

Version 3

Provide feedback to: cookbook@apollo.hp.com

Table of Contents

[Copyright and Legal Notices](#)

[Detailed Table of Contents \(Outline\)](#)

[1. System Start-up And Shutdown](#)

[2. Swap Space Configuration](#)

[3. Start-up Configuration Files](#)

[4. Devices](#)

[5. Operating System Installation](#)

[6. Layered Software Installation](#)

[7. TCP/IP Communications](#)

[8. NIS and NIS Plus](#)

[9. NFS and the Automounter](#)

[10. Domain Name System \(DNS\)](#)

[11. Electronic Mail](#)

[12. Printing](#)

[13. Common Login Environments](#)

[Hyperlinked Index](#)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Version 3.0

[\[To: Table of Contents\]](#)

Copyright 2000 by Hewlett-Packard Co. All rights reserved.

The information in this document is subject to change without notice. Hewlett-Packard makes no warranty of any kind with regard to this material including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages with the furnishing, performance or use of this material. Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains proprietary information that is protected by copyright.

UNIX is a registered trademark in the U.S. and other countries, licensed exclusively through X/Open Co, Ltd.

All other trademarks and registered trademarks are the property of their respective holders.

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Detailed Table of Contents

1. System Start-up And Shutdown

1.1 Solaris Start-up

1.2 Solaris Shutdown

1.3 HP-UX 10.x and HP-UX 11.x Start-up

1.4 HP-UX 10.x and HP-UX 11.x Shutdown

1.5 HP-UX Series 800 Firmware Start-up

1.6 HP-UX Series 700 Firmware Start-up

2. Swap Space Configuration

2.1 Swap Space on Solaris

2.1.1 Adding a Raw Swap Partition

2.1.2 Adding File System Swap

2.1.3 Disabling Swap Space

2.2 Swap Space on HP-UX 10.x and HP-UX 11.x

2.2.1 Adding a Raw Swap Partition

2.2.2 Adding File System Swap

2.2.3 Disabling Swap Space

3. Start-up Configuration Files

3.1 Solaris

3.2 HP-UX 10.x and HP-UX 11.x

3.3 Customizing the HP-UX Start-up Process

4. Devices

4.1 Naming Conventions

4.2 Configuring Devices At Boot Time

4.2.1 Configuring Devices on Solaris

4.2.2 Configuring Devices on HP-UX 10.x and HP-UX 11.x

4.3 Interoperability Issues

4.3.1 HP-UX and CD-ROMS

4.3.1.1 High Sierra

4.3.1.2 ISO 9660

4.3.1.3 RockRidge

4.3.2 General Tips for Adding SCSI Devices

4.4 Adding a Disk to Solaris

4.5 Adding an SCSI Disk To HP-UX 10.x and HP-UX 11.x

5. Operating System Installation

5.1 Installing Solaris

5.1.1 Solaris

5.1.2 Solaris 2.4

5.1.3 Alternative Installation Methods

5.1.3.1 Installing Via Disk Duplication

5.1.3.2 Installing Via JumpStart

5.2 Installing HP-UX 10.x

5.3 Installing HP-UX 11.x

6. Layered Software Installation

6.1 Installation on Solaris

6.1.1 Installing From the Command Line

6.1.2 Installing Using the swmtool GUI

6.3 Installation on HP-UX 10.x and HP-UX 11.x

6.3.1 Software Distributor HP-UX

6.3.2 Installing Software With swinstall

7. TCP/IP Communications

7.0.1 Interface Names

7.1 Solaris

7.1.1 Solaris and Routing

7.1.1.1 Solaris and Dynamic Routing

7.1.1.2 Solaris and Static Routing

7.2 HP-UX 10.x and HP-UX 11.x

7.3 Using SAM

7.4 Troubleshooting Hostname Changes

8. NIS and NIS Plus

8.1 Supporting NIS Clients From NIS+ on Solaris

8.2 Configuring an HP-UX 10.x and HP-UX 11.x System as an NIS Client

8.3 Configuring an HP-UX 10.x and HP-UX 11.x System as an NIS Slave Server

8.4 Configuring an HP-UX 10.x and HP-UX 11.x System as an NIS Master

8.4.1 Updating NIS Maps on and HP-UX 11.x HP-UX 10.x

8.5 Supporting NIS Clients From NIS+ on HP-UX 11.x

9. NFS and the Automounter

9.1 Configuring NFS

9.1.1 Solaris

9.1.2 HP-UX 10.x

9.1.3 HP-UX 11.x

9.2 The Automounter

9.2.1 Solaris

9.2.2 HP-UX 10.x

9.2.3 HP-UX 11.x

9.3 Troubleshooting Hints and Tips

9.3.1 Troubleshooting NFS

9.3.2 Troubleshooting the Automounter

10. Domain Name System (DNS)

10.1 Solaris Implementation of Name Resolution

10.1.1 Testing the Modifications

10.2 HP-UX 10.x and HP-UX 11 Implementation of Name Resolution

10.2.1 Configuring The Resolver

10.2.2 Testing DNS

11. Electronic Mail

11.1 Configuring Sendmail on Solaris

11.2 HP-UX 10.x and HP-UX 11.x

11.3 Testing the Sendmail Configuration

11.4 Sendmail Version 8

12. Printing

12.1 Printing Command Differences

12.2 Solaris

12.2.1 Adding a Local Printer With admintool

12.2.2 Adding a Local Printer via the Command Line

12.2.3 Adding a Remote Printer With admintool

12.2.4 Adding a Remote Printer via the Command Line

12.3 HP-UX 10.x and HP-UX 11.x

12.3.1 Adding a Local Printer With SAM

12.3.2 Adding a Remote Printer With SAM

12.3.3 Adding a Printer Manually

12.3.4 Printer Command Language (PCL)

13. Common Login Environments

13.1 Shells

13.2 Login Process

13.3 Concepts and Strategies

13.3.1 Understanding Login Scripts

13.4 Developing Common Login Scripts

13.4.1 Identifying the System Type

13.4.2 Defining the Search Paths

13.4.3 Providing Common Aliases

13.4.4 Setting the Correct Terminal Type

13.4.4.1 Terminal Emulation

13.4.4.2 Key Mapping

13.5 Miscellaneous Settings

13.5.1 umask Settings

13.5.2 Autologout and Tmout

13.5.3 Destructive Backspace

13.6 CDE Login Environment

13.7 Sample Script

Hyperlinked Index

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task

Chapter 1 System Start-up And Shutdown

[\[To: Table of Contents\]](#)

This chapter describes the start-up and shutdown procedures under Solaris, and HP-UX versions 10.x and 11.x.

Your first interaction with a booting system will be with the system firmware which is the software that controls the hardware. The firmware is more dependent upon the hardware involved than upon the operating system. The firmware allows modification of boot parameters such as those required to bypass the system start-up scripts. (This bypass procedure is covered later in this manual.)

1.1 Solaris Start-up

These steps describe the boot sequence for Solaris as shown in Figure 1-1.

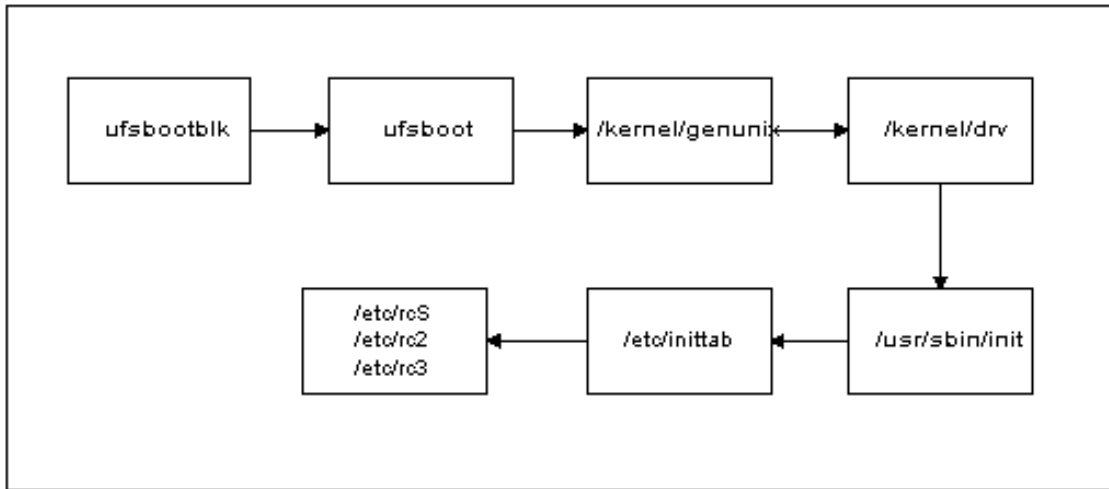


Figure 1-1: Solaris Boot Sequence

1. Power up the system. At this time, the RAM self-test will be performed.
2. The PROM loads ufsbootblk from the boot disk. ufsbootblk opens the boot device and loads ufsboot. (The monitor prompt will be either ">" or "ok," depending on the version of the PROM monitor running on the system.)
3. ufsboot loads the kernel from /kernel/genunix.
4. The kernel loads drivers from the /kernel/drv directory.
5. /usr/sbin/init creates processes to set up the system based on /etc/inittab. The /etc/inittab file specifies which processes are started at each run level. (For more information on init and /etc/inittab, refer to Chapter 3, Start-up Configuration Files.)

NOTE: Stop-a can be used to abort a bootup sequence. Press **Stop-A** (that is, press the **Stop** key and the **"a"** key simultaneously). At the ">" or "ok" prompt, type **b -s** to boot in single-user mode. Or, at the "OK" prompt, type **boot -s** to boot in single-user mode.

1.2 Solaris Shutdown

Shutdown the system using one of the following commands:

`/usr/sbin/shutdown` - shuts down the system or changes system states. shutdown executes the K-prefix files in `/etc/rc<N>.d`, where `<N>` is the new run level. (For more information on these scripts, refer to Chapter 3, Start-up Configuration Files.)

`/usr/sbin/init <run level>` - executing init followed by the new run level causes the system to execute at the new run level.

`telinit <run level>` - The `telinit` command is a symbolic link to the `init` command. This command is functionally equivalent to executing `init` followed by the new run level.

NOTE: You should avoid running `/usr/ucb/shutdown` because the kill files will not be executed and you may not get a clean shutdown.

1.3 HP-UX 10.x and HP-UX 11.x Start-up

These steps describe the boot sequence for HP-UX versions 10.x and 11.x.

NOTE: The firmware is the first step in booting. For more details, or to bypass the normal boot process, see Section 1.5 for Series 800 systems or Section 1.6 for Series 700.

1. Power up the system.
2. The firmware loads the booter which loads the Initial System Loader (ISL). The ISL loads the kernel `/stand/vmunix`.
3. The kernel locates the root file system and starts the `init` process as process ID 1.
4. `/sbin/bcheckrc` and `/sbin/rc` which start system daemons defined in the `/sbin/init.d` directory. The HP-UX 10.x and HP-UX 11.x boot processes are very similar to that for Solaris, as both use `init` and `inittab` as well as `kill/start` files for controlling the boot process. For more information, refer to Chapter 3, Start-up Configuration Files.
5. `init` switches to the default run-level specified in `/etc/inittab`, and starts all daemons appropriate to that level. For more information, refer to Chapter 3, Start-up Configuration Files.

1.4 HP-UX 10.x and HP-UX 11.x Shutdown

Shut the system down with one of the following commands:

shutdown - provides an automated shutdown procedure that notifies users that a system halt is pending. Processes are killed using `kill -SIGTERM`.

shutdown -r 600 - reboots the system five minutes after the command is received. The `-h` option, which is the default, will halt the system.

reboot - immediately synchronizes the disks, terminates all currently executing processes (using `kill -SIGKILL`) except those essential to the system, halts the system, then performs a reboot. When invoked with the `-h` option, the system is halted without rebooting.

`/sbin/init <run level>` - `init` followed by a run level causes the system to begin executing at the specified run level. For example, `init s` takes the system to single-user level.

`/sbin/telinit <run level>` - The `telinit` command is hard-linked to `init` and has the same functionality as `init` followed by the new run level.

1.5 HP-UX Series 800 Firmware Start-up

1. Start the boot process either by powering on the machine, or by shutting down a currently-running system (with `/usr/sbin/shutdown`). If the machine is hung, type **<control>B** followed by **RS** (restart) at the system console to restart without power-cycling.
2. The firmware will go through a hardware testing process, then will prompt for any key to be pressed to interrupt the normal boot process.
3. Once the normal boot process has been interrupted, the system will ask if you want to boot from the primary boot path or from an alternate boot paths. Select the appropriate boot path.
4. The system will ask if you want to interact with the Initial Program Loader (IPL). Enter Y (Yes).
5. To boot into single-user mode-for example, to correct a start-up file which is causing the boot process to hang-type the following at the `ISL>` prompt:

hpux -is

To correct problems with the Logical Volume Manager, boot into maintenance mode:

hpux -lm

1.6 HP-UX Series 700 Firmware Start-up

1. Start the boot process by either powering on the machine or by shutting down a currently-running system (with `/usr/sbin/shutdown`). No console mode break similar to the series 800 **<control>B** is available on series 700 machines.
2. The firmware will prompt you to press the escape **<ESC>** key to interrupt the boot sequence.
3. Once the boot process has been interrupted, the `BOOT_ADMIN>` prompt will be displayed. Type **boot pri** to interact with the Initial System Loader.
4. To boot into single-user mode-for example, to correct a start-up file which is causing the boot process to hang-type the following at the `ISL>` prompt:

hpux -is

To correct problems with the Logical Volume Manager, boot into maintenance mode:

hpux -lm

NOTE: The behavior of the boot process will vary depending on the version of the software in your firmware. Refer to the documentation for your system if the behavior you see is different from what is described here.

[\[Next Chapter: 2. Swap Space Configuration\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 2 Swap Sapce Configuration

[\[Last Chapter: 1. System Start-up And Shutdown\]](#)

Solaris, HP-UX 10.x and HP-UX 11.x support swapping to raw disk partitions and file systems and using physical memory as a swap area also. This chapter describes how to configure secondary swap areas on each system. (The primary swap area is configured at installation time.)

Use common sense when configuring swap space. It does not make sense to create secondary swap space on a disk that is already overloaded. Also try to spread out swap areas over different disks and different controllers. This is especially important on Sun systems, where there is no priority system to specify how swap space is to be used.

Swap size will depend on the amount of RAM available and disk size(s). At a minimum, swap should be equal to the amount of RAM when RAM is less then 2GB. For systems with RAM greater than 2GB. You will need to determine how the system will be utilized.

Physical memory is the most efficient swap area, but you are limited by the size of the physical memory. Usually, physical memory swap space is augmented by disk swap areas. It is more efficient to swap to a raw partition than to a file, since swapping

to the partition does not involve the overhead of the file system. However, it is time-consuming to re-partition the disks to free up a raw partition for swapping.

The swap file is a good temporary solution to a swapping problem. The swap file provides the necessary swap space to allow the system to work, and you can schedule time off-hours to re-partition the disks to create a raw partition.

2.1 Swap Space on Solaris

Swap space on Solaris is interleaved with no priority system. All swap partitions and files are used equally. To view the current swap space usage:

```
solaris% swap -l
```

swapfile	dev	swaplo	blocks	free
/dev/dsk/c0t0d0s1	136,1	16	1049312	1049312

The *-l* option to the *swap* command will display information about each swap device and swap file. In this example, there is only one swap device, */dev/dsk/c0t0d0s1*. The *dev* field shows the major and minor device numbers for raw swap devices (136 and 1, respectively). The *swaplo* field (16) is the offset in 512-byte blocks where usable swap space begins. The *blocks* field (1049312) is the size of the swap area in 512-byte blocks, and the *free* field (1049312) is the number of 512-byte blocks that are not currently being used. The *-l* option to *swap* is useful for determining which swap devices you have configured on a system.

2.1.1 Adding a Raw Swap Partition

To add a raw swap partition to a Solaris system, identify an unused partition on a disk. Add the following entry to */etc/vfstab*, taking care to use the correct device name:

```
/dev/dsk/c0t1d0s0 - - swap - no -
```

Enable the new partition with the *swap* command:

```
solaris# swap -a /dev/dsk/c0t1d0s1
```

Check your change:

```
solaris# swap -l
```

swapfile	dev	swaplo	blocks	free
/dev/dsk/c0t0d0s1	136,1	16	1049312	1049312
/dev/dsk/c0t1d0s1	136,1 8	16	3293120	3293120

Note that the new swap area is now displayed.

2.1.2 Adding File System Swap

Solaris supports swapping directly to a file. First, use `mkfile` to create the swap file:

```
solaris# mkfile 250m /opt/swapfile
```

This command will create a 250-megabyte file that can be used for swapping. Enable the new swap file with the `swap` command:

```
solaris# swap -a /opt/swapfile
```

Check your change:

```
solaris# swap -l
```

swapfile	dev	swaplo	blocks	free
/dev/dsk/c0t0d0s1	136,1	16	1049312	1049312
/opt/swapfile	-	16	511984	511984

Now two swap areas are defined. Note that no major/minor device numbers are listed for a file system swap file.

To enable the new swap file at the next system boot, add the following entry to `/etc/vfstab`:

```
/opt/swapfile - - swap - no -
```

The swap file will be added at boot time by `/sbin/swapadd`.

2.1.3 Disabling Swap Space

Solaris provides the ability to disable a swap file while the system is running. This is done with the `-d` option to `swap`. All allocated blocks are copied to other swap areas.

```
solaris# swap -d /opt/swapfile
```

```
solaris# swap -l
```

swapfile	dev	swaplo	blocks	free
/dev/dsk/c0t0d0s1	136,1	16	1049312	1049312

Note the use of *swap -l* to display the individual swap areas.

2.2 Swap Space on HP-UX 10.x and HP-UX 11.x

Currently configured swap space and swap space usage can be displayed with the `swapinfo` command. The `-m` option indicates that values should be displayed in megabytes instead of in kilobytes:

```
hpux# swapinfo -m
```

	Mb	Mb	Mb	PCT	START/	Mb		
TYPE	AVAIL	USED	FREE	USED	LIMIT	RESERVE	PRI	NAME
dev	4096	0	4096	0%	0	-	1	/dev/vg00/lvol2
reserve	-	298	-298					
memory	3159	28	3131	1%				

HP-UX 10.x and HP-UX 11.x supports using physical memory for swap space. This is called "pseudo-swap space" and it is configured by default into the kernel; up to 87.5% of physical memory may be used as swap space. To determine how much physical memory your HP-UX 10.x system has:

```
hpux# /usr/sbin/dmesg | grep -i physical
```

```
physical page size = 4096 bytes, logical page size = 4096 bytes
```

```
Physical: 4194304 Kbytes, lockable: 3226964 Kbytes, available: 3707876 Kbytes
```

This system has $4194304/1024 = 4096$ megabytes of memory.

Note that if there have been many system messages, or if the system has been up for a long time, this information may have been flushed from the kernel.

The priority field (PRI) indicates the order in which space is taken from the different swap areas. Space is used first from swap areas with the lowest priority number. When you add a new swap area, you may specify a priority. This feature allows you to assign a higher priority to the faster swap areas so that slower areas will only be used when the primary swap areas become full.

Swap areas with the same priority are interleaved. If a raw partition and a file system swap file have the same priority, the raw partition will be used first.

2.2.1 Adding a Raw Swap Partition

To add a raw partition as secondary swap space on an HP-UX 10.x/11.x system, first identify which disk will be used as the raw swap space. It is possible for a disk to be configured with both a file system and a raw swap partition. Refer to "Installing Peripherals" in the Instant Information CD for information on how to do this.

Once the device has been identified, enable the swap area:

```
hpux# swapon -e -p 1 /dev/dsk/c0t5d0
```

This command will add the device */dev/dsk/c0t5d0* as a raw swap area. The *-e* option should be used only for disks which contain both a file system and a swap area. The *-p* option assigns the priority.

To enable the new swap area at boot time, add the following entry to */etc/fstab*:

```
/dev/dsk/c0t5d0 / swap end 0 0
```

The new swap area will be enabled as a file system swap area when */sbin/rc2.d/S500swap_start* is executed at system boot time.

2.2.2 Adding File System Swap

File system swap on an HP-UX 10.x/11.x systems are defined using a directory instead of a file. The amount of file system space that is used for swapping can grow as additional space is needed. Maximum and minimum values can be specified when the swap space is added:

```
hpux# /usr/sbin/swapon -l 5000 -r 10000 -p 2 /opt
```

This example specifies the */opt* file system as the location of the new swap area. The *swapon* command will create a directory called */opt/paging* which will be used as the location for the new swap area. By default, the number of blocks initially allocated

for swapping is 0. The *-l* option indicates the maximum number of file system blocks to be used for swapping. This example specifies a maximum of 5000 blocks which, at 8 kb per block, produces a swap area of 40 Mb (maximum). If you are not sure of the block size of a file system, the block size can be checked with *dumpfs*.

The *-r* option is the amount of space reserved for file system use. The swap file usage will not exceed the size of the file system less this value, regardless of the number specified with the *-l* option. If the "l" and "r" values conflict, the "r" value is used. The "*-p*" option specifies the priority.

Check the change with *swapinfo*:

```
hpux# swapinfo -m
```

TYPE	Mb AVAIL	Mb USED	Mb FREE	PCT USED	START/ LIMIT	Mb RESERVE	PRI	NAME
dev	4096	0	4096	0%	0	-	1	/dev/vg00/vol2
localfs	40	0	40	0	40	78	2	/opt/paging
reserve	-	298	-298					
memory	3159	28	3131	1%				

To enable this swap area at boot time, add the following entry to */etc/fstab*:

```
default /opt swapfs min=0,lim=5000,res=10000,pri=3 0 0
```

/opt will be enabled as a file system swap area when */sbin/rc1.d/S500swap_start* is executed at system boot time.

NOTE: Only one file system swap area per file system can be specified. You should also modify the kernel parameter *maxchunksize* if you want a swap area greater than 512 Mb. You can use SAM to configure swap space and kernel parameters. (Generally, SAM is easier to use for swap space management than the command line is.)

2.2.3 Disabling Swap Space

There is no way to disable swap space while the system is running. To disable a swap area at the next system boot, remove the swap entry from */etc/fstab*, reboot, and remove the swap file (for file system swap).

[Next Chapter: 3. Start-up Configuration Files]

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 3 Start-up Configuration Files

[\[Last Chapter: 2. Swap Space Configuration\]](#)

This chapter describes the start-up configuration files which are used to enable swap space and to set up the system's IP address, hostname, system time, and mounted file systems.

3.1 Solaris

System V machines operate on the "run level" concept. The run-level configuration file */etc/inittab* specifies which processes and daemons run and at which run level.

Here is the complete list of Solaris run levels:

0 - System is completely shut down. All processes are terminated and all file systems are unmounted. You are at either the ">" or "ok" prompt.

1 - Single-user mode. All system services and daemons are terminated and all file systems are unmounted.

2 - Multi-user mode, except NFS is not enabled.

3 - Multi-user mode. This is the normal operational default state. NFS is enabled.

4 - Alternative multi-user mode. This run level is currently not used.

5 – Shutdown the system so that it is safe to remove power. Note: On Newer systems it will perform a power off.

6 - Reboot. All active processes are terminated and all file systems are unmounted. The system will be rebooted to the run-level specified by the default run level entry in */etc/inittab*.

S,s - Single-user mode. This run level is equivalent to run level 1.

Normal activities require only run levels 0, 1, and 3.

When the system boots, *init* scans the */etc/inittab* file to determine the default run level:

```
is:3:initdefault:
```

In this example, the default run level is 3, or multi-user mode. The syntax of the *inittab* file is: *id:runlevel:action:process*. Here is the breakdown of the syntax:

id - A one- to four-character unique identifier for this entry in the *inittab* file.

runlevel - A valid run level. You may specify more than one run level. For example:

```
s2:23:wait:/sbin/rc2
```

This entry will cause the */sbin/rc2* script to be executed at both run levels 2 and 3.

action - The action *init* should take for the process specified. Valid actions are:

- respawn (respawn the process when entering the specified run level),
- wait (wait for the process to complete),
- once (run the process only once),
- boot (execute the process only at boot time),
- bootwait (execute at boot time, and wait for the process to complete),
- powerfail (execute only when *init* receives a *SIGPWR* signal),
- powerwait (like powerfail, and wait for process to complete),
- off (send a *SIGTERM* to process before terminating it during a shutdown),
- Initdefault (denotes the default run level), and
- Sysinit (Process listed in this entry is to be run before *init* starts processes associated with allowing logins).

process - A Bourne shell command to be executed. The process is executed as `sh -c exec <process>`, where `<process>` is the command specified in this field of the *inittab* file.

Here is an excerpt from a Solaris *inittab* file:

```
s2:23:wait:/sbin/rc2          >/dev/msglog 2<>/dev/msglog </dev/console
s3:3:wait:/sbin/rc3          >/dev/msglog 2<>/dev/msglog </dev/console
p3:s1234:powerfail:/usr/sbin/shutdown -y -i5 -g0 >/dev/msglog 2<>/dev/msglog
```

By default, when the system boots, it will go to run level 3-the default level. All processes associated with run level 3 will be executed. Run levels are additive, so in addition to executing the run level 3 processes, all lower level processes will be executed also. In the example above, if the system is booting to run level 3, */sbin/rc1*, */sbin/rc2* and */sbin/rc3* will be executed. If there is more than one process specified for a specific run level, the processes will be executed in the order in which they appear in the *inittab* file. Therefore, */sbin/rc1* and */sbin/rc2* will be executed before */sbin/rc3*.

Note that although the *p3* entry includes a run level of 3, it will be executed only in the event that a *SIGPWR* signal is sent to *init*. Commonly, this occurs when a Universal Power Supply (ups) is connected to the system. Many ups products have the ability to let the computer know when a power failure has occurred. The user can then write a program (if the ups vendor does not provide one) to send *init* a *SIGPWR* and execute commands to start shutting down processes gracefully.

By default, the */sbin/rcN* script is executed for run level *N*. Each */sbin/rcN* script runs through the files in */etc/rcN.d*. Note that */etc/rc*.d* files are hard-linked to scripts in */etc/init.d*. All the scripts in */etc/rc*.d* have a name in the format:

```
ANN<descriptive_name>
```

where

A = S or K,

NN = a unique number between 00 and 99, and

<descriptive_name> = string describing what the script does.

The descriptive name is also the file name used for the hard link in */etc/init.d*. For example, */etc/rc3.d/S15nfs.server* is a hard link to */etc/init.d/nfs.server*.

When the system enters the run level specified in the *inittab* file, all files beginning with a *K* are executed with the command line argument "*stop*." This is done first to ensure that any processes that should not be running are killed. (An example of when

processes should be killed is going from run level 3 to run level 2.) After all *K* files are executed, the *S* files are executed with the command line argument "*start*."

The order of execution of *K* and *S* files is based on the numeric values of the *NN* fields of the file names. Files are executed in numeric order (*S20syssetup* is executed before *S21perf*). Note that files within the same *rc*.d* directory are permitted to have the same numeric value, but this is not recommended as the order in which the files will be executed is not predictable.

3.2 HP-UX 10.x and HP-UX 11.x

Like Solaris, HP-UX 10.x and HP-UX 11.x operate on the "run level" concept. Here is the complete list of HP-UX run levels:

- 0 - System is completely shut down. All processes are terminated and all file systems are unmounted.
- 1 - Single-user mode. All system services and daemons are terminated and all file systems are unmounted.
- 2 - Multi-user mode, except NFS is not enabled.
- 3 - Multi-user mode. This is the normal operational default state. NFS and CDE are enabled.
- 4 - Multi-user mode with NFS and VUE (10.x only).
- S,s - Single-user mode. This run level is equivalent to run level 1.

NOTE: The syntax of */etc/inittab* is identical to that for Solaris: please refer to Section 3.1 for Solaris information.

When the system boots, *init* scans the */etc/inittab* file to determine the default run level:

```
init:3:initdefault:
```

Here is an excerpt from an HP-UX 11.x */etc/inittab* file (HP-UX 10.x looks almost identical):

```
init:3:initdefault:
```

```
brc1::bootwait:/sbin/bcheckrc </dev/console >/dev/console 2>&1 # fsck, etc.
```

```
cpirt::bootwait:/sbin/cat /etc/copyright >/dev/syscon      # legal req
```

```
sqnc::wait:/sbin/rc </dev/console >/dev/console 2>&1      # system init
```

```
#powf::powerwait:/sbin/powerfail >/dev/console 2>&1      # powerfail
```

```
cons:123456:respawn:/usr/sbin/getty console console      # system console
```

In this example, the default run level is 3. */sbin/rc* starts the appropriate start-up scripts in */sbin/rc*.d* where * is the run level. If the system is booting to run level 3, the scripts in */sbin/rc1.d*, */sbin/rc2.d*, and */sbin/rc3.d* will be executed.

The naming convention for these scripts is similar to that for Solaris. All the scripts have a name in the format:

```
ANNN<descriptive_name>
```

where

A = S or K,

NNN = a unique number between 000 and 999, and

<descriptive_name> = string describing what the script does.

The syntax of these scripts is identical to that for Solaris; again, each script takes a *start* or *stop* argument. When going from run level 3 to run level 2, the following scripts will be executed by *init*:

```
/sbin/rc2.d/K* stop
```

3.3 Customizing the HP-UX Start-up Process

Users may add individual scripts in one of two ways:

1. Add an entry to */etc/inittab*

or:

2. Add a script to the appropriate */sbin/rc*.d* directory

The advantage of adding an entry into */etc/inittab* is that the */etc/inittab* file can be scanned easily to identify modifications. The advantage of adding a script to the */sbin/rc*.d* directory is being able to take advantage of the stop/start capabilities described above.

/etc/inittab can be used to put a getty on a modem serial port.

To start up another daemon or service, simply add the script to the appropriate run-level directory (first determine on what run level the script should be run) and it will be executed automatically. Existing scripts can be used as a guideline; check the start/stop command line options. Also, choose high numbers for the numeric value of the file name, since HP starts with the low values. In cases wherein an individual script is to be executed before a vendor-provided script, name the script with a lower number.

[\[Next Chapter: 4. Devices\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 4 Devices

[\[Last Chapter: 3. Start-up Configuration Files\]](#)

This chapter describes both the differences between device-naming conventions on each system and the interoperability issues with moving devices between systems. Because SCSI (Small Computer System Interconnect) devices are common to both HP and Sun systems, this chapter will focus on these devices.

4.1 Naming Conventions

A summary of naming conventions is shown in the following table. Note that Solaris, HP-UX 10.x, and HP-UX 11.x use a hierarchical naming scheme wherein devices with similar characteristics are located in the same directory. For example, all cooked (or block) disks are located in */dev/dsk*.

Device	Solaris	HP-UX 10.x	HP-UX 11.x
Disk	/dev/dsk/cLt#d@s%	/dev/dsk/cLt@d%	/dev/dsk/cLt@d%
Example	/dev/dsk/c0t0d0s4	/dev/dsk/c0t6d0	/dev/dsk/c0t6d0
Tape	/dev/rmt/@{D,b,n}	/dev/rmt/cLt#d@D{n,b}	/dev/rmt/cLt#d@D{n,b}
Example	/dev/rmt/0bn	/dev/rmt/c0t3d0BESTnb	/dev/rmt/c0t3d0BESTnb

Table 4-1: Naming Conventions

C = Controller or drive type (e.g., sd for SCSI disks). An SCSI tape drive will be st or mt.

= SCSI target number (0-15).

{n} = Indicates no rewind option for tape drives. Optional.

{b} = Indicates which device should be accessed as a Berkeley-style device. Should be used with the mt command. Optional.

{c} = Enables data compression. Optional.

{D} = Density. HP-UX 10.x/11.x systems allow you to specify a density of SBEST which will use the optimal density for the tape drive.

{r} = Indicates raw (or character) device. Omitting this will cause the drive to be accessed as a cooked (or block) device. Optional.

@ = The drive number.

% = Slice number.

L = Logical controller number.

For more information, refer to the appropriate manual page. For example, *mt(7)*

4.2 Configuring Devices At Boot Time

4.2.1 Configuring Devices on Solaris

Sun systems have a prom monitor command, *probe-ide* (IDE drives) *probe-scsi* (SCSI drives), that will tell you all the IDE/SCSI devices currently connected to the first IDE/SCSI bus on the Sun system. Here's an example:

```
ok probe-ide
```

```
Device 0 (Primary Master)
```

```
    ATA Model: ST39140A
```

```
Device 1 (Primary Slave)
```

```
    ATA Model: ST39140A
```

```
Device 2 (Secondary Slave)
```

```
    Removeable ATAPI Model: CRD-8322B
```

```
Device 3 (Secondary Slave)
```

```
    Not Present
```

There are three IDE devices on the system: a ST39140A disk with IDE address 0, a ST39140A disk with IDE address 1, and a CDROM drive with IDE address 2. This is a good way to check addresses when a new IDE/SCSI device is added.

Run *probe-ide* or *probe-scsi* before and after adding the new device to ensure that everything is addressed properly and that there are no conflicts. Newer systems have a command called *probe-scsi-all* which lists all SCSI devices on all SCSI buses.

Table 4-1 lists the logical device names for Solaris. These files are actually symbolic links to the physical device file.

The physical name for */dev/dsk/c0t0d0s4* is:

```
/devices/pci@1f,0/pci@1,1/ide@3/dad@0,0:e
```

The physical device name describes the device with respect to how the device is installed on the system.

Devices are created automatically at boot time on Solaris if the system is booted with the following command:

```
ok boot -r
```

This command causes the system to perform an autoconfigure as it boots. The */devices* tree will be re-created based on the devices that are connected to the system and powered on at the moment. The */dev* tree will be updated to reflect changes in the */devices* tree.

4.2.2 Configuring Devices on HP-UX 10.x and HP-UX 11.x

These examples are specific for Series 700. HP Systems vary depending on the series, so check the documentation to see how other series differ from what is described here.

Let's first revisit the device names. Here is a long listing of a block disk device:

```
hpux% ls -l /dev/dsk
brw-r----- 1 bin      sys      31 0x026000 Sep 27 02:09 c2t6d0
```

The major device number is 31. The value of this number is defined in the master kernel configuration file */usr/conf/master.d/core-hpux*. Here is the line of interest:

```
*****
* Driver   Block major   Char major   Required for
*
*              minimal system
sdisk      31              188
```

This line indicates that the major number for SCSI block special device is 31, and that the major number for SCSI character special devices is 188. The major number defines to the kernel which device driver should be used.

The minor device number is 0x026000. The 0x indicates that the number is in hexadecimal format. The rest of the number can be broken down this way:

02 = The instance of the SCSI interface card. This will probably be 00 for the first SCSI controller.

6 = SCSI target number.

000 = Device driver specific information. For most SCSI disks, this will be 000.

The *ioscan* command will report on the active or usable hardware on a system. Here is the partial output on a Series 700:

```
hpux# ioscan -fn
```

```
Class   I H/W Path      Driver  S/W State      H/W Type  Description
=====
target  0 10/0/14/0.0    tgt     CLAIMED        DEVICE
disk    0 10/0/14/0.0.0  sdisk   CLAIMED        DEVICE    TEAC  CD-532E-B
                /dev/dsk/c0t0d0 /dev/rdisk/c0t0d0
```

The *-fn* option to *ioscan* indicates which devices are currently in use. This is useful for identifying unused SCSI target numbers when adding a new SCSI device.

Note that device files are automatically created at system boot time for all the devices connected to the system and *powered on* at boot time.

4.3 Interoperability Issues

NOTE: The following configurations have been tested by the authors of this manual, and the specific operation systems are listed. As hardware configurations can differ, be sure to perform your own tests when moving peripherals between systems

Device	Solaris	HP-UX 10.X	HP-UX 11.X
Sun CD-ROM	Yes	Yes*	Yes*
Sun Disk	Yes	Yes	Yes
HP DAT	Yes	Yes	Yes
HP CD-ROM	No	Yes	Yes
HP Disk	Yes	Yes	Yes

Table 4-3: Interoperability of Configurations and Systems

Yes = Successfully tested.

No = Not known to work.

* = Except as a boot device

NOTE: When a Sun disk is moved to an HP-UX system, all the disk's label and geometry information will be lost since HP-UX uses a different format. Save this information with *prtvtoc* (Solaris) before moving the disk. This will ensure that the required geometry information will be available if the disk ever needs to be re-installed on a Sun system.

4.3.1 HP-UX and CD-ROMS

The most common CD formats in use today are High Sierra, ISO 9660, and RockRidge. HP-UX can mount CDs which were created with any of these formats, but not all formats are read correctly.

4.3.1.1 High Sierra

No problems reading this format on HP-UX.

4.3.1.2 ISO 9660

If the version numbers are used in the file names, HP-UX will not be able to read the file names correctly. A file name can look something like *ABCDEFGH.ASC;1*. The ";" gets interpreted by the shell unless it is preceded by "\". This is an annoyance for users. Using the "-o cdcase" option when mounting the ISO 9660 CD will correct this behavior.

4.3.2 General Tips for Adding Devices

Shut down and power off the system and the peripherals before making any changes to the configuration. It is possible to get the bus into a hung state or to accidentally unplug the wrong cable; you don't want the system to be up when this happens. When booting the system, first turn on the peripherals, then turn on the CPU.

4.4 Adding a Disk to Solaris

1. Shut down the system.
2. Set the disk address to a unique address. Use *probe-scsi*, *probe-ide*, or *probe-scsi-all* to determine which addresses are being used.
3. Power off the system and all peripherals. Connect the new device and power on all the devices.
4. Interrupt the boot process by pressing **Stop** key and the "a" key simultaneously. Run *probe-scsi*, *probe-ide*, or *probe-scsi-all* to verify that the system sees all the devices. If a device does not show up at this point, it will not be available when the system boots. Recheck all the SCSI/IDE targets and cable connections. If necessary, re-configure so that only the device in question is connected to the system.
5. Boot the systems with the -r option to ensure that the device files for the new device get created.
6. It may be necessary to format the drive and/or re-partition the disk with the format program. Create one or more file systems with the *newfs* command.
7. Add the appropriate mount entry to */etc/vfstab*.

4.5 Adding an SCSI Disk to HP-UX 10.x and HP-UX 11.x

1. Using *ioscan*, find an available SCSI ID for the new device.
2. Shut down and power off the system and SCSI peripherals.
3. Set the disk address to the SCSI ID selected in step (1) above. Connect the new drive and power up the peripherals and the system.
4. Normally, the system will scan for new devices and create the device files when the system boots. However, if the device was turned off or if it was not connected properly, it will not show up. Once the problem with the device is corrected, start **SAM**. Select *Disks and Filesystems* then *Disk Devices*, under action select *Diagnose Missing Disks*. Alternatively, create the device file manually by using either *insf* or *mksf*.
5. If desired, you can initialize with *mediainit(1M)*. This will test reading and writing data to every block on the device:

```
hpux# mediainit /dev/rdisk/c0t5d0
```

6. Use the Logical Volume Manager to configure the new drive as a logical volume, or bring it into an already-existing logical volume. See the man page for *lvm(7)* for more information. Or, you can use **SAM** to configure the disk, add the appropriate entry to */etc/fstab*, and mount the new file system, if appropriate.

[Next Chapter: 5. Operating System Installation]

Provide feedback to: cookbook@apollo.hp.com

[Table of Contents] [Outline] [Index] (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 5 Operating System Installation

[\[Last Chapter: 4. Devices\]](#)

This chapter covers software installation of the operating systems from **CD-ROM**.

WARNING! If the system has already been installed, backup the entire system before installing or upgrading the operating system.

5.1 Installing Solaris

The following documents the installation of Solaris specifically onto a ULTRA-10 from CD-ROM. The system is installed as a networked stand-alone system. The log file from the installation is `/var/sadm/system/logs/install_log`.

1. Assemble the system. For full details, see the Hardware Owner's Guide that comes with the system.
2. Power on the system. Boot from CD-ROM.
3. Answer the questions for system information.

NOTE: If the system console is a terminal instead of a monitor and keyboard, the system will ask for the terminal type. Answer appropriately. The cursor keys will then function. Rather than pressing a function key when instructed (such as F2), press Esc and the number (e.g., Esc,2).

4. Choose to install or upgrade the system.

The installation software will search the root disk to determine if Solaris has already been installed there. The software will look in `/etc/vfstab` for a root entry and a valid device name (`/dev/dsk/c0t0d0s0`, for instance). If these are found, the system will ask if you want to upgrade your old software or install new software. Choose the appropriate answer.

5. Perform system installation and patch installation. The system confirms the choices, and installs Solaris.

5.1.1 Alternative Installation Methods

There are some instances in which CD-ROM installation is not the preferred method. These include when many systems are being installed, and when there is no CD-ROM local to the system being installed. In this section we provide an overview of more efficient methods for installing Solaris on multiple or remote machines.

5.1.2 Installing Via Disk Duplication

A "template" disk can be used to install the operating system onto another disk—for instance, onto the root disk in a new or old system. A template disk has the exact contents you want to place onto the target disk. The target disk must be at

least as large as the template disk. To duplicate the template to the target, follow these steps:

1. Install the template disk onto the target system, or install the target disk onto the system containing the template. Be sure there are no SCSI ID conflicts.
2. Boot from the template disk.
3. Run `format`, and partition the target disk. The partitions should be at least as large as the template disk's partitions. Place the new partition information onto the target disk with the `label` subcommand to the `format` command.
4. `newfs` each target-disk partition that is going to contain a file system.
5. Mount a target partition, using a mount point such as `/dup` (create it if necessary):

```
solaris# mount /dev/dsk/c0t1d0s0 /dup
```

In this instance, our target disk is `c0t1d0` and we're copying slice 0—usually the root partition.

6. Use a dump-restore pipe to duplicate each template file system onto the target system. For example:

```
solaris# cd /dup ; ufsdump 0f - /dev/dsk/c0t3d0s0 | ufsrestore rf -
```

In this instance, we're doing a `ufsdump` of `/dev/dsk/c0t3d0s0` and restoring it to the partition mounted as `/dup`.

NOTE: The location to which to restore the files is the current working directory. Be sure to `cd` to the restore directory before executing `ufsrestore`.

7. When the copy completes, modify the system files on the target root disk to change the hostname and network information. Otherwise, both the template and target disk will have identical system identifications.

Files to check and change if appropriate include `/etc/hosts`, `/etc/hostname.*`, `/etc/nodename`, `/etc/defaultrouter`, and `/etc/resolv.conf`.

8. "unmount" /mnt and repeat from step 5 with the next partition. Continue this cycle until all file systems from the template disk are copied to the target disk.

9. Before the target disk can be used as a boot device, we must install the boot blocks. Do so with a command such as:

```
solaris# /usr/sbin/installboot /usr/lib/fs/ufs/bootblk /dev/rdisk/c0t1d0s0
```

Note: your device name may be different.

The second argument is the root raw-disk partition of the target drive.

10. Replace the template and target disks onto the appropriate systems after setting their id's as desired.

Although there are many steps in this procedure, it is a useful method for the rapid copy of a template system. If the template has the operating system installed, patches installed, and facilities such as printing and terminal service configured, this procedure can be MUCH faster than installing each new system from scratch.

5.1.3 Installing Via JumpStart

JumpStart, or "auto-install," is a good method to use when installing many network-connected hosts. It involves extra administrative overhead to set up, when compared with CD-ROM installation. JumpStart is appropriate when more than a few similarly-configured systems are to be installed.

JumpStart is a turnkey, hands-off, automated, consistent, customizable installation process. It uses the diskless boot mechanism to allow a client to boot remotely. Then it takes over, partitions the clients disk(s), and loads specified software. Pre- and post-installation scripts allow the clients to be customized, patches to be installed, services to be added, and any other changes possible within a Unix shell script.

A detailed discussion of JumpStart is beyond the scope of this manual. See *Automating Solaris Installations, A Custom Jumpstart Guide*, by Paul Kasper and Alan McClellan, SunSoft (Prentice Hall) ISBN 013312505X for more information on JumpStart.

5.2 Installing HP-UX 10.x

The following documents the installation of HP-UX 10.x onto a new HP workstation from CD-ROM. The log file from the installation is `/var/adm/sw/swinstall.log`.

During the HP-UX 10.x cold install process, the user is expected to:

- * connect and test devices,
- * boot a minimal operating system from media,
- * install software products from media or a network,
- * execute HP-UX commands,
- * interact with user interface screens, and
- * create customized configuration files.

WARNING! A cold install destroys all data on the system disk! Be sure to back up the system if it already contains data.

1. Assemble the system if this is a new workstation.
2. Power on the CD-ROM drive and load the install CD. Power up the rest of the system. The system will select a default disk from which to boot. If this is not the device wanted-that is, if it does not contain the Install Media-quickly press the **Esc** key (older systems) or press any key (newer system). This will stop the boot process and display a menu of boot commands. (This menu varies, depending on the hardware.) Select the boot device containing the Install Media.
3. Enter the number of the language you want.
4. Within a few minutes, the installation interface appears:

Welcome to the HP-UX installation process!

Use the Tab and/or arrow keys to navigate through the following menu. Use the return key to select an item. If the menu items are not clear, select "Help" for more information.

```
[  Install HP-UX      ]
[  Run a Recovery Shell ]
[  Cancel and Reboot  ]
[  Advanced Options   ]
[      Help           ]
```

Choose *Install HP-UX* and when prompted

Would you like to enable networking now?[y]

answer **n**.

5. The input data screen appears. Answer the questions appropriately.
6. The system will perform "self discovery" to determine which disks are attached to the system. Highlight the appropriate disk with the arrow keys and use Tab to choose *OK*. Press **Return**.
7. The system asks for the desired file system layout. The default for Series 700 systems is non-lvm. Non-lvm creates one large partition. lvm creates several smaller partitions. Choose lvm if you have multiple disks or if you prefer separate partitions. Tab to *OK* and press **Return**.

NOTE: Be sure to configure */opt* with enough disk space to include both the current software and any other software you plan to install.

8. The configuration is displayed for review and for any necessary modifications. Press **F1** or **CTRL-F** for help with the choices at the current cursor location. Press **Return** at each highlighted parameter for pop-up help. When the parameters are correct, choose *OK*.

NOTE: Select *Long File System Names* for compatibility with Sun systems.

9. The configuration choices are displayed. The screen asks if you want to create a custom configuration. Note that doing so will create additional parameters to change. Press *Modify Disk/F8* to create custom file systems. Select *OK*.
10. The **SD-UX** *swinstall* command now allows customizing of the software bundle to be installed on the system. Choose *No* to install without modification the Desktop or Runtime bundle previously chosen.

Open bundles, products, and filesets to see contents, sizes, etc. Customize the system by marking some filesets for installation (filesets not marked will be ignored). After selection, the system will load the software within 30 to 60 minutes.

11. The system will automatically reboot from the system disk. You will be prompted for network information such as the hostname and IP address. Once this information is provided, you can administer the system and install application software.

5.3 Installing HP-UX 11.x

The following documents the installation of HP-UX 11.x onto a new HP workstation from CD-ROM. The log file from the installation is */var/adm/sw/swinstall.log*.

During the HP-UX 11.x cold install process, the user is expected to:

- * connect and test devices,
- * boot a minimal operating system from media,
- * install software products from media or a network,
- * execute HP-UX commands,
- * interact with user interface screens, and
- * create customized configuration files.

WARNING! A cold install destroys all data on the system disk! Be sure to back up the system if it already contains data.

1. Assemble the system if this is a new workstation.
2. Power on the CD-ROM drive and load the install CD. Power up the rest of the system. The system will select a default disk from which to boot. If this is not the device wanted-that is, if it does not contain the Install Media-quickly press any key. This will stop the boot process and display a menu of boot commands. (This menu varies, depending on the hardware.) Select the boot device containing the Install Media.
3. Within a few minutes, the installation interface appears:

Welcome to the HP-UX installation process!

Use the <tab> key to navigate between fields, and the arrow keys within fields. Use the <return/enter> key to select an item. Use the <return> or <space-bar> to pop-up a choices list. If the menus are not clear, select the "Help" item for more information.

[Install HP-UX]

[Run a Recovery Shell]

[Advanced Options]

[Reboot]

[Help]

4. Choose *Install HP-UX* and when prompted

User Interface and Media Options

This screen lets you pick from options that will determine if an Ignite-UX server is used, and your user interface preference.

Source Location Options:

- Media only installation
- Media with Network enabled (allows use of SD depots)
- Ignite-UX server based installation

User Interface Options:

- Guided Installation (recommended for basic installs)
- Advanced Installation (recommended for disk and filesystem management)
- No user interface - use all the defaults and go

Hint: If you need to make LVM size changes, or want to set the final networking parameters during the install, you will need to use the Advanced mode (or remote graphical interface).

[OK] [Help] [Cancel]

5. Choice “No user interface – use all the defaults and go”
6. Press OK to start the install.

The install will take approximately 30 minutes. At the end, you will be required to answer a few questions

7. Enter the Keyboard language
8. Are you ready to link this system to a network?

If you type “Y” then you will be required to answer the following questions:

- Enter the number of the Language you want
- Do you wish to use DHCP to obtain networking information?
- Enter a system Name

- Enter Time Zone
- Verify Time
- IP Address
- Subnetmask
- Gateway
- Domain Name
- DNS Server IP Address
- NIS (If you are running NIS)

If you type “N” then you will be required to answer these questions:

- Do you wish to use DHCP to obtain networking information?
- Enter a system Name
- Enter Time Zone
- Verify Time
- Set root password
- After these questions are answered the system will display a login prompt. Once logged in you can run “/etc/set_parms ip_address” to configure the ip address for the system.
- Second you can run “/etc/set_parms addl_netwrk” to configure the gateway, DNS, and NIS.

5.4 Installing HP-UX 10.x/11.x via Ignite-UX

Ignite-UX (Similar to Sun’s Jumpstart) is good method to use when installing many network-connected hosts. It involves extra administrative overhead to set up, when compared with CD-ROM installation. Ignite-UX is appropriate when more than a few similarly-configured systems are to be installed.

Ignite-UX is a turnkey, hands-off, automated, consistent, customizable installation process. It uses the diskless boot mechanism to allow a client to boot remotely. Then it takes over, partitions the clients disk(s), and loads specified software. Pre- and post-installation scripts allow the clients to be customized, patches to be installed, services to be added, and any other changes possible within a Unix shell script.

A detailed discussion of Ignite-UX is beyond the scope of this manual. See *Ignite-UX Administration Guide* for more information.

[\[Next Chapter: 6. Layered Software Installation\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 6 Layered Software Installation

[\[Last Chapter: 5. Operating System Installation\]](#)

This chapter covers the installation from CD-ROM of layered software products.

6.1 Installation on Solaris

The following documents an installation from CD-ROM of Solaris Documentation onto a Solaris system.

The Solaris-standard format for bundling software is a "package." There are a few methods available for installing layered products as packages. We will discuss two. The first method uses the command-line interface, and the second uses "Web Start".

6.1.1 Installing From the Command Line

Follow these steps:

1. Insert the Solaris Documentation CD-ROM into the drive.
2. Determine the space available in the recommended /opt partition:

```
solaris# df -k /opt
```

```
Filesystem    kbytes  used  avail capacity Mounted on
/dev/dsk/c0t3d0s5  75335 39114 28691   58% /opt
```

NOTE: Always install packages in /opt if possible, or unless instructed otherwise by the installation directions. If there is not

enough space in /opt, create a link from /opt to another partition so that all packages are available via the /opt directory.

3. Verify that the CD-ROM is mounted and verify the name:

```
solaris# ls /cdrom  
cdrom0      Solaris_8_sparc_2
```

4. (optional step) Run pkginfo to examine the list of available packages:

```
solaris# pkginfo -d /cdrom/sol_8_sparc_2/Solaris_8/Product
```

```
system  SUNWaccr   System Accounting, (Root)  
system  SUNWaccu   System Accounting, (Usr)  
system  SUNWapchS  Source for the Apache httpd server  
system  SUNWapchd  Apache Web Server Documentation  
.  
.  
.  
system  SUNWzsh    Z shell (zsh)  
system  SUNWzshS   Source for the Z shell (zsh)
```

NOTE: Installing a new version of an already-installed package will result in the new version being given a new version number by the system. The newest package will override the previous package(s).

NOTE: In all cases it is recommended that the user check the product notes accompanying the layered product for specific information about upgrading to a new release. Complex packages such as Solstice Disk Suite and the Veritas Sparc Storage Array management software have complex upgrade directions. Failure to follow these

directions exactly can result in data loss. Therefore, it is recommended also that the user run pkgrm on the old version of the package before installing a new version.

5. To begin the installation, run pkgadd:

```
solaris# pkgadd -d /cdrom/sol_8_sparc_2/Solaris_8/Product
```

6. The system will pose a series of questions. Follow the instructions in the installation guide and make the appropriate choices concerning the installation.

NOTE: If there is more than one package in the specified directory (`solaris_8_sparc_2` in this example), a menu of packages will be presented. Select individual items or select all the packages for installation. Each package will have its own set of questions and options.

7. Watch for a message indicating a successful installation.

8. The `eject` command will eject the CD and unmount the CD-ROM.

NOTE: `/var/sadm` contains information used by the package interface.

NOTE: Use `pkgrm` to remove packages.

6.1.2 Installing Using the Web Start GUI

1. Mount the CD in the CDROM Drive.
2. Run the installer. `/cdrom/cdrom0/installer`
3. Follow the prompts on the screen.

6.2 Installation on HP-UX 10.x/11.x

6.2.1 Software Distributor HP-UX

Software Distributor HP-UX (SD-UX) can be used to create and manage software packages on a local system.

There are several other "sw" tools of interest:

- *swinstall* installs software from a CD-ROM, tape, or software depot. (This program will be discussed in more detail in the next section.)
- *swcopy* copies software from CD-ROM or tape into one or more "depots" for later installation onto this or other systems.
- *swremove* deletes installed software or depots.
- *swlist* displays software object information for the system.
- *swconfig* configures, reconfigures, or unconfigures an installed software object.
- *swverify* compares installed software against the installation source to verify the integrity of the system.
- *swmodify* allows changes to the Installed Products Database, which contains information on software installations.

6.2.2 Installing Software With `swinstall` on HP-UX 10.x/11.x

The following documents an installation from CD-ROM of HP-UX 10.x onto a new HP workstation from CD-ROM:

1. Connect the CD-ROM drive to the workstation. Insert the product CD.
2. Create a mount point for the CD-ROM and mount the product CD:

```
hpux# mkdir /cdrom
```

```
hpux# chmod 555 /cdrom
```

```
hpux# /etc/mount /dev/dsk/c0t2d0 /cdrom
```

3. Start the installation tool `swinstall`:

```
hpux# swinstall -s /cdrom
```

Unless another location is specified through the `-s` option, `swinstall` assumes the software is installed in the local depot location. (The depot is a repository of software packages to be installed. By default, the depot location is `/var/spool/sw`.) To change the default location of `swinstall`'s software source, add the following line to `/var/adm/sw/defaults`:

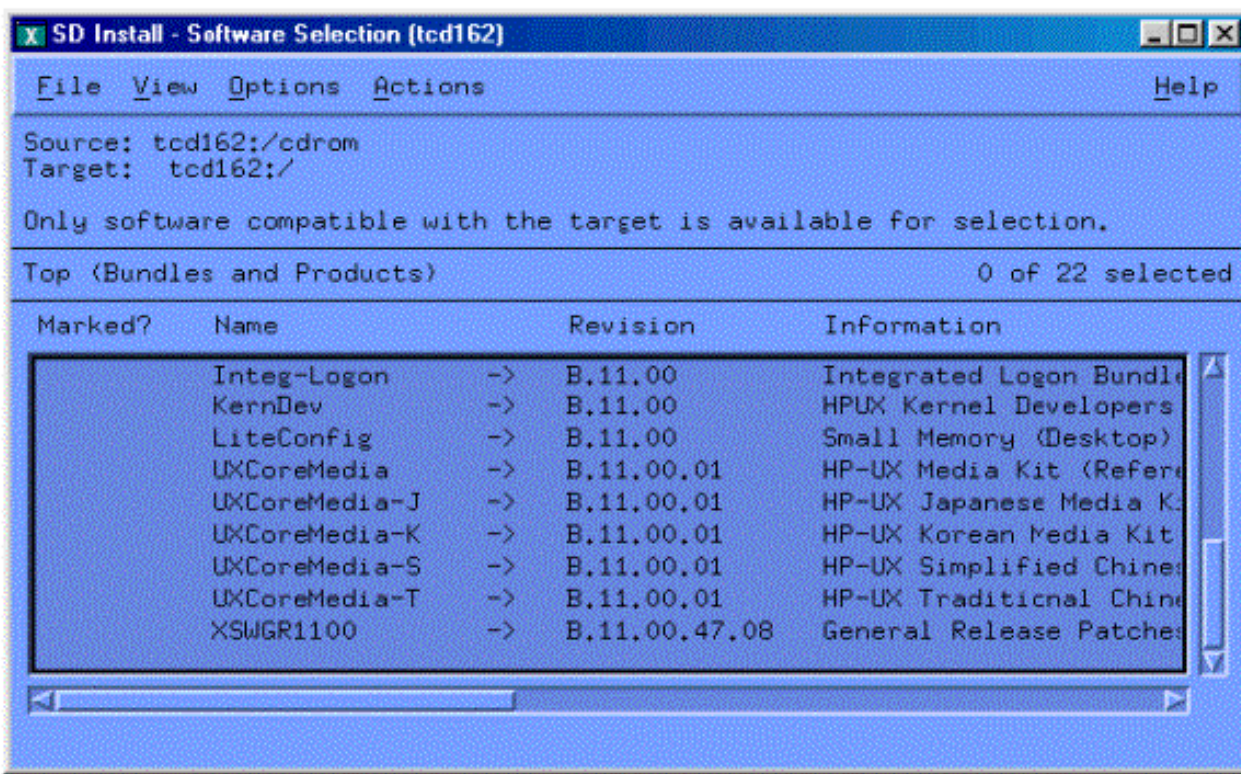
```
swinstall.source_directory = /mylocation
```

where `/mylocation` is the desired location.

`/usr/lib/sw/sys.defaults` has examples of other values that can be changed.

NOTE: `swinstall` can also be started by using **SAM**. To do this, select "*Software Management*" and then "*Install Software to Local Host*."

The `swinstall` window is shown here:



4. Select the item (i.e., software object) to install.

Click on the item, and select *Action -> Mark for install*.

swinstall will perform a dependency check to ensure that all software required by that item, or software object, is already installed.

5. Install the selected software objects.

Once all desired items have been marked, select *Action -> Install (analysis)*.

swinstall will analyze disk space requirements and will report problems if any exist. To see the details of the analysis, view the logfile by selecting Logfile in the "Install Analysis" window.

Select OK to begin installation.

NOTE: Make sure */opt* has plenty of available space. Many software packages expect to be installed in */opt*. If **LVM** is being run, */opt* can be increased, but it must be done from single-user mode because at higher run levels the file system will be active.

[\[Next Chapter: 7. TCP/IP Communications\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 7 TCP/IP Communications

[\[Last Chapter: 6. Layered Software Installation\]](#)

ARPA Services is a collection of utilities developed by the University of California at Berkeley (UCB), Cornell University, Carnegie-Mellon University (CMU), and other institutions. These utilities allow file transfers, remote logins, remote command execution, lookup services (such as hostname lookups), and electronic messages between computer systems. Most of the utilities are TCP/IP (Transmission Control Protocol/Internet Protocol) services. (e.g., telnet and the file transfer protocol (FTP)).

The services available on these machines are:

Service	Description	Solaris	HP-UX10.x	HP-UX11.x
telnet	Remote terminal login	Y	Y	Y
rlogin	Remote terminal login	Y	Y	Y
rsh	Remote command execution	Y	N	N
remsh	Remote command execution	N	Y	Y
rexec	Retruns stream to command	Y	Y	Y
ftp	File transfer	Y	Y	Y
tftp	Trivial file transfer	Y	Y	Y
talk	Talk service	Y	Y	Y
biff	Mail notification	Y	N*	N*
finger	User identification	Y	Y	Y

Table 7-1: Services Available by Operating System

* Public domain versions of these programs can be obtained from the InterWorks anonymous FTP site ([ftp.interworks.org](ftp://interworks.org)), or from one of the HP-UX Software Archive anonymous FTP sites (hpux.cae.wisc.edu). For additional sites, refer to the HP-UX Frequently Asked Questions (FAQ) List which is available at ftp://rtfm.mit.edu/pub/usenet/comp.sys.hp.hpux/comp.sys.hp.hpux_FAQ.

Note that *remsh* and *rsh* are essentially the same command with different names. *remsh* is the HP-UX command to spawn a remote shell; *rsh* is the Solaris command. On HP-UX, *rsh* runs a restricted shell that is, a shell that limits the features available to the user. You may consider aliasing *rsh* to *remsh* (or vice versa) if you find it difficult to remember which command to use on which system.

7.0.1 Interface Names

Network designation	Network Type	Platform
lo0	Loopback	All
lan#	Ethernet (802.3)	HP-UX 10.X, 11.X
le#	LANCE Ethernet Board	Solaris
ie#	Intel Ethernet Board	Solaris
hme#		Solaris

Table 7-2: Interfaces

The le0, ie0 and hme0 designation will vary depending on the model of the Sun system. The trailing digit is incremented for each controller found on the system (for example, le0, le1, le2, etc.).

7.1 Solaris

To place a Solaris machine into an established network and set up TCP/IP communications:

1. Set the system's hostname and IP address in `/etc/hosts`. In this example, the example host's name is "roadrunner":

```
198.252.182.23 roadrunner.foo.com roadrunner
```

2. Modify `/etc/hostname.hme0` (or `hostname.ie0`) and `/etc/nodename` to contain the correct hostname:

```
roadrunner
```

3. Modify `/etc/inet/netmasks` to have the appropriate entry:

```
198.252.182.0    255.255.255.0
```

The `/etc/inet/netmasks` file contains network masks used to implement IP standard subnetting. For each network that is subnetted, a single line should exist in this file. The line should contain the network number, any number of **SPACE** or **TAB** characters, and the network mask to use on that network.

4. Add the IP address of the default gateway to `/etc/defaultrouter`:

```
198.252.182.20
```

5. Reboot the system.
6. Check network interface configuration with `ifconfig`:

```
solaris# ifconfig hme0
hme0:
flags=863<UP,BROADCAST,NOTRAILERS,RUNNING,MULTICAST> mtu 1500
inet 198.252.182.23 netmask fffffff0 broadcast 198.252.182.255
ether 8:0:20:11:34:6e
```

7. Check the route information with `netstat -r`. In this example, host "bob" has IP address 198.252.183.20.

```
solaris# netstat -r | grep default
default    bob      UG    0    3
```

8. Verify basic network connectivity to other systems using `ping`:

```
solaris% /usr/sbin/ping -s 15.2.72.150
PING 15.2.72.150: 64 data bytes
72 bytes from hp-ux (15.2.72.150): icmp_seq=0. time=7. ms
72 bytes from hp-ux (15.2.72.150): icmp_seq=1. time=1. ms
72 bytes from hp-ux (15.2.72.150): icmp_seq=2. time=1. ms
72 bytes from hp-ux (15.2.72.150): icmp_seq=3. time=1. ms

----15.2.72.150 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 1/2/7
```

The packets are numbered (*icmp_seq*) and the round trip time for each packet is displayed. The packet loss value is an indication of performance in communicating with a remote host. On a local area network, this packet loss value should be very close to zero. However, on a wide area network, there may be some packet loss, and on portions of the Internet, there may be high packet loss. (High packet loss is not uncommon on the Internet and it is extremely annoying.)

9. If you are running NIS or DNS, add the hostname and IP address of the new system to the NIS or DNS databases. If you are not running either of these services, add the hostname and IP address of other systems to the */etc/inet/hosts* file.

7.1.1 Solaris and Routing

Solaris systems will execute *in.rdisc*, which uses the icmp router discovery protocol to locate routers on the network. (Note that your router must support ICMP in order for the discovery to work properly.)

The script will then automatically update the routing tables accordingly. Thus, a default route will not have to be specified manually. Typically, you will configure your system for dynamic routing, but instructions for static routing configuration are included for completeness.

7.1.1.1 Solaris and Dynamic Routing

To use dynamic routing, make sure that */etc/defaultrouter* does not exist. If you wish for the system to also behave as a router, configure */etc/gateways*. Here is an example of an */etc/gateways* entry that defines a default active route:

```
net 0.0.0.0 gateway 198.252.182.134 metric 1 active
```

Execute *route* with the *-f* option to flush the current routing table of all gateway entries, then re-start *in.routed*:

```
solaris# /etc/rc2.d/S69inet stop
solaris# /etc/rc2.d/S69inet start
```

7.1.1.2 Solaris and Static Routing

To use static routing, add the names of one or more default routers to */etc/defaultrouter*. The default route will be configured at system boot time from */etc/rc2.d/S69inet*.

Execute *route* with the *-f* option to flush the current routing table of all gateway entries, then re-start *in.routed*:

```
solaris# /etc/rc2.d/S69inet stop
```

```
solaris# /etc/rc2.d/S69inet start
```

7.2 HP-UX 10.x and HP-UX 11.x

To place an HP-UX 10.x or 11.x machine into an established network and set up TCP/IP communications.

1. Set the system's hostname, IP address, and subnet mask in */etc/rc.config.d/netconf*. In this example the hostname is "bob":

```
HOSTNAME="bob"  
IP_ADDRESS[0]="198.252.182.22"  
SUBNET_MASK[0]="255.255.255.0"
```

NOTE: Each interface name is specified by a number in the */etc/rc.config.d/netconf* file. The first interface is 0. This is the reason for using the specification: *IP_ADDRESS[0]="198.252.182.22"*.

2. Execute the system start-up script:

```
hpux# /sbin/rc2.d/S340net stop  
hpux# /sbin/rc2.d/S340net start
```

The *S340net* script takes the information from */etc/rc.config.d/netconf* and performs *ifconfig* commands on each interface using the netconf data.

3. Check the network interface configuration with *ifconfig*:

```
hpux# ifconfig lan0  
lan0: flags=863<UP,BROADCAST,NOTRAILERS,RUNNING>inet 198.252.182.22 netmask ffffff00  
broadcast 198.252.182.255
```

4. Verify the host name change. It may be necessary to reboot in order to effect the change:

```
hpux# hostname  
bob
```

or:

```
hpux# uname -n
```

```
bob
```

5. Configure either static and/or dynamic routing. Which one you choose will depend upon the topology of your network. For more information, refer to *TCP/IP Network Administration book, O'reilly and Associates*

- a. To configure dynamic routing, first configure */etc/gated.conf*. A sample configuration file is located in */usr/newconfig/etc/gated.conf*. The format of */etc/gated.conf* is outside the scope of this manual.

Next, enable gated in */etc/rc.config.d/netconf* by setting the GATED variable to 1:

```
GATED=1
```

Start gated:

```
hpux# /sbin/init.d/gated start
```

- b. To configure static routing, set a default route in */etc/rc.config.d/netconf*:

```
ROUTE_DESTINATION[0]="default"
```

```
ROUTE_GATEWAY[0]="198.252.182.22"
```

```
ROUTE_COUNT[0]="0"
```

Execute the system start-up script:

```
hpux# /sbin/rc2.d/S340net stop
```

```
hpux# /sbin/rc2.d/S340net start
```

The */sbin/rc2.d/S340net* reads the route information from */etc/rc.config.d/netconf* and executes the appropriate route commands to set the routing values accordingly.

6. Check the route with *netstat*:

```
hpux# netstat -r
```

Routing tables:

Destination	Gateway	Flags	Refs	Use	Interface
127.0.0.1	127.0.0.1	UH	0	0	lo0
default	bob	UG	5	1329565	lan0

7 Test the connection by using *ping*. Ping reports statistics about network performance between two hosts; if there is a problem with the connection, *ping* assists in determining if the problem is in the upper layers of the network stack or if it is in the lower layers:

```
hpux# ping oak.zilker.net
PING oak.zilker.net: 64 byte packets
64 bytes from 198.252.182.129: icmp_seq=0. time=95. ms
64 bytes from 198.252.182.129: icmp_seq=1. time=35. ms
64 bytes from 198.252.182.129: icmp_seq=2. time=35. ms

----oak.zilker.net PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms) min/avg/max = 35/55/95
```

The packets are numbered (*icmp_seq*) and the round trip time for each packet is displayed. The packet loss value is an indication of performance in communicating with a remote host. On a local area network, this packet loss value should be very close to zero. However, on a wide area network, there may be some packet loss, and on portions of the Internet, there may be high packet loss.

7.3 Using SAM

A limited set of networking tasks may be performed from SAM. SAM can be used to change the host name and/or IP address of a system, but it cannot be used to configure routing.

7.4 Troubleshooting Hostname Changes

When changing hostname or other network configuration, the following list may help to verify proper operation or to track down a configuration error:

1. Does */etc/hosts* have the correct hostname and IP address?
2. If using DNS, does */etc/resolv.conf* reflect changed IP addresses, if necessary, for the DNS server(s)? Has the DNS nameserver administrator been informed about changed IP address and/or hostname?
3. If the machine is a DNS server, either run *hosts_to_named* or edit the *db.** files by hand if needed and re-start DNS.
4. If the machine is an NIS client and not a server, has the NIS Master server administrator been informed of the change, or has */etc/hosts* been edited on the NIS master server to reflect the new hostname or IP address?
5. If changing IP addresses, were any changes necessary to */etc/bootptab*? Was it necessary to re-configure any network printers and X-terminals that boot with bootp?

[Next Chapter: 8. NIS and NIS Plus]

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 8 NIS and NIS Plus

[\[Last Chapter: 7. TCP/IP Communications\]](#)

This chapter describes how to configure an NIS client and an NIS server on a Solaris system; how to support NIS clients on a Solaris systems, and how to configure both HP-UX 10.x and 11.x systems as NIS clients, NIS slave servers, and NIS master servers.

When adding an HP-UX system to a Solaris NIS environment, making the HP-UX system an NIS client is the fastest way to get the system on-line and well integrated. As discussed in Chapter 10, Domain Name Service.

NOTE: This chapter does not describe how to configure the Domain Name Service. See Chapter 10, Domain Name Service, for information on DNS and for details on the interaction between NIS and DNS.

8.1 Configuring an NIS Master on Solaris

1. To configure a Solaris system as an NIS server, indicate that the system will be an NIS master during the initial OS install. The install program will copy the necessary files into `/var/yp`.
2. Copy `/etc/nsswitch.files` to `/etc/nsswitch.conf`
3. Edit the `/etc/hosts` or `/etc/inet/ipnodes` file to add the name and IP address of each of the NIS servers.
4. Add the domain name to `/etc/defaultdomain`. For example:

```
class.hp.com
```

The domain name will be set at the next system boot. To set the domain name immediately:

```
solaris# domainname `cat /etc/defaultdomain`
```

The host name of the NIS server is solaris.

5. Create and initialize the NIS database:

```
solaris# /usr/sbin/ypinit -m
```

6. Reboot the system. Then logon and verify that NIS is running.

```
solaris# ypwhich
```

```
solaris
```

```
solaris# ypcat passwd
```

<User should see a copy of the NIS password file.>

8.2 Configuring an NIS Client on Solaris

To set up a Solaris system as an NIS client, perform the following steps:

1. run `ypinit` command.

```
solaris# ypinit -c
```

2. Start ypbind (or reboot) to bind to the NIS server.

3. Test the configuration:

```
solaris2# ypwhich
```

```
solaris
```

```
solaris2# ypcat passwd
```

<User should see a copy of the NIS password file.>

8.3 Supporting NIS Clients From NIS+ on Solaris

1. Modify /etc/init.d/rpc by uncommenting the following line:

```
#EMULYP = Y
```

2. Re-start rpc.nisd with the -Y option:

```
solaris# /etc/init.d/rpc stop
```

```
solaris# /etc/init.d/rpc start
```

3. Finally, add a des credential for each NIS client. The following command will create a des credential for host oak.class.hp.com in the NIS+ domain named test.class.hp.com:

```
solaris# nisaddcred -p unix.oak@class.hp.com \
```

```
-P oak.test.class.hp.com. des test.class.hp.com.
```

The files necessary to run NIS on a Solaris system are on the Solaris CD-ROM.

8.4 Configuring an HP-UX 10.x/11.x System as an NIS Client

1. To configure an HP-UX 10.x system as an NIS client, execute the following command:

```
hpux# /sbin/set_parms addl_netwrk
```

The system will offer the option of modifying the network configuration information, DNS, and NIS. Indicate "no" to the first two questions and change only the NIS configuration.

The system will ask the following questions (user's answers supplied):

Enter the NIS domain name, then press [Return] or just press [Return] to select the current NIS domain name (noname): class.hp.com

Wait for NIS server on bootup?

Press [y] for yes or [n] for no, then press [Return], or just press [Return] to keep the current selection: y

(You should answer y for "yes" to this question.)

/sbin/set_parms will modify the following lines in /etc/rc.config.d/namesvrs.

```
NIS_CLIENT=
```

```
NISDOMAIN=
```

will be changed to:

```
NIS_CLIENT="1"
```

```
NISDOMAIN="class.hp.com"
```

/sbin/set_parms will not start ypbind.

2. Make the appropriate changes to /etc/passwd and /etc/group

For password file add the following to the last line: "+:*:-2:-2::/:"

For group file: "+:*:*"

3. Start the NIS client daemons:

```
hpux# /sbin/init.d/nis.client start
```

4. Check the binding with ypwhich:

```
hpux# ypwhich
```

```
solaris
```

8.5 Configuring an HP-UX 10.x/11.x System as an NIS Slave Server

1. Manually set the domain name:

```
hpux# domainname class.hp.com
```

2. Initialize the slave server with ypinit:

```
hpux# /usr/sbin/ypinit -s solaris
```

where solaris is the name of the NIS master.

The general form of this command is:

```
/usr/sbin/ypinit -s <ip addr of NIS MASTER> DOM="mydomain.com"
```

The system will offer the option of aborting on errors. Answer no. (Errors will have to be corrected and ypinit rerun, however.) If the NIS domain already exists, the system will offer the option of deleting the old database files and directory. The NIS slave server will download its copy of the NIS maps from the master server.

3. Modify /etc/rc.config.d/namesvrs by changing the following lines:

```
NIS_SLAVE_SERVER=
```

```
NISDOMAIN=
```

to:

```
NIS_SLAVE_SERVER=1
```

```
NISDOMAIN="class.hp.com"
```

4. Add the name of the slave server to the ypservers map on the NIS master.
5. Start the NIS daemons:

```
hpux# /sbin/init.d/nis.server start
```

```
hpux# /sbin/init.d/nis.client start
```

6. Check the binding with ypwhich:

```
hpux# ypwhich
```

```
solaris
```

NOTE: For the HP-UX system to be an NIS client also, follow the steps in Section 8.4 for configuring an HP-UX 10.x system as an NIS client.

8.6 Configuring an HP-UX 10.x/11.x System as an NIS Master

The NIS programs are located in /etc/yp and NIS database files are located in /var/yp.

1. Initialize the NIS master:

```
hpux# /usr/sbin/ypinit -m DOM=class.hp.com
```

Note that the command line options are slightly different from those on Solaris. Otherwise, ypinit is identical to the Solaris version. If the domain name is set using the domainname command, omit the DOM command line option. The domain name will need to be set manually at some point as ypinit will not do this.

2. Modify /etc/rc.config.d/namesvrs by changing the following lines:

```
NIS_MASTER_SERVER=
```

```
NISDOMAIN=
```

to:

```
NIS_MASTER_SERVER=1
```

```
NISDOMAIN="class.hp.com"
```

3. Start the NIS daemons:

```
hpux# /sbin/init.d/nis.server start
```

```
hpux# /sbin/init.d/nis.client start
```

4. Check the binding with ypwhich:

```
hpux# ypwhich
```

```
hpux
```

NOTE: For the HP-UX system to be an NIS client also, follow the steps in Section 8.7 for configuring an HP-UX 10.x system as an NIS client.

8.6.1 Updating NIS Maps on HP-UX 10.x/11.x

The commands for updating the NIS maps are the same as those for Solaris. `/var/yp/Makefile` has support for making all maps or individual maps. The HP-UX 10.x /11.x Makefile also includes support for `auto.master`. Use this entry as a template for adding support for other maps (such as `auto.home`).

8.7 Configuring an HP-UX 11.x System as an NIS Plus Client

From the NIS Plus master server:

```
solaris# nisclient -co hpux
```

1. When prompted for a password, type the default password, (niplus).

From the NIS Plus client

2. Modify `/etc/rc.config.d/namesvrs` by changing the following lines:

```
NIS_MASTER_SERVER=  
NIS_SLAVE_SERVER=  
NIS_CLIENT=  
NISPLUS_SERVER=  
NISPLUS_CLIENT=
```

to:

```
NIS_MASTER_SERVER=0  
NIS_SLAVE_SERVER=0  
NIS_CLIENT=0  
NISPLUS_SERVER=0  
NISPLUS_CLIENT=1
```

3. Issue the following command to initialize the client host:

```
hpux# nisclient -I -h solaris
```

4. start the NIS Plus daemons:

```
hpux# /sbin/init.d/nisplus.client start
```

5. Check to verify that you can get information from the NIS Plus tables

```
hpux# niscat passwd.org_dir
```

8.8 Configuring an HP-UX 11.x System as an NIS Plus Master Server

1. Log in as root to the host that will be the root master server.
2. Set the domainname, for example:

```
/usr/bin/domainname class.hp.com
```

3. Set the PATH variable to include /usr/lib/nis. If you are running the C shell, type the following command:

```
setenv PATH $PATH:/usr/lib/nis
```

If you are running the Bourne or Korn shell, type the following commands:

```
PATH=$PATH:/usr/lib/nis
```

```
export PATH
```

4. Issue the following command to set up the root master server:

```
hpux# nisserver -r
```

If you want the server to run in NIS compatibility mode so that it can serve NIS clients, add the `-Y` option.

```
hpux# nisserver -r -Y
```

The `nisserver` script asks you if the information it has is correct. You can change it by typing `n`. The script then allows you to change each piece of information. To make a change, just type the correct information after the incorrect information and press [Return]. You cannot change the security level.

When the `nisserver` script asks you for a password, type the root password. The `nisserver` script will use the root password to create credentials for the local host in the `cred` table.

5. To verify that the `nisserver` script created the root domain successfully, issue the following command:

```
hpux# nisl -lR
```

The `nisl` command should list the domain name, the `org_dir` and `groups_dir` NIS+ directories, the `admin` group, and all the standard tables listed in Standard NIS+ Tables.

6. Create a cron job that runs `nisping -Ca` at least once a day, during a time when the network is not busy. The following example crontab file runs `nisping -Ca` once a day, at 3:00 AM. It directs standard output and standard error from the `nisping` command to the file `/tmp/nisping.log`.

```
0 3 * * * /usr/lib/nis/nisping -Ca > /tmp/nisping.log 2& >1
```

The `nisping -Ca` command causes all servers of the domain to update their tables with the changes in the transaction log and to clear the transaction log. If you do not issue the `nisping -Ca` command regularly, your transaction log may grow too large, and you may not have enough disk space to checkpoint it.

8.8.1 Populating the NIS+ Tables on the Master Server for HP-UX 11.x

The nispopulate script may fail if there is insufficient /tmp space on the system. To keep this from happening, you can set the environment variable TMPDIR to a different directory. If TMPDIR is not set to a valid directory, the script will use the /tmp directory instead.

1. Log in as root to the master server.
2. If you will populate the NIS+ tables from files, create a temporary directory, and copy the files into it from the /etc directory:

```
mkdir /nis+files
cd /etc
cp auto_home auto_master group hosts mail/aliases netgroup \
networks passwd protocols rpc services TIMEZONE ../nis+files
```

3. In the temporary directory, remove the entry for root from the passwd file. Remove all other entries with user ID 0 (zero) from the passwd file. Remove all the system entries such as root and bin, with user IDs of less than 100, from the passwd file. This should be done both for security and for interoperability with NIS and other NIS+ implementations. Remove any other entries from the passwd, aliases, or hosts files that you do not want distributed across the namespace.
4. In the temporary directory, remove any fully-qualified DNS domain names from the hosts file. NIS+ cannot find fully-qualified DNS domain names in its hosts table unless the DNS domain name matches one of the NIS+ domain names in its namespace.
5. Replace any periods in automounter map names with underbars. For example, if your master map is called auto.master, rename it to auto_master. If you will populate your NIS+ tables from NIS maps, make sure your NIS map names contain no periods. If you will populate your NIS+ tables from files, make sure your file names contain no periods.
6. To populate the NIS+ tables from files, issue the following command. The -p option specifies the path to the files.

```
hpux# nispopulate -F -p /nis+files -d class.hp.com
```

7. To verify that your tables have been populated successfully, issue the niscat command for several standard tables. The standard tables are listed in Standard NIS+ Tables. The following example lists the contents of the NIS+ passwd table:

```
hpux# niscat passwd.org_dir.class.hp.com.
```

8. Issue the following command to checkpoint the domain:

```
hpux# nisping -Ca
```

9. Reboot the host to force long-running processes to read the new /etc/nsswitch.conf file. (The nisserver script copies

`/etc/nsswitch.nisplus` to `/etc/nsswitch.conf`.)

The `nispopulate` script populates the `cred` table from the `passwd` and `hosts` files or NIS maps. Every NIS+ principal except the root user on the root master server is given the default NIS+ password, which is `nisplus`. (The credentials for the root user on the root master server were created using the root password.) When you run the `nisclient` script to initialize a client host in the root domain, the `nisclient` script will change the client's credentials to use the client's root password instead of the default NIS+ password. See Section 8.7 Configuring an HP-UX 11.x System as an NIS Plus Client

[\[Next Chapter: 9. NFS and the Automounter\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 9 NFS and the Automounter

[\[Last Chapter: 8. NIS and NIS Plus\]](#)

The Network File System (NFS) allows computers to share file systems across the network. The automounter is a daemon that automatically mounts NFS file systems when they are needed and unmounts them when they are no longer needed.

On the HP-UX 11.x a new automounting utility, AutoFS, is available in addition to the pre-existing Automounter. You can configure your system to use either Automounter or AutoFS. Automounter is the default on a newly installed or updated system. However, you may choose to migrate to AutoFS, since it has several advantages over Automounter:

- AutoFS can be used to mount any type of file system, including NFS Protocol Version 3. (The pre-existing Automounter can be used only for NFS PV2.) Solaris has support NFS Protocol Version 3 since Solaris 2.5.
- With AutoFS the configured mount points are the actual mount points. (The pre-existing Automounter mounts directories under `/tmp_mnt` and creates symbolic links from the configured mount points to the actual ones under `/tmp_mnt`.)
- You do not have to stop AutoFS to change your automounter maps. The AutoFS daemon, `automountd`, runs continuously.

When you make a change to an automounter map, you run the automount command, which reads the maps and then exits. (The pre-existing automounter has to be killed and restarted whenever you make a change to an automounter map.)

In this chapter, the machine or machines mounting a file system via NFS shall be called the "*client*," and the system hosting the file system to be made available via NFS shall be called the "*server*." All systems covered in this chapter can be both an NFS client and server simultaneously.

There are three basic steps to using NFS:

1. Start the client and server NFS daemons as appropriate for each machine. (The daemons are *biod* and *nfsd*, respectively. Additional daemons required are *statd* and *lockd*.) By default, Solaris, HP-UX 10.x and 11.x machines are configured as NFS clients. Solaris systems do not run a *biod* daemon, as support for NFS is built into the kernel.

All systems must be configured manually as NFS servers.

2. Export local file systems on the server to the clients.
3. Mount the file systems exported in step (2) onto the clients in one or more of the following ways: manually (with the *mount* command), at boot time (in the system mount table), or via the automounter.

All machines in the examples that follow will run the automounter to mount at least one directory from another machine.

NOTE: Hostnames used in this chapter have been chosen for clarity and are not actual hostnames.

9.1 Configuring NFS

9.1.1 Solaris

A Solaris system is an NFS client by default. It will become an NFS server if */etc/dfs/dfstab* exists. The */etc/dfs/dfstab* file contains share commands that will be executed at boot time:

```
share -o rw=hpux10:hpux11: solaris,root=hpux10:solaris /export/home
```

After making the appropriate changes to */etc/dfs/dfstab*, reboot, or enable NFS service by executing:

```
solaris# /etc/init.d/nfs.server start
```

The script will export `/export/home` by executing the `shareall` command, and will start the `nfsd` daemons.

NOTE: By default, an exported file system is accessible to everyone within your network. You will want to investigate using the access control flags documented in the `share(1m)` manual page if you want to restrict access to your exported file systems.

9.1.2 HP-UX 10.x /11.x

Reference:

NFS Client Configuration and Troubleshooting

Installing and Administering NFS Services

An HP-UX 10.x system is an NFS client by default. It can become as NFS server either through **SAM** or through modifications to `/etc/rc.config.d/nfsconf`.

Set the following variable to 1:

```
NFS_SERVER=1           # Is this machine a server?
```

Set the number of `nfsd` processes with the `NUM_NFSD` and `nfs bio` with `NUM_NFSIOD` variables:

```
NUM_NFSD=30
```

```
NUM_NFSIO=16
```

NOTE: The number of `nfsd` processes you need depends on several factors. For a technical explanation of how you should choose the appropriate number of `nfsd` processes, refer to Hal Stern's *Managing NFS and NIS*. The full reference is at the front of this chapter.

Add the following line to the `/etc/exports` file.

```
/export/home -rw=hpux10:hpux11:solaris,root=hpux10:solaris
```

Enable the NFS service with the following command:

```
hpux# /sbin/init.d/nfs.server start
```

This script will export the file systems listed in */etc/exports* and start the *nfsd* daemons.

9.2 The Automounter

The following automounter map files will be used for examples in this section of the chapter:

/etc/auto_master:

```
# Master map file
# Directory      Map          NFS Options
/home            auto.home    -rw,intr
```

/etc/auto_home:

```
# Home map file
# Directory      NFS Options <NFS server>:/<directory>
bill            -rw,info          hpux10:/export/home/bill
margo           -rw,info          hpux11:/export/home/margo
molly           -rw,info          solaris:/export/home/molly
shannon         -rw,info          solaris:/export/home/shannon
```

/etc/auto_master defines one or more mount points to be managed by the automounter. In this example, the mount point is */home*. Any access to files under */home* will trigger the automounter daemon to read the *auto.home* file, which in turn will determine how */home/** will be resolved into an NFS mount.

In this example, accessing */home/bill* will cause a mount of *hpux10:/export/home/bill* under */tmp_mnt*. If */home/bill* is unused for five minutes, */home/bill* will be automatically unmounted. The default idle period is five minutes, but an alternative value may be specified with the *-t* command line option.

Note that */etc/aut0_home* is an indirect map file. An indirect map file is the most common type of automounter map file. Direct map files are supported, but direct mounts can cause mount storms and should be avoided.

In these examples, the physical file system on the NFS server where the user's home directory resides should be mounted somewhere other than */home*. The reason for this is that */home* will become unavailable after automounter is started.

/export/home is the defacto standard home directory location, and its use is strongly recommended for Solaris

8 and HP-UX 10.x./11.x

9.2.1 Solaris

Solaris will automatically start the automounter via the commands in the file */etc/rc2.d/S74autofs*. Refer to Chapter 3: *Start-up Configuration Files* for details about start-up configuration files.

The automounter daemon will search for valid automounter maps, depending on how the automount database entry in */etc/nsswitch.conf* is configured. Refer to Chapter 10: *DNS* for information on */etc/nsswitch.conf*. As an example, the database entry should be:

```
automount: files
```

This entry tells the automounter to look in */etc* for the automounter maps.

On Solaris, the automounter maps have been renamed to include an underscore instead of a dot: *auto_master* and *auto_home*. This has been done to avoid a naming conflict with the NIS+ naming scheme. Refer to Chapter 8: *NIS* for information about NIS+.

9.2.2 HP-UX 10.x and 11.x

The automount daemon is started if the *AUTOMOUNT* variable in */etc/rc.config.d/nfsconf* is set to **1**:

```
AUTOMOUNT=1
```

Set *AUTOOFS*=1 to use the new version of automount (supports PV3)

To access an automounter map in */etc*, set the *AUTO_MASTER* variable:

```
AUTO_MASTER="/etc/auto_master"
```

Increase the block transfer size to improve performance:

On the HP client, increase the read and write block sizes from the default value of 8192 to 32768:

In the `/etc/auto_master` file, add the following line:

```
/home auto_home -rsize=32768,wsiz=32768
```

NOTE Do not use tabs — use only spaces. If you use a tab, AutoFS will not work.

Make sure the `/etc/nsswitch.conf` file has the following:

```
automount: files [NOTFOUND=continue] nis
```

9.3 Troubleshooting Hints and Tips

9.3.1 Troubleshooting NFS

If there is trouble with NFS, try mounting the file system on a different host. This will show if the problem is specific to the first host.

Also, the `exportfs` (for solaris use `share`) command with no options will show how all the file systems are exported. This can be a useful diagnostic, as a common error is modifying `/etc/exports` but forgetting to re-export the file systems with the new changes. Here is an example of running `exportfs`:

```
# exportfs
/export/home -access=solaris:hpux10:hpux11
```

On Solaris systems, the `share` command will provide the same information.

9.3.2 Troubleshooting the Automounter

The following tips should help diagnose most automounter problems:

1. Try to perform the mount manually by using the `mount` command:

```
# mount solaris:/export/home/bill /mnt
```

Be sure to use a different local mount point from that managed by the automounter. If the mount is successful, there is probably a problem with the automounter map.

The built-in *-hosts* map is useful for troubleshooting problems with a particular NFS server. Make sure that */net* exists, as */net* is the default mount point for the *-hosts* map. Then, issue the following command:

```
# cd /net/<nfs server>
```

where *nfs_server* is the name of the NFS server from which the file system is to be mounted. This command will cause all file systems exported on the NFS server to be mounted under */net*.

You should be able to locate the file system in which you are interested using either *ls* or *df*. If the file system is mounted, then you have a problem with your automounter maps. If the file system is NOT mounted, there is probably a problem with the */etc/exports* (for solaris; */etc/dfs/dfstab*) entry on the NFS server.

2. An invalid line in an automounter map can cause a mount to fail. To troubleshoot this problem, remove all lines from */etc/auto.master* (for NFS Version 3; */etc/auto_master*) except the entry for the failing mount point. Then, reduce the automounter map (e.g., *auto.home* or *auto_home*) to a bare number of lines. Try the mount again. If the mount works, add one of the removed line and try the mount again. Repeat until the line causing the failure is identified.

NOTE: To stop the automounter daemon, do not send it a **SIGKILL** signal (i.e., *kill -9 <pid>*). Instead, use **SIGTERM** (i.e., *kill <pid>*), which is the default on all the systems covered in this manual; this allows the automounter to exit cleanly.

[\[Next Chapter: 10. DNS\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 10 Domain Name System (DNS)

This chapter covers the use of the Domain Name System (DNS) and the interaction of the DNS and the Network Information Service (NIS) with Solaris, and HP systems. Only the use of the DNS resolver configuration files will be covered, however, as the installation and configuration of DNS name servers and databases are beyond the scope of this manual.

See Chapter 11, Sendmail, for information about setting up sendmail when using DNS.

The Domain Name System maps hostnames and IP (Internet Protocol) addresses. DNS is used by most internetworking services, such as sendmail, and by TCP/IP services such as Telnet and FTP.

DNS specifies the protocol that hosts use to exchange name information.

BIND (the Berkeley Internet Name Domain) is a specific software program which implements the DNS protocol. BIND is by far the most commonly used DNS software for Unix machines and is used by Solaris, HP-UX 10.x and HP-UX 11.x. *named* (*in.named* for solaris) is part of BIND and it is the daemon that services all DNS requests.

Table 10-1 lists which version of BIND is available on each system:

Solaris	HP-UX 10.X	HP-UX 11.X
8.1.2	4.8.3	4.9.7

Table 10-1: BIND versions

These versions were determined by executing the following command:

```
% strings <fully qualified path of named executable> | grep "@(#)named" (HP-UX)
% strings <fully qualified path of named executable> | grep "@(#)in.named" (Solaris)
```

For example:

```
hp-ux_11% strings /usr/sbin/named | grep "@(#)named"
```

@(#)named 4.9.7 Thu Jan 27 17:03:37 GMT 2000 PHNE_20619

The client side of BIND is called the name resolver. It is typically implemented as a separate library (*libresolv.a*) and has replacements for common functions found in *libc* (i.e., *gethostbyname*, *gethostbyaddr*).

The resolver configuration is specified in the file */etc/resolv.conf*. This file can specify the local DNS domain name and the IP address of at least one local host that is running *named*. For example:

```
domain fc.hp.com
nameserver 15.2.72.254
nameserver 15.2.83.254
```

The *domain* keyword specifies the local domain name. The resolver will append this string to DNS queries: if the resulting query fails then the resolver will perform the query without the local domain name.

The *nameserver* keyword specifies the IP address of a single name server (a host running *named*). If multiple name servers are listed in *resolv.conf*, they will be tried in the order listed. A query is sent to the first address listed and if a response is not received within five seconds, a query will be sent to the next name server. In cases where no name server responds, the exact behavior is largely dependent on the operating system, configuration, and application. For more details on the exact behavior of the BIND resolver, see the reference *DNS* and *BIND*.

10.1 Solaris Implementation of Name Resolution

Sun recommends that Solaris be configured to resolve host names using NIS+ and to query DNS if the NIS+ lookup fails. The policy for resolving host names is set using */etc/nsswitch.conf*.

To configure Solaris to use DNS as a backup for NIS+:

1. Create the DNS resolver configuration file */etc/resolv.conf*, and add entries for one or more name servers:

```
solaris# cat /etc/resolv.conf
domain fc.hp.com
nameserver 15.2.72.254
```

2. Modify the hosts entry in */etc/nsswitch.conf* to specify the order in which to query NIS+ and DNS:

```
hosts: nisplus dns [NOTFOUND=return] files
```

This indicates that host names should first be resolved using NIS+. If the resolution fails, then DNS should be queried. If the resolution still fails, then the resolver should return an error indicating that the name did not resolve. If NIS+ and DNS are not available, then */etc/hosts* will be used.

Another way of specifying this is:

```
hosts: nisplus [NOTFOUND=continue] dns [NOTFOUND=return] files
```

The string [NOTFOUND=continue] indicates that if the resolution fails, then the next host name source (e.g., DNS) should be used. This is the default behavior.

10.1.1 Testing the Modifications

1. Test the configuration file by using a host that is configured in the *NIS_host.org_dir* table:

```
solaris# ping hpux
hpux is alive
```

2. Test the host name resolution from NIS+ to DNS by using a host that is NOT in the *host.org_dir* NIS+ table:

```
solaris# ping hprpap.fc.hp.com
hprpap.fc.hp.com is alive
```

3. Verify that the name was resolved via DNS using *nslookup*:

```
solaris# nslookup hprpap.fc.hp.com
Server: hpfcmgw.fc.hp.com
Address: 15.2.72.254

Name:  hprpap.fc.hp.com
Address: 15.2.72.80
```

10.2 HP-UX 10.x/11.x Implementation of Name Resolution

10.2.1 Configuring The Resolver

1. Create an appropriate `/etc/nsswitch.conf` file (sample files are located in `/usr/examples/nsswitch`, for *HP-UX 10.x* and `/etc/nsswitch.*` for *HP-UX 11.x*). The host in this example will query DNS first, then pass any unresolved queries to NIS:

Here is a real world example of an `/etc/nsswitch.conf` file:

```
# /etc/nsswitch.easyssetup
#
# This /etc/nsswitch.conf file uses NIS in conjunction with local files.
# This file was designed with real-world input from existing network
# configurations. This file is installed by Easy Setup. A copy is also
# kept in /etc/nsswitch.easyssetup
#
# Append + to /etc/passwd & /etc/group - Applications compiled on HP-UX 10.20
# or earlier may need it. Note that a + has been appended to passwd/group
# during Easy Setup.
#
# "compat" will allow full use of old NIS + and - syntax in passwd/group
# passwd:    compat
# group:     compat

passwd:     files [NOTFOUND=continue] nis
group:      files [NOTFOUND=continue] nis

# Files first to make sure loopback and local host are found when lan is offline.
# Reverse order of dns and nis if dns is primary or remove dns or nis as needed.
hosts:      files [NOTFOUND=continue] nis [NOTFOUND=continue] dns

# Also try NOTFOUND=continue for some of these if you have any problems
networks:   nis [NOTFOUND=return TRYAGAIN=continue] files
protocols:  nis [NOTFOUND=return TRYAGAIN=continue] files
rpc:        nis [NOTFOUND=return TRYAGAIN=continue] files
publickey:  nis [NOTFOUND=return TRYAGAIN=continue] files
netgroup:   nis [NOTFOUND=return TRYAGAIN=continue] files
aliases:    files [NOTFOUND=continue] nis
```

```
# To make automount use NIS and not an ascii master file change the AUTO_MASTER
# variable to AUTO_MASTER="" and AUTO_OPTIONS variable to AUTO_OPTIONS=""
# in the file /etc/rc.config.d/nfsconf .
# Also change the automount line below to: automount: nis
# Note that HP-UX requires explicitly-passed rsize=32768,wsize=32768 so
# the use of a local /etc/auto_master file is preferred.
```

```
automount: files [NOTFOUND=continue] nis
```

```
# files should be first for services since services may not be in use by other
# hosts and may be incomplete or inaccurate
services: files [NOTFOUND=continue] nis
```

2. Create the resolver configuration file. The *domain* keyword should be used ONLY if you do not want other domains searched. Using the *domain* keyword may cause resolution of partially qualified hostnames to fail.

```
hpux# cat /etc/resolv.conf
search fc.hp.com
nameserver 15.2.72.254
```

10.2.2 Testing DNS

The *server* command changes the policy to DNS and changes the server to the one specified. The *server* command can now be used when *nslookup* is doing either */etc/hosts* or NIS lookups. In these cases, the policy is switched to DNS using the given server, and may be switched back to the default by using the *reset* command:

1. Look up names in the local domain:

```
hpux# nslookup -swtrace hpfcfa
  Initializing defaults for sources
  lookup source is DNS
  DNS Lookup (GetHostInfoBy) Successful
  DNS defaultInfo filled
  defaultAddr set to 15.2.72.254
```

DNS defaultInfo filled

lookup source is NIS

NIS defaultInfo filled

Batch mode Lookup

lookup_type = 0

Name Server: hpfcmgw.fc.hp.com

Address: 15.2.72.254

lookup source is DNS

Name Server: hpfcmgw.fc.hp.com

Address: 15.2.72.254

Name: hpfcla.fc.hp.com

Address: 15.254.48.2

2. If DNS has been configured using the search option in */etc/resolv.conf*, look up a host name in a different domain:

```
hpux# nslookup support.mayfield
```

```
Server: hpfcmgw.fc.hp.com
```

```
Address: 15.2.72.254
```

```
Address: 192.6.148.19
```

```
Aliases: support.mayfield.hp.com
```

Notice that the name is found in the *hp.com* domain instead of *fc.hp.com*.

[\[Next Chapter: 11. Electronic Mail\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 11 Electronic Mail (E-Mail)

[\[Last Chapter: 10. DNS\]](#)

Solaris, HP-UX 10.x and HP-UX 11.x all use *sendmail* as the mail transfer agent (MTA). This is the software which is responsible for the transfer of electronic mail messages to other systems or to programs which deliver the message.

A typical network is configured with a primary mail host which uses NFS to export the electronic mail spool directory to all client workstations. The client *sendmail* is configured to relay all electronic mail through the hub machine. This chapter will cover both client and hub configurations, though in most cases the hub machine will require additional configuration for environments which exchange mail with hosts on the Internet. Organizations that are not going to be using NFS to share mail spool directories should typically configure all machines as mail hubs when using the examples in this chapter. For more information on NFS, refer to Chapter 9, *NFS* and the *Automounter*.

For all examples, hosts will be assumed to belong to the *bar.com* DNS domain.

SECURITY WARNING! Some versions of the *sendmail* daemon have well-known security flaws. Contact the vendor's customer support department for information on patches that may apply. This will assure the most up-to-date *sendmail* fixes available.

11.1 Configuring Sendmail on Solaris

Solaris uses Sendmail Version 8.9.3 and the configuration files (including the aliases file) are located in */etc/mail*.

1. Configure the Solaris hub to export the mail spool directory (which is by default */var/mail*):

```
solaris_hub# /usr/sbin/share -F nfs -o rw=host1:host2:host3 /var/mail
```

Also add this command to */etc/dfs/dfstab* so that the directory will be NFS-exported at boot time.

Note that because the client will forward all mail to the hub, there is no need for root write access by the client. If the mail client is a Solaris system, you should set the file mode access and ownerships on */var/mail* as follows:

This can be done with the following commands:

```
solaris# chmod 1777 /var/mail
```

```
solaris# chmod g+s /var/mail
```

2. Configure the client to mount the mail spool directory. First add the following entry to */etc/vfstab*:

```
mailhub:/var/mail - /var/mail nfs - yes rw,hard,fg
```

Next, mount the spool directory:

```
solaris_client# mount /var/mail
```

3. Copy */etc/mail/main.cf* to */etc/mail/sendmail.cf*. This sendmail configuration file has the functionality to support both the mail hub and client systems.
4. Edit */etc/mail/sendmail.cf* and follow the directions to set the appropriate values for the host name and domain name. For mail routed to the Internet via a gateway, modify the following macros as shown here:

```
# major relay host  
DRmailhub.bar.com  
CRmailhub.bar.com
```

5. If DNS is NOT being used, use the default */usr/lib/sendmail* executable. Make sure that the major relay host is reachable via NIS. Kill the *sendmail* daemon and re-start it using the following command (which can be found in */etc/rc2.d*):

```
solaris# /etc/rc2.d/S88sendmail
```

Test the configuration file using the steps described in Section 11.3.

6. If DNS is being used, save the original sendmail executable and replace it with the */usr/lib/sendmail.mx*:

```
solaris# mv /usr/lib/sendmail /usr/lib/sendmail.orig
```

```
solaris# cp /usr/lib/sendmail.mx /usr/lib/sendmail
```

7. Kill the sendmail daemon and re-start it using the following command:


```
solaris# /etc/rc2.d/S88sendmail stop/start
```

Test the configuration file using the steps described in Section 11. 3.

11.3 HP-UX 10.x and HP-UX 11.x

NOTE: SAM can be used to enable sendmail only.

1. Obtain and install the latest sendmail patch(es).
2. Configure the hub to export the mail spool directory (which is by default */var/mail*). First add the following entry to */etc/exports*:

```
hpux_hub# cat /etc/exports  
/var/mail -access=host1:host2:host3
```

Next, export the file system:

```
hpux_hub# exportfs /var/spool/mail
```

Note that because the client will forward all mail to the hub, there is no need for root write access by the client.

3. Configure the client to mount the mail spool directory. First add the following entry to */etc/fstab*:

```
mailhub:/var/mail /usr/mail nfs rw,hard,fg 0 0
```

Next, mount the spool directory:

```
hpux_client# mount /var/spool/mail
```

4. By default, sendmail is started at system boot time. The file */etc/rc.config.d/mailservs* contains definitions which control how the sendmail daemon is started during the boot process. Make the following modifications to this file:

(client)

```
export SENDMAIL_SERVER=0
export SENDMAIL_SERVER_NAME="mailhub"
```

(hub)

```
export SENDMAIL_SERVER=1
export SENDMAIL_SERVER_NAME=
```

Setting *SENDMAIL_SERVER* to a value of "1" specifies that the sendmail daemon is to be started on this machine. If mail is to be forwarded to a hub, *SENDMAIL_SERVER_NAME* must be set to the hub machine's hostname. The hub machine must have *SENDMAIL_SERVER_NAME* left blank.

Note that if the mail client is a Solaris system, you should set the file mode access and ownerships on */var/mail* as follows:

```
drwxrwsrwt 2 bin mail 512 Apr 2 14:59 mail
```

5. Copy the default sendmail configuration file from */usr/newconfig/etc/mail/sendmail.cf* to */etc/mail/sendmail.cf*. The default configuration file is suitable for sending mail to the hub machine. For more elaborate configurations (such as routing to the Internet), modify */etc/mail/sendmail.cf* by following the instructions at the beginning of the file.

Sendmail can be configured in a variety of ways. While the steps presented here will work for many users, other sites may have needs such as routing certain mail destinations via UUCP and other mail destinations via a firewall to the Internet. Instructions are included at the top of the *sendmail.cf* to assist with customizing *sendmail* as needed.

6. On the hub server, start sendmail:

```
hpux_hub# /sbin/init.d/sendmail stop
hpux_hub# /sbin/init.d/sendmail start
```

This start-up script will rebuild the database of user aliases from */etc/mail/aliases* and will freeze the configuration file via the */etc/rc.config.d/mailservs* file if directed to do so. The default directs sendmail to use a frozen configuration file.

Test the configuration file using the steps described in Section 11.3.

11.4 Testing the Sendmail Configuration

Whenever changes are made to the *sendmail.cf* file, it is advisable to first rename *sendmail.cf*, then execute sendmail with the `-C<file name>` where `<file name>` is the name of the test file. This allows you to safely test a new *sendmail* configuration file on a production system.

If there is Telnet access between hosts, you can connect to the *sendmail* daemon, which uses smtp (Simple Mail Transport Protocol) to communicate.

According to the */etc/services* file, smtp uses port 25.

Telnet to the *sendmail* daemon by using one of the following commands:

```
telnet <hostname> smtp
```

or:

```
telnet <hostname> 25
```

or:

```
telnet <hostname> mail
```

If sendmail is not running on the remote host, you will see:

```
% telnet rocky 25
Trying 146.84.2.51 ...
telnet: connect: Connection refused
telnet>
```

Possibly that system was not configured to start sendmail at boot time. Start sendmail on the remote host.

When sendmail is running, you will see:

```
% telnet rocky 25
Trying 146.84.2.51 ...
```

Connected to rocky.

Escape character is '^']'.

220 rocky.bar.com ESMTP Sendmail 8.8.6 (PHNE_17190)/8.8.6; Tue, 31 Oct 2000 11:06:41 -0700 (MST)

In this Telnet mode, sendmail accepts four-letter commands. A list of valid commands can be displayed with the help command:

help

214-This is Sendmail version 8.8.6 (PHNE_17190)

214-Topics:

214- HELO EHLO MAIL RCPT DATA

214- RSET NOOP QUIT HELP VRFY

214- EXPN VERB ETRN DSN

214-For more info use "HELP <topic>".

214-smtp To report bugs in this implementation or suggestions for

214-smtp future enhancements, contact your HP representative.

214-For local information send email to Postmaster at your site.

214 End of HELP info

First, "introduce" yourself with the *HELO* command:

```
HELO <your host name>
```

For example:

```
HELO foo
```

250 rocky.bar.com Hello foo [foo.bar.com], pleased to meet you

After the introduction, check the mail addresses using the *VRFY* command:

```
VRFY molly@hp.com
```

```
250 <molly@hp.com>
```

This message tells you that the address *molly@hp.com* will not be re-addressed by sendmail.

```
VERFY susan
```

```
250 <susan@bighost.hp.com>
```

In this example, "susan" is expanded to the complete mail address as listed in the mail aliases file. If a mail alias has more than one entry, the expanded entry is displayed:

```
VERFY support
```

```
250-Molly Mayhem <molly>
```

```
250-<rab@bar.hp.com>
```

If mail is bouncing from a particular host, check aliases and mail addresses in the mail aliases file. If a *newaliases* command was not issued after the alias file was changed, or if new aliases were not tested with *sendmail -bt* (see below), there may be a discrepancy between what *sendmail* "thinks" the aliases are and the aliases actually in the mail aliases file.

If there is no telnet access to the remote system, log onto the system remotely (or ask the administrator of that machine for assistance). The administrator of the remote machine can either use the process described to check the local host, or send you test mail.

Also, determine if the problem is isolated-occurring only between these two machines-or if it occurs under other circumstances. Can the local system send mail to other hosts? Is the host name being resolved properly?

To run *sendmail* in debug mode, the syntax is:

```
/usr/lib/sendmail -bt -d21.XX
```

where XX is the debugging level. Valid level values are 00-99, where 00 is no debugging and 99 is the highest level. The higher the debugging level, the more output produced by *sendmail*, so it is advisable to choose a relatively low number (around 20) to start. With debugging turned on, you can see the token matches. (This procedure is recommended only if enough information CANNOT be obtained from *sendmail* with debugging turned off.)

You can also run */usr/ucb/mail* in verbose mode. The output follows the same format as that used by *sendmail* in Telnet mode.

Another test is to run *sendmail* in verbose mode:

```
/usr/lib/sendmail -v < /dev/null rab@hp.com
```

This command will send an empty message to the user "rab" at the DNS domain *hp.com*. The output from *sendmail* will show whether mail forwarding is working correctly:

```
% /usr/lib/sendmail -v < /dev/null root@tcd170.fc.hp.com
root@tcd170.fc.hp.com... Connecting to hpfcla.fc.hp.com. via esmtp...
220 hpfcla.fc.hp.com ESMTP Sendmail 8.9.1/8.9.3 SMKit7.01; Tue, 31 Oct 2000 11:58:29 -0700 (MST)
>>> HELO tcd162.fc.hp.com
250-hpfcla.fc.hp.com Hello tcd162.fc.hp.com [15.2.75.162], pleased to meet you
250-EXPN
250-VERB
250-8BITMIME
250-SIZE 25000000
250-DSN
250-ONEX
250-ETRN
250-XUSR
250 HELP
>>> MAIL From:<root@tcd162.fc.hp.com>
250 <root@tcd162.fc.hp.com>... Sender ok
>>> RCPT To:<root@tcd170.fc.hp.com>
250 <root@tcd170.fc.hp.com>... Recipient ok
>>> DATA
354 Enter mail, end with "." on a line by itself
>>> .
250 LAA00444 Message accepted for delivery
root@tcd170.fc.hp.com... Sent (LAA00444 Message accepted for delivery)
Closing connection to hpfcla.fc.hp.com.
>>> QUIT
221 hpfcla.fc.hp.com closing connection
```

```
rab@hp.com... Connecting to mailhub.bar.com via ether...
Trying 198.252.182.129... connected.
220 mailhub.bar.com Sendmail 4.1/SMI-4.1 ready at ready at Thu, 2 Nov 1995
09:44:34 -0600 (CST)
>>> HELO titmouse.hp.com
250 mailhub.bar.com Hello titmouse.bar.com [198.252.182.18], pleased to meet
```

you

```
>>> MAIL From:<dinah@titmouse.bar.com>
```

```
250 <rab @hp.com>
```

Notice that the sendmail daemon on the client workstation connects directly to the local mail hub (*mailhub.bar.com*), instead of trying to send directly to *hp.com*. Try all combinations of mail addresses:

Local addresses: <user>

Site-wide addresses: <user>@bar.com

Off-site addresses: <user>@<domain>

[\[Next Chapter: 12. Printing\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 12 Printing

[\[Last Chapter: 11. Electronic Mail\]](#)

This chapter covers local and remote printing on the Solaris, and HP systems, and includes explanations of how to configure remote and local printers.

12.1 Solaris

12.1.1 Adding a Local Printer With admintool

1. Physically connect the printer to the serial port A on host "solaris."
2. Start *admintool*:

```
solaris# /usr/bin/admintool
```

3. Select "*Browse, Printers*"
4. Select "*Edit*."

Select "*Add*"

Select "*Add Local Printer*."

Admintool: Add Local Printer

Printer Name: hpljet

Print Server: tcd170

Description:

Printer Port: /dev/term/a

Printer Type: PostScript

File Contents: PostScript

Fault Notification: Write to superuser

Options: Default Printer
 Always Print Banner

User Access List: all

Add **Delete**

OK **Apply** **Reset** **Cancel** **Help**

The "*Admintool: Add Local Printer*" dialog will be displayed. In this example, admintool is instructed to add a local printer named "hpljet." The HP Laserjet in this instance has PostScript capabilities. Set "*Default Printer:*" to "Yes" if this printer is to be the system default. Complete the dialog and select "OK."

-

NOTE: The Solaris printer daemon can accept either System V or BSD print protocols when receiving remote printing requests from other systems. Therefore, when configuring a local printer, there is no need to specify which protocol to use. A printer installed as above can be the target of remote print requests from either BSD- or System V-based systems. Future releases of Solaris will no longer support BSD print protocols.

5. Check the status of the printer and test the printer:

```
solaris# lpstat -phpljet
solaris# lp -d hpljet /etc/motd
```

NOTE: This example uses the System V-style printer management commands. Generally, these commands work better than the BSD-style commands that Solaris provides. If possible, use System V-style commands to print to, and to manage, Solaris printers.

12.1.2 Adding a Local Printer via the Command Line

1. Connect the printer to the system.
2. Set the permissions on the printer device. For a printer connected to serial port A:

```
solaris# chown lp /dev/term/a
solaris# chmod 600 /dev/term/a
```

3. Use lpadmin to add the printer device. In this example, "laser" is the printer's class, and the printer supports PostScript. A printer class is used when printing to a class of one or more printers:

```
solaris# lpadmin -v /dev/term/a -p hpljet -c laser
solaris# lpadmin -p hpljet -T PS
solaris# lpadmin -I postscript -p hpljet
```

4. Allow queuing to the printer:

```
solaris# accept hpljet
```

5. Allow printing to the printer:

```
solaris# enable hpljet
```

6. Check the status of the printer:

```
solaris# lpstat -p hpljet
```

7. Test the printer:

```
solaris# lp -d hpljet /etc/motd
```

12.1.3 Adding a Remote Printer With admintool

This example configures a Solaris system to access a remote printer on a HP-UX 11.0 system.

1. Start *admintool*:

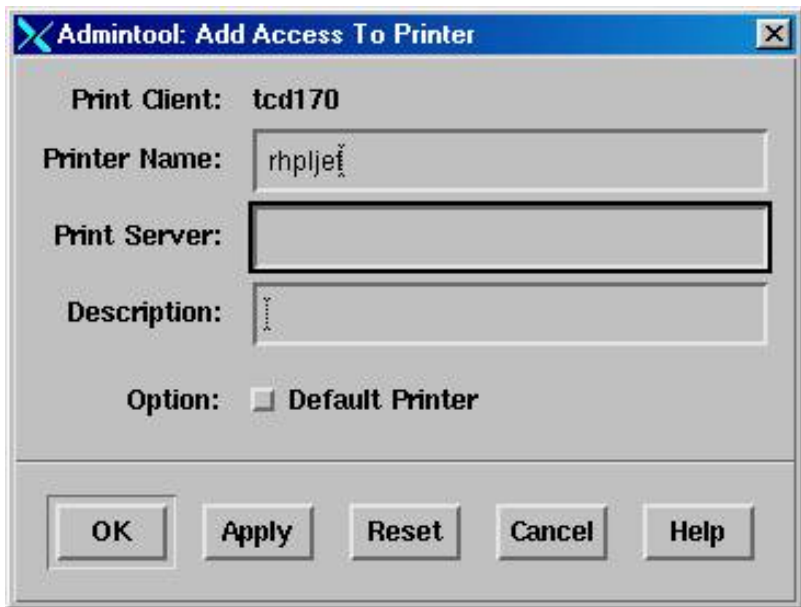
```
solaris# admintool
```

2. Select "*Browse, Printers.*"
3. Select "*Edit.*"

Select "*Add*"

Select "*Admintool Access to Printer.*"

The "*Admintool: Add Access to Printer*" dialog will be displayed:



4. Once the dialog is complete, select "OK"
5. Check the status of the printer:

```
solaris# lpstat -p rhpljet
```

6. Test the printer:

```
solaris# lp -d rhpljet /etc/motd
```

12.1.4 Adding a Remote Printer via the Command Line

Generally, use the admintool to add printers. When adding many printers, or when adding printers as part of a JumpStart script, it is necessary to be able to add printers via the command line. This section is useful for those circumstances.

This example configures a Solaris system to access a remote printer on a HP-UX 11.0 system.

1. Use lpadmin to designate the remote system.

```
solaris# lpadmin -p printer2 -s hp11! Printer1
```

2. Allow queuing to the printer:

```
solaris# accept rhpljet
```

3. Allow printing to the printer:

```
solaris# enable rhpljet
```

4. Check the status of the printer:

```
solaris# lpstat -t
```

5. Test the printer:

```
solaris# lp -d rhpljet /etc/motd
```

12.2 HP-UX 10.x and HP-UX 11.x

12.2.1 Adding a Local Printer With SAM

SAM is the easiest way to configure a printer because SAM will also initialize the LP spooler. The screen shots shown here are from an HP-UX 10.x, but the steps are identical on an HP-UX 11.x.

1. Configure with SAM:

- a. Select "*Printers and Plotters.*"

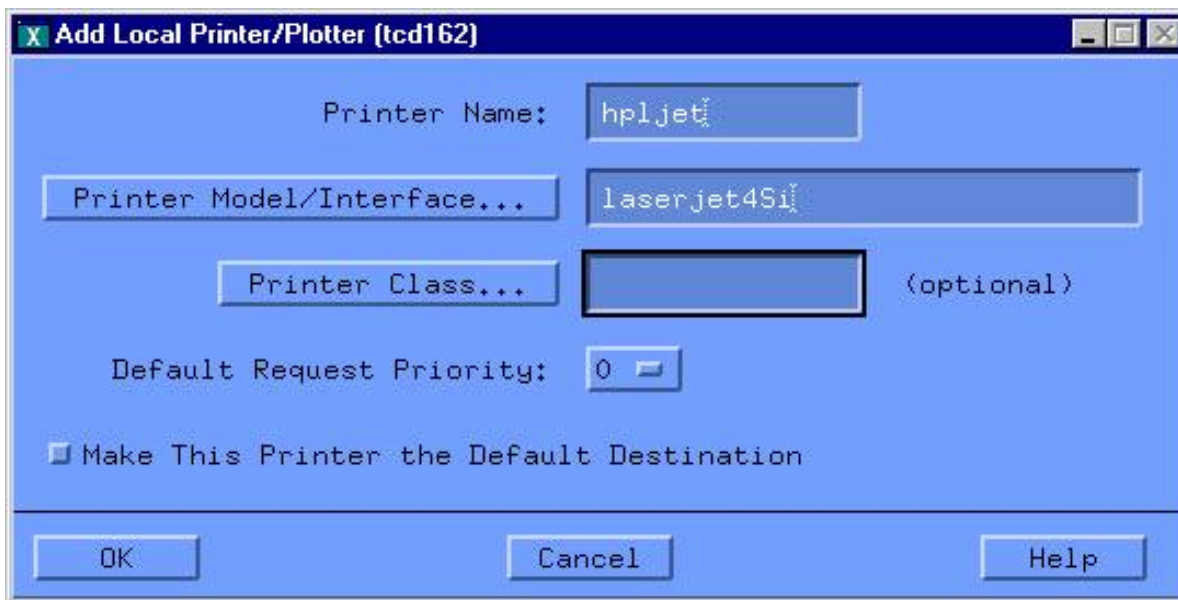
- b. Select "*LP Spoolers.*"

- c. Select "*Printers/Plotters*"

- d. From the "*Actions*" menu, select "*Add Local Printer/Plotter.*"

- e. Select "*Add Serial (RS-232) Printer/Plotter.*"

The "*Add Serial (RS-232) Printer/Plotter*" dialog appears. Select the appropriate serial interface and select "*OK.*" The "*Add Local Printer/Plotter*" dialog appears:



Complete the dialog and select "OK." Note that this dialog screen has available all supported models through the "*Printer Model/Interface...*" field. The model files are located in `/usr/lib/lp`. These files describe the characteristics supported by the printer. You may add your own model files to this directory.

If there are no printer classes defined, the value entered into the printer class field will be used to create a new printer class. The printer class may be used instead of the name of the printer. The print job will be queued to the next available printer. This feature allows printers with similar characteristics to be grouped; users can print their jobs faster by directing the jobs to a group of printers than by queuing them to a single printer.

2. Test your configuration by printing a file. The first line is an example using the printer model; the second line is an example using the print class name:

```
hpux# lp -dhpljet /etc/hosts
```

```
hpux# lp -dpostscript /etc/hosts
```

12.2.2 Adding a Remote Printer With SAM

1. Use SAM to set up remote printer access to a remote printer on host "solaris":

```
hpux# sam
```

- a. Select "*Printers and Plotters.*"

- b. Select "*Printers and Plotters.*"

c. Select "LP Spooler."

d. From the "Actions" menu, select "Add Remote Printer/Plotter."

The "Add Remote Printer/Plotter" dialog will be displayed. Enter the information as shown:

The screenshot shows a window titled "Add Remote Printer/Plotter (tcd162)". The window has a blue background and contains the following fields and controls:

- Printer Name: rhpljet
- Remote System Name: solaris
- Remote Printer Name: rhpljet
- Remote Printer is on a BSD System
- Remote Cancel Model...: rcmode1
- Remote Status Model...: rsmode1
- Default Request Priority: 0
- Allow Anyone to Cancel a Request
- Make This Printer the Default Destination
- Buttons: OK, Cancel, Help

Once the information has been entered, select "OK."

2. Test your configuration by printing a file:

```
hpux# lp -drhpljet /etc/hosts
```

You can print directly to the printer or the printer class.

12.2.3 Adding a Printer Manually

1. Stop the LP spooler:

```
hpux# lpshut
```

2. Add the printer to the LP spooler. In this example, the printer device is */dev/lp* and the model file is *laserjet*:

```
hpux# lpadmin -phpljet -v/dev/lp -mlaserjet
```

(Be sure to use the appropriate model file for the printer.)

3. Allow print requests to be accepted:

```
hpux# accept hpljet
```

This command allows the LP spooler to accept print requests for this printer.

4. Enable printing:

```
hpux# enable hpljet
```

This command allows the LP spooler to actually print the job.

5. Restart the LP spooler:

```
hpux# lpsched
```

6. Test the printer. The first line is an example using the printer model; the second line is an example using the print class name:

```
hpux# lp -dhpljet /etc/hosts
```

```
hpux# lp -dpostscript /etc/hosts
```

[\[Next Chapter: 13. Common Login Environments\]](#)

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 NOV 00)

Copyright 2000 Hewlett-Packard Company. All rights reserved.

HP-UX <> Solaris System Administrator Task Guide

Chapter 13 Common Login Environments

[\[Last Chapter: 12. Printing\]](#)

The network file system examples presented in [Chapter 9](#), *NFS and the Automounter*, demonstrate a method of distributing a user's home directory across all machines in a network. For this distribution to be successful, a login environment must be created which is compatible with each machine type.

This chapter presents the information necessary to create a common login environment that will work on all systems. Sample scripts for both the Korn shell and the C shell are included.

13.1 Shells

The Bourne shell, the C shell, and the Korn shell are common to all operating systems. There are several freely-distributable shells (e.g., *tcsh* and *bash*) that will work on all systems, but these shells should be compatible with those discussed in this chapter and are not referenced specifically.

HP-UX 10.x and 11.x both include the POSIX shell which is located in */bin/posix/sh* and */usr/bin/sh*, respectively. The POSIX shell includes much of the functionality of the Korn shell. Therefore, in most cases the POSIX shell and the Korn shell (*ksh*) are interchangeable.

The examples in this chapter will be based on the Korn shell, but the concepts can be applied easily to other shells.

13.2 Login Process

In Solaris and HP-UX 10.x/11.x, when a new user is created, the contents of the */etc/skel* directory are copied into the user's home directory.

The login process is complex. A simplified view of the login process for each system is shown in Figures 13-1 and 13-2:

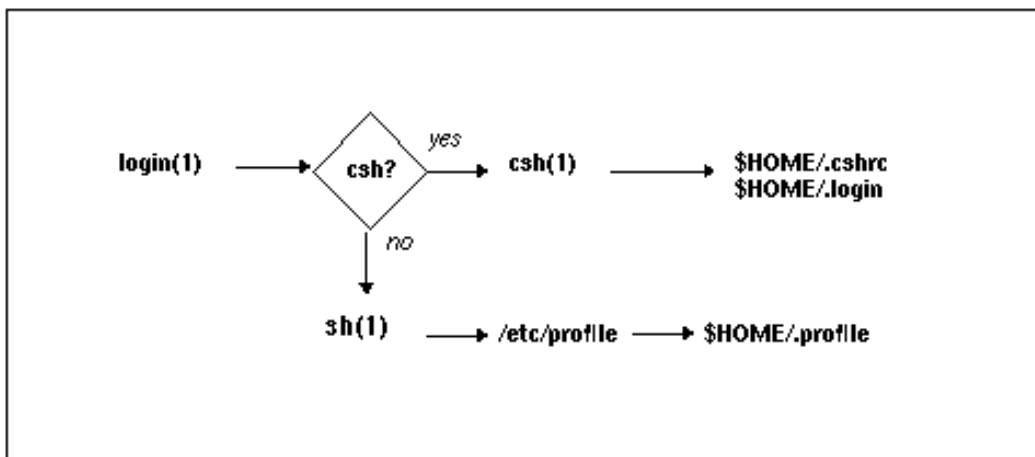


Figure 13-1: Solaris Login Process

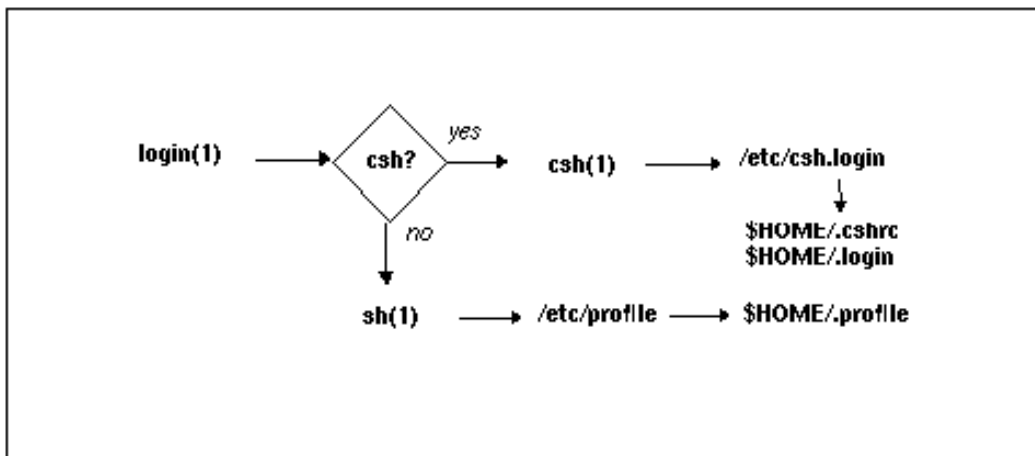


Figure 13-2: HP-UX 10.x and 11.x Login Process

Note that HP-UX 10.x and 11.x support a centralized `/etc/csh.login` file for C shell users. All platforms will allow you to run `$HOME/.cshrc` on the creation of all new `csh` processes, while `$HOME/.login` is executed only once on initial login. This allows you to separate commands for terminal configuration into `$HOME/.login` so processes that do not have a terminal associated with them (such as `rsh` and `rcp` commands) will work properly.

A more detailed description of the HP-UX login process is shown in Figure 13-3:

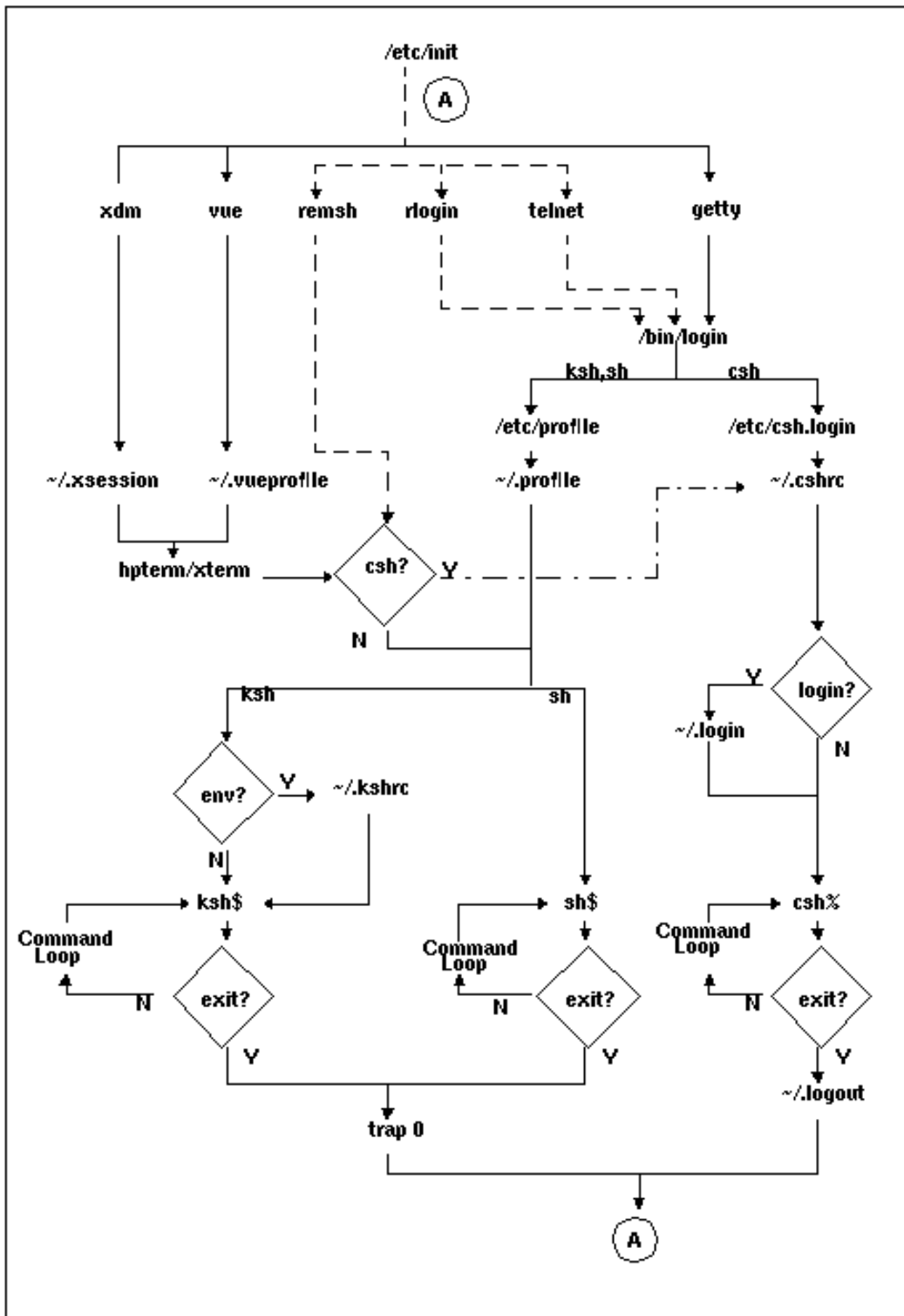


Figure 13-3: Detailed HP-UX 10.x and 11.x Login Process

A summary of the login configuration files for Solaris and the corresponding files used in HP-UX 10.x and 11.x are shown in Table 13-1:

Solaris	HP-UX 10.X	HP-UX 11.X
/etc/skel/local.login	/etc/csh.login	/etc/csh.login
\$HOME/.login	\$HOME/.login	\$HOME/.login
\$HOME/.dtprofile (CDE)	\$HOME/.dtprofile (CDE)	\$HOME/.dtprofile (CDE)
/etc/skel/local.cshrc	/etc/profile	/etc/profile
\$HOME/.dtprofile (CDE)	\$HOME/.profile	\$HOME/.profile
	\$HOME/.dtprofile (CDE)	\$HOME/.dtprofile (CDE)
\$HOME/.x11start	\$HOME/.x11start	\$HOME/.x11start
\$HOME/.X11defaults	\$HOME/.X11defaults	\$HOME/.X11defaults
	\$HOME/.dt/sessions	\$HOME/.dt/sessions
/{current home}/dt.resources	/{current home}/dt.resources	/{current home}/dt.resources
\$HOME/.mwmrc		
\$HOME/.dt/dtwmrc	\$HOME/.dt/dtwmrc	\$HOME/.dt/dtwmrc

Table 13-1: Login Configuration Files**13.3 Concepts and Strategies**

The key to implementing a common login environment is understanding when and how the shell

start-up scripts are processed. Typically, there are three different scripts which need to be developed to provide a common environment across systems for POSIX and Korn shell users:

1. */etc/profile*
2. *\$HOME/.profile*
3. *\$ENV*

13.3.1 Understanding Login Scripts

The shell can be invoked as either a login shell or as a sub-shell. The difference is which start-up scripts are processed. The POSIX and Korn shells use these start-up scripts:

- */etc/profile* - System-wide configuration file. Typically contains system-wide configuration information only, and is executed upon login.
- *\$HOME/.profile* - User configuration file. Contains system and user settings, and is executed only on login. Usually contains information that is passed from one shell to another.
- *ENV* - If this environment variable is set, shell configuration information is read from the file it points to. Typically *ENV* is set to *\$HOME/.shrc* and contains command aliases and function definitions.

All invocations of the shell will read configuration information from the *\$ENV* file if the *ENV* environment variable is set. Non-login shells (also known as sub-shells) get configuration information **ONLY** from the *\$ENV* file and from the environment inherited from the parent process. This environment is usually another shell which has exported some of its variables.

Note that */etc/profile* is not run when using *rlogin*. *rlogin* bypasses the normal login sequence, which is where most of the environment setup is done (including setting the location of the shell configuration file).

Login shells are identified by a "-" as the first character of *echo \$0*. The shells first load configuration information from

/etc/profile. Next, *\$HOME/.profile* is loaded if it exists. If the environment variable *ENV* is set after the reading of *\$HOME/.profile*, the file it references (usually *\$HOME/.shrc*, or *\$HOME/.kshrc* for the Korn shell) is run.

The system configuration script */etc/profile* is typically only used for displaying copyright information, showing the message of the day, setting a default *PATH*, and other activities that are "required" on login. This is also a good place to put common settings which may be the same across all users. Beware of doing more than environment setup, however! Users cannot customize this file and therefore can be limited by its contents. In particular, this file should NOT try to use *tset* which may generate escape sequences that lock up certain terminal types.

The user login start-up script *\$HOME/.profile* must set up everything for a new shell and its sub-shells. This means that values which would normally be passed from shell to shell (such as exported environment variables, or terminal settings) are set only once. Anything that would be run only when first logging in (such as displaying a to-do list) would also be placed here.

The shell configuration file, specified by the environment variable *\$ENV* (which is typically exported in *\$HOME/.profile*), is read for every new shell that is started. Values that do not propagate from shell to shell, or that are different from shell to shell, are placed here. For example, functions and aliases are not passed to sub-shells and belong

here. If these settings are missing from the *\$ENV* file when you start a shell from within an editor session, their values will not be available later. Also, if the prompt has been defined to contain user information (such as a trailing # for the root user) that you expect to propagate through a *su*, it will need to be in this file.

13.4 Developing Common Login Scripts

The four major activities required to create a common login environment are:

1. identifying the system type,
2. defining the command and manual page search paths,
3. providing common aliases across machines, and
4. setting up the appropriate terminal types.

Most of the other environment settings will be the same on all machines.

13.4.1 Identifying the System Type

The *uname* command is used to identify the system type and OS release level.

First, identify the type of system using *uname* with no options.

The first example assumes a Solaris system and the second an HP-UX system:

```
% uname
```

```
Solaris
```

```
% uname
```

```
HP-UX
```

Next, identify the OS release using the `-r` option.

Output from a Solaris system:

```
% uname -r
```

```
5.8
```

The release information output from two different versions of HP-UX is as follows:

```
% uname -r
```

```
B.10.20
```

```
% uname -r
```

```
B.11.00
```

Since you are usually interested only in the major release number, you can combine `uname` with `awk` to extract this information:

```
% uname -r | awk -F. '{print $2}'
```

The first activity is to identifying the operating system and its version and store the values into shell variables. Conditional statements can then be built which depend on the values stored in the variables. These variables must be set in `$HOME/.profile` or `/etc/profile`.

13.4.2 Defining the Search Paths

There are two types of search paths commonly used: command and manual page. The examples in this section address the command-search path. The same concepts apply to the manual page search path, which is specified with the `MANPATH` environment variable.

A good strategy for setting the command search path, which is specified with the `PATH` environment variable, is to first set

system-dependent paths, and then set system-independent paths. For example:

```
case $SYSTYPE in
  HP-UX)
    export PATH=/usr/bin:/usr/sbin:/usr/bin/X11
  fi
;;
```

<lines deleted>

```
export PATH=$PATH:/usr/local/bin:/site/bin
```

Notice how the system-dependent paths are set within the case statement. The directories common to all systems are added to the path later in the script.

An example of common search paths for each operating system is shown in Table 13-2:

OS Version	Command Search Path
Solaris	/bin:/sbin:/usr/bin:/usr/openwin/bin:/etc:/usr/etc: /usr/ccs/bin:/usr/local/bin:/usr/ucb
HP-UX 10.X	/usr/dt/bin:/usr/bin/X11:/usr/bin:/sbin:/usr/ccs/bin: /usr/contrib/bin:/usr/local/bin:/usr/contrib/bin/X11:/opt/lrom/bin
HP-UX 11.X	/usr/dt/bin:/usr/bin/X11:/usr/bin:/sbin:/usr/ccs/bin: /usr/contrib/bin:/usr/local/bin:/usr/contrib/bin/X11:/opt/lrom/bin

Table 13-2: Common Search Paths

13.4.3 Providing Common Aliases

The following commands can be troublesome when you are moving between Solaris, HP-UX 10.x and HP-UX 11.x: *ps*, *df*, and *ls*. Each command requires different options to produce the same information. Therefore, you may want to "alias" the commands to get consistent output.

For example, On HP-UX systems you may want to alias "bdf" to "df -k"

```
% alias "df -k=bdf"
```

Another example of a command you may wish to alias is *rsh*. On HP-UX systems, the command to execute a remote shell is *remsh*, while on Sun systems the command is *rsh*. Depending on personal preference, you may choose to alias *remsh* to *rsh* (or vice versa.)

Be careful when aliasing commands. Some administrators prefer to rename commands when using aliases. This makes troubleshooting easier, since there is no need to distinguish between problems with an alias and problems with a real command. However, to make things transparent for a user, you may want to use the name of the command for the alias (as shown in the example above).

Remember that you can use "*whence*" and "*alias*" to display the values of aliases. When troubleshooting a user's environment wherein a problem is reported with a command, these commands (*whence* and *alias*) can be used to ensure that there is not an incorrect alias defined:

```
solaris% alias
```

```
bdf    df -k
```

```
hp-ux% alias
```

```
rsh    remsh
```

Note that *whence* is a built-in POSIX and Korn shell command.

You can also preface a command with a backslash (\) which will prevent the shell from executing the alias. The actual command will be executed instead or use the full pathname to the command.

13.4.4 Setting the Correct Terminal Type

Setting the correct terminal type involves two main issues: terminal emulation and keypad mapping.

13.4.4.1 Terminal Emulation

When there are problems with terminal emulation, the problems are usually caused by vendors using proprietary emulators such as Sun's *commandtool* or HP's *hpterm*. For example, when you remotely log into an HP-UX system using a *commandtool* window, the *sun-cmd* terminal type is unrecognized on the HP system.

A simple solution to this problem is to specify a terminal type of *sun1* on the HP-UX system. Add the following lines to the HP-specific portion of the login script:

```
if [ "$TERM" = "sun-cmd" ]; then
    export TERM=sun1
```

fi

A more permanent solution to the problem is to add the correct sun-cmd entry into the *terminfo* database.

Then, copy the sun-cmd.info file to the HP system and load the file into the terminfo database:

```
hp% tic sun-cmd.info
```

This command creates a sun-cmd terminfo entry in the file */usr/lib/terminfo/s/sun-cmd*. This file then can be copied to other HP-UX systems. You may copy terminfo data files only between identical systems. You may not copy a file from an HP-UX system to a Sun system.

Conversely, HP-UX users using an hpterm will find that their hpterm is not defined on Solaris systems.

Once again the simple solution is to specify a terminal type of hp:

```
solaris% export TERM=hp
```

To create a valid entry on Solaris systems, retrieve a copy of the *terminfo* definition using the *untic* command:

```
hp% untic hpterm > hpterm.info
```

Copy this file to the Solaris system and issue the following command:

```
solaris% /usr/bin/tic hpterm.info
```

This command will compile the terminfo file and store the results in */usr/share/lib/terminfo/h/hpterm*.

13.4.4.2 Key Mapping

When you bring up an x-term or hpterm on an HP-UX system, the following are default settings: (Common equivalence values shown in parentheses.)

Erase # (Delete or Backspace key)

Suspend ^@ (^Z)

Interrupt DEL (^C)

Kill @ (^U)

You can redefine these settings with the *stty* command:

```
% stty erase ^H susp ^Z intr ^C kill ^U
```

These default settings can be redefined in either the system-dependent or system-independent portion of the login script.

13.5 Miscellaneous Settings

The following sections describe miscellaneous settings that you may want to consider for your common login scripts.

13.5.1 umask Settings

By default, on both HP-UX 10.x and 11.x, *umask* is set to 0. This gives world read/write access to files unless you specifically set the file access modes to a different setting. The default setting on Solaris is 2, which allows world read access, but prohibits world write access.

13.5.2 Autologout and Tmout

The C shell, Korn shell, and POSIX shell have an autologout feature. For C shell users, if the "autologout" environment variable is set to a number other than 0, the shell will be logged out after the specified number of minutes have passed. For Korn shell and POSIX shell users, the *TMOU*T environment variable specifies the number of minutes before the shell logged out. A value of 0 indicates the shell should not be logged out.

13.5.3 Destructive Backspace

If you find your backspace is non-destructive (i.e., the ERASE character does NOT get echoed as a backspace-space-backspace string), then you should execute the following command:

```
hpux% stty echoe
```

This will correct the problem.

13.6 CDE Login Environment

Reference:

CDE User's Guide

CDE prints `/etc/copyright` at start-up. This is a good place to put information that would traditionally go into `/etc/motd`.

If a user logs in using the CDE windowing environment, the `$HOME/.dtprofile` script executed instead of the standard start-up files (e.g., `/etc/profile` and `$HOME/.profile`). This is to avoid problems in case the scripts run terminal I/O-based commands such as `tset`.

For increased maintainability, it is best to have settings in one location only. In the case of CDE, this means that the `.profile` should be modified so that terminal I/O routines are excluded when executed from the CDE environment. This can be done by setting the `$CDE` variable to "true" within `$HOME/.dtprofile`:

```
export DTSOURCEPROFILE=true
```

Now, check for the CDE environment variable in `$HOME/.profile`:

```
if [ ! "$DT" ]
then
  stty erase ^H intr ^C
  eval `tset -Ae` # ask terminal type and set environment
fi
```

Errors in `.vueprofile` or in `.profile` may prevent a successful login. If this happens, log in using the "fail-safe" menu item from the VUE login screen and correct the error.

13.7 Sample Script

A sample .profile follows. Note the system-dependent and the system-independent sections of the script. (Lines should be added as appropriate for the site.)

```
#!/bin/ksh

export SYSTYPE=`uname`
case $SYSTYPE in
  HP-UX)
    export OSVER=`uname -r | awk -F. '{print $2}'`
    if [ "$TERM" = "sun-cmd" ]; then
      export TERM=sun1
    fi
    case $OSVER in
      10)
        export PATH=/usr/dt/bin:/usr/bin/X11:/usr/bin:/sbin:/usr/ccs/bin:\
          /usr/contrib/bin:/usr/local/bin:/usr/contrib/bin/X11:/opt/lrom/bin
        ;;
      11)
        export PATH=/usr/dt/bin:/usr/bin/X11:/usr/bin:/sbin:/usr/ccs/bin:\
          /usr/contrib/bin:/usr/local/bin:/usr/contrib/bin/X11
        ;;
    esac
    ;;
  Solaris)
    export OSVER=`uname -r`
    if [ "$TERM" = "hpterm" ]; then
      export TERM=hp
    fi
    case $OSVER in
      4.*)
        export PATH=/usr/ucb:/bin:/usr/bin:/usr/openwin/bin:/usr/local/bin:\
          /etc:/usr/etc:/usr/5bin
        alias "ps=ps -aux"
        ;;
    esac
  ;;
```

5.*)

```
export PATH=/bin:/sbin:/usr/bin:/usr/openwin/bin:/etc:/usr/etc:\
    /usr/ccs/bin:/usr/local/bin:/usr/ucb
;;
esac
esac
# System-independent portion of script
umask 002
export TMOUT=0
export PATH=$PATH:/usr/local/bin
stty erase ^H susp ^Z intr ^C kill ^U
if [ -z "$LINES" ] ; then
    export LINES=`tput lines`
fi
if [ -z "$COLUMNS" ] ; then
    export COLUMNS=`tput cols`
fi
```

Provide feedback to: cookbook@apollo.hp.com

[\[Table of Contents\]](#) [\[Outline\]](#) [\[Index\]](#) (Updated 11 OCT 96)

Copyright 2000 Hewlett-Packard Company. All rights reserved.