usr/bin/find / -user root -perm -4000 -print | /usr/ucb/mail -s "Setuid root files" corries`

23.547 **[File Permissions]**

> chmod 770 /dev/kmem
> chmod 644 /etc/passwd; /etc/group

23.548 **[Restricting root login by TTY]**

Solaris /etc/default/login

::/etc/hosts.equiv and ~/.rhosts – Same function. They allow users to login and copy files between machines without typing their passwords. Safest to create these for each user as a zero-length file.

23.550 **[Security Power Tools]**

::cops – Warns of security problems. (on CD)

::crack – (on CD)

::tcpd – logs connections to TCP services such as telnet, rlogin, and finger (on CD).

::tripwire – Warns of file alterations.

::crypt

25.600 **[Printing Commands]**

Command Function

accept	Starts accepting jobs for a device
cancel	Cancels a queued or printing job
disable	Disables printing to a device
enable	Enables printing to a device
lp	Queues jobs for printing
lpadmin	Configures the printing system
lpmove	Moves jobs from one device to another
lpsched	Printer scheduling daemon
lpshut	Disables lpsched
lpstat	Shows the status of the system
reject	Stops accepting jobs for a device

::/var/spool/lp/request/dest – Usually the spool directory location

::/usr/spool/lp/log – Log files

25.602 **::lpadmin** – Configure a printing environment, add a printer, etc.

26.620 **::quot** – Information about disk usage for each user on each filesystem.

::du – Summarizes disk usage within a directory.

26.623 **::tunefs** – Set Filesystem Parameters

26.624 **[Disk Quotas]** - **::Study**

28.643 **::/var/adm/wtmp** – Connect-time accounting is turned on by the existence of this file. Login, port, and login/logout times are recorded.

::ac – summarizes the data by person in the wtmp file.

> ac -p

28.647 **[Setting up Accounting]**

28.650 **[What Accounting Does]**

28.652 Table 28.3 – Summary of ATT accounting files in /usr/lib/acct

28.653 **[File and command locations by vendor]**

File	Solaris	HP-UX
ATT commands	/usr/lib/acct	/usr/lib/acct
ATT reports	/var/adm/acct	/usr/adm/acct
acct or pacct	/var/adm	/usr/adm
wtmp	/var/adm	/etc

Table 28.5 – Solaris Bonus commands

29.658 **::sar** – Analyzes CPU utilization

Ex: > sar -u 5 5

::vmstat – Same as sar, but more verbose.

29.666 **::iostat** - Analyze disk I/O

30.674 **::uux; ::uucp; ::uucico** – UUCP commands.

30.680 Table 30.1 – Files in the /usr/lib/uucp directory

31.708 **[Configuring inetd]** – Configuration file is usually in **::/etc/inetd.conf**

31.712 **[ATT Daemons]**

::snmpd – Network Management Server

::rnpdaemon – HP-UX. Print from BSD to HP-UX

::glbd and **::llbd** – Location Brokers

NFS Daemons

::nfsd – Serve files

::rpc.mountd – Respond to Mount Requests **{inetd}**

::rpc.lockd and **::rpc.statd** – Team that manages NFS locks

::bioid – Cache Blocks

NIS Daemons

::ypbind – Locate NIS servers. Runs on all NIS clients and servers.

::ypserv – NIS server. Runs on all NIS servers

::ypxfrd – Transfer NIS Databases

::rpc.nisd – NIS+ Server. Runs on all NIS+ servers.

Internet Daemons

::comstat – Notify users of new mail. **{inetd}**

::talkd – Talk Server **{inetd}**

::rwhod – Maintain Remote User List

::ftpd – File Transfer **Server {inetd}**

::popper – Mailbox Server **{inetd}**

::rlogind – Remote Login Server **{inetd}**

::telnetd – Telnet Remote Login Server **{inetd}**

::rshd – Remote Command Execution Server **{inetd}**

::rexecd – Command Execution Server **{inetd}**

::rpc.rexd – Command Execution Server **{inetd}**

::timed – Synchronize Clocks

::xntpd – Synchronize Clocks even better (on CD)

::routed – Maintain Routing Tables

::gated – Maintain Complicated Routing Tables

::named – DNS Server

::nntpd – Transfer Usenet News

::syslogd – Process Log Messages

::fingerd – Lookup Users **{inetd}**

::httpd – WWW Server **{inetd}**

HP 6294S Network Administration

2.1 [Chapter 2 - Configuring Network Connectivity]

Slide: The Network Map

[Network Map]

Keep a map for all of the LAN hardware that you administrate. With the following information:

- approximate dimensions of the building or room containing the LAN.
 - Type of the cables, including terminators.
 - Locations of repeaters, hubs, bridges, routes, and gateways
 - location of nodes and, for each node, the following:
 - hostname
 - Internet address
 - link level address
 - hardware address of the network interface
 - operating system information
- (Create a spreadsheet)

2.17 **::ifconfig**

=> ifconfig interface addr_family [address] [parameters]

interface A string of up to four alphabetic characters followed by one digit, such as lan0. The alphabetic characters represent the network interface. The integer denotes the card instance. For all HP 9000 workstations and servers, card instances are assigned to each LAN card automatically. The first configured LAN card will be lan0, lo0 specifies the local loopback interface.

addr_family The only supported address family is inet (default) for the DARPA Internet protocol family).

address The Internet address expressed in "dot" notation.

parameters The most important parameters are up, down, arp, -arp and netmask.

up - activates the interface

down - shuts down the interface

[-] arp - disables/enables the address resolution protocol to map between OSI levels 2 and 3 (link level and network level).

netmask subnet - The subnet bit mask for your subnet. It specifies how much of the address to reserve for subdividing the networks into sub networks.

[Check the configuration of the LAN interface after reboot with]

> ifconfig lan0

::lanconfig

The lanconfig command is used to configure the encapsulation mode for the network interface protocol.

=> lanconfig interface [ether | -ether] [ieee | -ieee]

[-] ieee Enables/disables the IEEE 802.3 protocol over the network interface

[-] ether Enables/disables the Ethernet protocol over the network interface

[Check connectivity by using the ping command after reboot.]

> ping remote_ip -n 10

2.28 **[::/etc/networks]** - Maps network addresses to network names.

=> network_name network_ip_address [alias ...]

Note: If you have configured this file, the commands netstat -l and netstat -r show the network name instead of the network IP address.

2.32 Slide - **[Routing Tables]**

A routing table typically contains pairs of addresses. Each pair represents:

- a destination network address
- the IP address of the router to which to send packets destined for that network

Table-driven IP routing is considered address-only routing. This means that the packet traveling across the internet takes along only the destination address. Along the route, the packet is "told where to turn" by routers.

The router listed in the routing tables must lie on a network to which the particular machine connects directly.

Use destination network addresses instead of destination host addresses to keep the tables small.

2.37 **::route**

-f Flushes the routing table of all gateway entries

-n Any host and network addresses are printed in the Internet dot notation, except for the default network address which is printed as "default".

add Adds the specified host or network to the routing table

delete	Deletes the specified host or network from the routing table
net	Specifies that the specified destination is a network. Use net when <i>dest</i> is a network address, for example 192.45.3.
host	Specifies that the destination is a host
<i>dest</i>	Specifies the IP address of the host or network where packets will be routed. <i>dest</i> may be either a host name (or alias as listed in /etc/hosts), a network name (or alias as listed in /etc/networks), an Internet address in "dot" notation, or the keyword default. If default is specified, the default gateway entry is changed to gateway. gateway is where packets are routed if they match no other destinations in the route table.
<i>gateway</i>	Specifies the gateway node through which <i>dest</i> is reached. It must be specified as an Internet address in "dot" notation or as a host name. The system's own IP address can be used when the remote router supports proxy ARP routing.
<i>count</i>	Integer (also called "hops") that indicates whether the gateway is the local or a remote host. If <i>count</i> is greater than 0, the gateway is a remote host. If <i>count</i> equals 0, the gateway is the local host. The default is 0.

2.38 Slide - Configuring Your system as a router

Add entries for ifconfig lan1 .. and lanconfig lan1 .. to /etc/rc.config.d/netconf

```
> INTERFACE_NAME [1] = lan1
> IP_ADDRESS [1] = 192.10.3.2
> LANCONFIG_ARGS [1] = "ether"
```

Modify /etc/hosts

```
192.6.5.11          leo    leo0    #lan0    net      192.6.5
192.10.3.2         leo    leo1    #lan1    net      192.10.3
```

Setup routing (if needed).

Check your kernel for the LAN 9000 drivers.

Halt your system, add the new LAN interface, and reboot.

Verify your configuration.

The new LAN card has to be initialized with separate ifconfig and lanconfig commands. Typical entries in /etc/rc.config.d/netconf are shown below:

```
INTERFACE_NAME [1] = lan1
IP_ADDRESS [1] = 192.10.3.2
LANCONFIG_ARGS [1] = "ether"
```

Next, modify the /etc/hosts file. Add an entry for the new LAN interface. Because the entries for lan0 and lan1 will have the same host name, you should add a host name alias to both entries.

```
192.6.5.11          leo    leo0    #lan0
192.10.3.2         leo    leo1    #lan1
```

Your system will belong to two networks:

- lan0 to network 192.6.5
- lan1 to network 192.10.3

For these two networks you don't need any additional route command. A route command is needed only if a third network is reachable through another router.

If you have an HP 9000 server with Precision Bus, the standard LAN 9000 interface is part of the personalit card and needs the driver lan3. The new LAN interface is added to a free Precision Bus slot and needs the driver lan1. Check your kernel and add the driver if it is not included.

Halt your system to install the new LAN interface card. Boot the system, and check the startup messages.

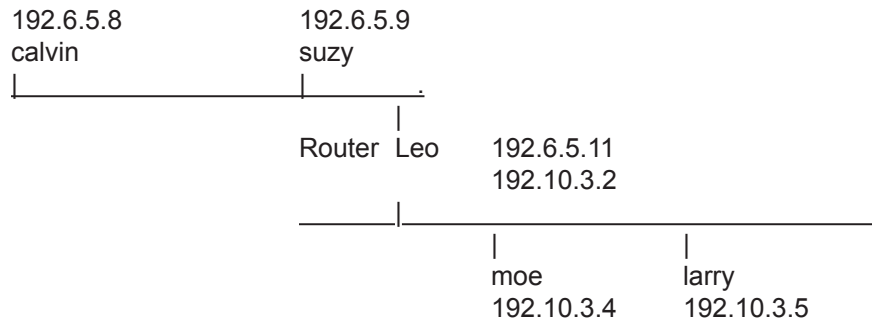
Perform several checks to verify your new network configuration:

- Check the new LAN device files with ll, lssfs, and ioscan -fn -H hardware_path.
 - /dev/lan1
 - /dev/diag/lan1
 - /dev/ether1
- Check the IP configuration of lan1
 - > ifconfig lan1
- Check the routing table
 - > netstat -rn
- Check you connectivity to a system in this second network:
 - > ping leo1
 - > ping 192.10.3.5

2.40 [Use SAM to configure a New LAN card]

Sam > Networking and Communications > Network Interface Cards

2.41 Slide - Using your system as a router



```

> /etc/hosts (on all systems)
192.6.5.8    calvin
192.6.5.9    suzy
192.6.5.11  leo    leo0
192.10.3.2  leo    leo1
192.10.3.4  moe
192.10.3.5  larry
  
```

```

> /etc/rc.config.d/netconf (on moe and larry)
ROUTE_DESTINATION [0] = default
ROUTE_GATEWAY [0] = 192.10.3.2
ROUTE_COUNT [0] = 1
  
```

System leo is now a router. Because leos executes two ifconfig and lanconfig commands (one for each LAN card), leos's LAN driver is smart enough to make each of the cards talk to each other. Therefore, for local routing (from 192.6.5 to 192.10.3 or vice versa), you do not need any route statements on leo.

If you wanted leo to route to a network connected to a different router (remotely, this is not shown in the slide), you would need to add the proper route statements; however, in order for systems on network 192.6.5 (calvin and suzy) to talk to nodes on 192.10.3 (moe and larry), you must add route entries to the /etc/rc.config.d/netconf file on each node. For example, the nodes on the network 192.6.5 (calvin and suzy) should have a default route entry of:

```

> ROUTE_DESTINATION [0] = default
> ROUTE_GATEWAY [0] = 192.6.5.11
> ROUTE_COUNT [0] = 1
  
```

The following route command will be executed:

```
> route add default 192.6.5.11 1
```

The nodes on the network 192.10.3 (moe and larry) should have the following default route entry:

```

> ROUTE_DESTINATION [0] = default
> ROUTE_GATEWAY [0] = 192.10.3.2
> ROUTE_COUNT [0] = 1
  
```

The following route command will be executed:

```
> route add default 192.10.3.2 1
```

(Warning) Routes are reciprocal, that is, both sides of the network must have the proper route statements. If calvin has the proper route statement but moe does not, then calvin can talk to moe but moe cannot answer. This example assumes you have only one gateway. You can use default route statements if you have only one gateway. If you have more than one gateway, all but the last route statements need to specify exactly to which gateway to route.

Proxy ARP routing may be used instead of the routes described in the examples in this topic, if the router supports this routing feature. In this case, the routing of calvin, suzy, moe, and larry can be configured as follows:

```

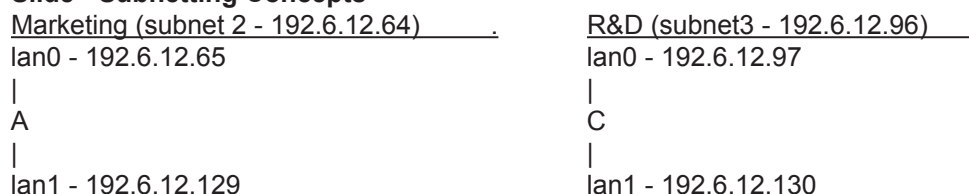
> ROUTE_DESTINATION [0] = default
> ROUTE_GATEWAY [0] = $ { IP_ADDRESS [0] } # <--- Own IP address
> ROUTE_COUNT [0] = # <--- Local flag
  
```

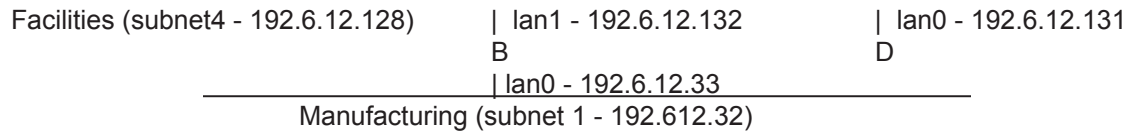
This reduces the work in administrating the routing commands, but an HP 9000 system used as router does not support proxy ARP routing.

Use the netstat -rn to look at the entries in your routing table.

2.43

Slide - Subnetting Concepts





Subnetting is an optional addressing scheme that allows you to partition the host address portion of an Internet address into discrete subnetworks. By doing this, several physical networks share the same network address to form one logical network.

Subnets are useful when you only have one network address, but you must divide your network into subnets because:

- You have reached the limit of a single LAN technology
- You want to isolate select nodes to their own subnet due to congestion
- You want to change your connectivity but don't want other network to modify their configuration to reach you.

Subnet addressing is implemented by specifying the keyword `netmask` and designating a 32-bit subnet mask in the `ifconfig` command when a LAN interface card is powered up with its Internet address.

The subnet mask is ANDed with the address attached to a message coming across the network to determine if that message should be routed to a node on the local network or ignored.

Ex:

```

IP address      192.6.12.65          11000000.00000110.00001100.01000001
Netmask        255.255.255.224 AND 11111111.11111111.11111111.11100000
LAN interface
to subnet      192.6.12.64          11000000.00000110.00001100.01000000

```

All nodes on a network (with a given network address) must specify the same subnet mask.

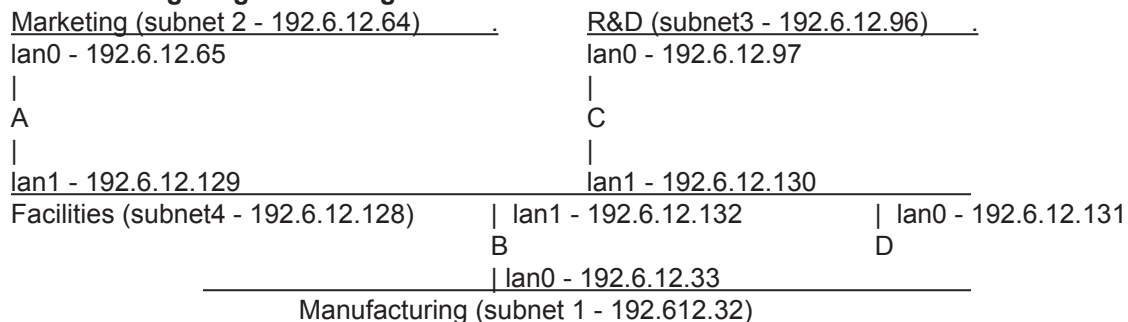
Explanation of the Example on the Slide

The example on the slide is a Class C network. The network address is 192.6.12, which leaves the last field to be used as the host address. Thus, the host address space is eight bits. A portion of these eight bits must be used to define the subnet. The example shows four subnets. Because there are four subnets, it takes three bits of the eight-bit host address space to specify the subnets. Note that subnet 0 is not allowed. Therefore, you must set the three most significant bits of the eight-bit space. For example, the subnet mask for the facilities subnet (4) would be 255.255.255.224, while the subnetwork address for subnet (4) would be 192.6.12.128. 128.64.32.16.8.4.2.1

Because the three most significant bits of the host part are used to specify the subnets there are five bits remaining to specify the host address. The following are the valid host addresses subnet addresses, and broadcast addresses for all subnets:

Subnet	Subnet Address	Hosts IP Addresses	Broadcast Address
Not allowed *	192.6.12.0 *	192.6.12.1-30	192.6.12.31
Manufacturing	192.6.12.32	192.6.12.33-62	192.6.12.63
Marketing	192.6.12.64	192.6.12.65-94	192.6.12.95
R & D	192.6.12.96	192.6.12.97-126	192.6.12.127
Facilities	192.6.12.128	192.6.12.129-158	192.6.12.159
For Future	192.6.12.160	192.6.12.161-190	192.6.12.191
For Future	192.6.12.192	192.6.12.193-222	192.6.12.223
Not Allowed *	192.6.12.224	192.6.12.225-254	192.6.12.255 *

2.46 **Slide - Configuring Subnetting**



ifconfig for System A:

```

ifconfig lan0 192.6.12.65 netmask 255.255.255.224 up
ifconfig lan1 192.6.12.129 netmask 255.255.255.224 up

```

route command for System A:

```

route add net 192.6.12.32 192.6.12.132 1

```

```
route add net 192.6.12.96      192.6.12.130 1
To configure the ifconfig and route command add entries to /etc/rc.config.d/netconf
```

The system administrator must assign the host addresses for each system on the subnet to fall within the ranges described in the previous topic. An ifconfig statement must be executed for each interface in each system with the same subnetmask.

The subnetmask is also configured in /etc/rc.config.d/netconf

Ex:

```
INTERFACE_NAME [0] = lan0
...
...
INTERFACE_NAME [1] = lan1
IP_ADDRESS [1] = 192.6.12.129
SUBNET_MASK [1] = 255.255.255.224
LANCONFIG_ARGS [1] = "ether"
```

2.47 **Routing Examples**

The route command must be used to set up routes between the subnets. For all this you have to configure the file /etc/rc.config.d/netconf.

The route commands and the entries in /etc/rc.config.d/netconf are shown below:

```
ROUTE_DESTINATION [0] = "net 192.6.12.32"
ROUTE_GATEWAY [0] = 192.6.12.132
ROUTE_COUNT [0] = 1
```

```
route add net 192.6.12.32 192.6.12.132 1
```

```
ROUTE_DESTINATION [1] = "net 192.6.12.96"
ROUTE_GATEWAY [1] = 192.6.12.130
ROUTE_COUNT [1] = 1
```

Note: For all systems in a subnet that have only one LAN interface card (not routers), proxy ARP routing may be used if the router supports proxy ARP routing. This reduces much of the work in administering the routing commands. However, an HP-UX 9000 system with more than one LAN interface, which works as a router, does not support proxy ARP routing.

2.49 **Slide - Troubleshooting Network Connectivity**

[Build a Router]

```
> vi /etc/rc.config.d/netconf
interface_name      [0] = lan0
ip_address          [0] = "156.153.198.161" (change)
subnet_mask         [0] = "255.255.254.0" (change)
broadcast_address   [0] = ""
lanconfig_args      [0] = "ether"
dhcp_enable         [0] = 0
(copy the above six lines. Increment the [0] by one, for each new interface.)
```

```
hostname = "new name" (change if needed)
route_gateway = "192.7.11.161" (change)
```

```
> vi /etc/hosts (to add an alias)
192.7.11.161      na157s4      router1
156.153.198.161  na157s4      router2
> reboot (to stop and restart networking)
or
> /sbin/init.d/net stop
> /sbin/init.d/net start
> ping 192.7.11.0 (to test)
> netstat -r
```