Information Technology Institute

Red Hat System Administration II

Open Source Applications Development

Information Technology Institute

KERNEL SERVICES AND CONFIGURATION

Kernel Modules

- Many of the kernel's components are not part of the standard kernel image, but are dynamically loadable modules.
- These modules increase the kernel's functionality without increasing the size of the kernel image loaded at boot time.
- A smaller kernel image results in a faster boot process as well as less memory use.

The /lib/modules directory

- Kernel modules reside in the /lib/modules/<kernel-version> directory.
- Kernel version can be obtained by uname –r command.

Listing and unloading modules

- The kernel normally loads modules automatically when they are required without any human intervention.
- While rarely needed, there are commands that will
  - List the kernel modules that are loaded into memory -> lsmod, modeprobe
  - Load kernel modules into memory -> modprobe
  - Remove kernel modules from memory -> rmmod, modprobe
  - See what each module will do -> modinfo

Examples

- Unload the vfat module using rmod command

 # lsmod | grep fat

vfat 14529 0

fat 44257 1 vfat

 # rmmod vfat

 # lsmod | grep fat

fat 44257 0

- Loading vfat module using modprobe command

 # lsmod | grep fat

#

 # modprobe vfat

vfat 14529 0

fat 44257 1 vfat

- Unload the vfat module using modprobe command

 # modprobe --remove vfat

 # lsmod | grep fat

#

Loading modules

- These module dependencies are defined in the /lib/modules/<kernelversion>/modules.dep file, which is generated by the depmod command.
- You seldom need to run the depmod command manually because the default /etc/rc.d/rc.sysinit script is configured to run the depmod command when the system is booted.

Configure kernel modules

- When you load a module with the modprobe command, the /etc/modprobe.conf configuration file is consulted for appropriate default values.

more /etc/modprobe.conf

alias etho eloo # tells the modprobe command to load the eloo module if the etho device is activated.

alias snd-card-o snd-intel8xo # tells the modprobe command to load the snd-intel8xo module if the snd-card-o device is activated.

options snd-card-o index=o # tells the modprobe command to provides the index=o option to the kernel when the snd-card-o module is loaded.

Update a kernel RPM

- To install an updated kernel, use rpm -i to install an additional kernel. Then, test the new kernel
- Only after you are convinced that the new kernel is working properly will you want to delete the older kernel
- To delete an older kernel

rpm -e kernel-<*version*>

The /proc filesystem

- It contains a virtual filesystem that has information about the running kernel, such as system hardware, network settings and activity, and memory usage.
- Features of the /proc filesystem
  - It is not a disk-based filesystem; all of the data is stored in memory.
  - It is mounted during system boot-up through an entry in the /etc/fstab file.

The /proc filesystem (cont.)

- Listing the files and directories under /proc will reveal that virtually all of them have a size of zero, but you can cd into the directories and can view files by cat

Some of the key files

- /proc/cpuinfo
  - Information about the system's CPU
- /proc/meminfo
  - Information on avialable memory, free memory, swap, cached memory, and buffers
- /proc/uptime
  - System uptime and idle time
- /proc/version
  - Information on Linux kernel version, host, date , etc

Some of the key Directories

- /proc/scsi
  - Information about SCSI devices
- /proc/ide
  - Information about IDE devices
- /proc/net
  - Information about network activity and configuration
- /proc/sys
  - Kernel configuration parameters
- /proc/$PID
  - Information about process PID

Examples

 # echo 1 > /proc/sys/net/ipv4/ip_forward # Turn on IP forwarding

 # echo 16384 > /proc/sys/fs/file_max

- /proc/sys modification are temporary and not saved at system shutdown
- Kernel configuration file is /etc/sysctl.conf

- To change any parameter in /etc/sysctl.conf
  - For example: net.ipv4.ip_forward = 1

    # sysctl -p
  - then check /proc/sys

    # cat /proc/sys/net/ipv4/ip_forward

FILESYSTEM MANAGEMENT

Adding a new disk

- If you added a new disk the system should detect its presence after reboot.
- You would be able to see this discovery in the /var/log/dmesg log file.
- The tool that partitions disks is called fdisk.
- You use the device name of the disk as the argument.
- Only the root account can adjust disk partition settings.

Starting the fdisk Utility

- **# fdisk /dev/hda**
- To show the menu of command options, type m.
  - Delete a partition -> d
  - Lists known partition types -> l
  - Adds a new partition -> n
  - Prints the partition table -> p
  - Quits without saving changes -> q
  - Writes table to disk and exit -> w

Did Kernel Feel The Changes ?

- If you cat /proc/partitions
  - The new partition is not available
- So for the kernel to feel the changes use partprobe command.

Formatting a partition

- The format utility is generally known as mkfs

 # mkfs –t ext2 /dev/hda5

 # mkfs –t ext3 /dev/hda5

Create a mount point

- Create a mount point for this partition

  # mkdir /rdbm

- Mount a partition to a directory

  # mount -t ext2 /dev/hda5 /rdbm

- Disconnect a filesystem from mount point with umount

  # umount /dev/hda5

  # umount /rdbm

Adding Additional Partitions to /etc/fstab

- At boot-up time, the rc.sysinit command reads the fstab file to determine which filesystems should be mounted.

 # more /etc/fstab

```
# DEV LOCAL-ACCESS FILE-TYPE OPTIONS   DUMP FSCK-ORDER
/dev/hda2      /        ext3    defaults  1        1
/dev/hda1   /boot       ext3    defaults   1       2
/dev/hda3   swap        swap    defaults   0       0
```

Filesystem Labels

 # e2label /dev/hda8 mydisk

 # mount LABEL=mydisk /data1

 # mount -L mydisk # in case it is present in /etc/fstab

 # e2label /dev/hda8

mydisk

- Using labels in /etc/fstab

  LABEL=mydisk /data1 ext3 defaults 1 1

ADDING ADDITIONAL SWAP TYPE DISK SPACE

Introduction

- Swap space allows processes to use more memory than actually exists on the system

- If the amount of memory requested by the process running on the system exceeds the amount of available RAM, the Linux kernel can swap some of the pages of memory being used by sleeping or idle processes to disk to make room for the additional memory needed by running processes or new processes.

Swap summary usage with the swap command

- You can display your current swap usage using the –s option to the swapon command

 # swapon -s

Filename	Type	size	Used	Priority
/dev/hda2	partition	2040244	0	-1

Adding a new swap type partition

- Add a new swap partition
  - Using the fdisk utility to create a partition
  - Set the system ID to the value hex 82
    - Use the t command within fdisk to change a partition's system ID to Linux swap
    - Save the changes
  - Use partprobe to force the system to recognize the changes.

- Format swap partition

 # mkswap /dev/hda6

Implement and Display swap Partition Usage Information

- Use the swapon utility
  - To begin using the device as swap space.
- Use the -s option
  - To display the swap usage summary information
- Add the new swap partition to the /etc/fstab file

Example

```
[root ~]# fdisk /dev/hda

Command (m for help): n
First cylinder (3001-4864, default 3001):
Using default value 3001
Last cylinder or +size or +sizeM or +sizeK (3001-4864, default 4864): +199

Command (m for help): t
Partition number (1-6): 6
Hex code (type L to list codes): 82
Changed system type of partition 6 to 82 (Linux swap)

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource
 busy.
The kernel still uses the old table.
The new table will be used at the next reboot.
Syncing disks.
[root ~]# partprobe
```

Example cont'd

```
# mkswap /dev/hda6
    Setting up swapspace version 1, size = 1645019 KB
# swapon /dev/hda6
# swapon -s
    Filename        Type        Size        Used    Priority
    /dev/hda2    partition    2040244        0        -1
    /dev/hda6    partition    1606460        0        -2
# grep swap /etc/fstab
    /dev/hda2  swap    swap    defaults    0    0
# vi /etc/fstab
    /dev/hda6  swap    swap    defaults    0    0
```

Adding Temporary File Space as swap

- Create a local file of the required size using the dd command.
- Format this file just as if it were a partition device file.

Example cont'd

 # dd if=/dev/zero of=/myswap bs=1024 count=1024

1024+0 records in

1024+0 records out

 # ls -l /myswap

-rw-r--r-- 1 root root 1048576 Jul 6 06:37 /myswap

 # chmod 600 /myswap

 # mkswap /myswap

Setting up swapspace version 1, size = 1044 kB

 # swapon /myswap

 # swapon –s

Filename	Type	Size	Used	Priority
/dev/hda2	partition	327672	2868	-1
/myswap	file	1016	0	-2

 # vi /etc/fstab

/myswap	swap	swap	defaults	0	0

TAPE DEVICES/DRIVES

Controlling Tape Drives

- The `mt` command is the general-purpose command that manipulates tapes. It is used to assist the backup process.

- Some of the options for `mt`
  - `rewind`: Rewinds a tape
  - `offline` : Prepares the currently loaded tape for ejection and, if possible, ejects it
  - `fsf`: Moves the currently loaded tape to the specified position
  - `erase`: Erases the currently loaded tape

Controlling Tape Drives cont'd

- The `mt` command syntax is:

```
mt -f device command
```

- To specify the device, use the `-f` option followed by the desired target
  - The standard SCSI tape devices are named `st0`, `st1`, etc., and `nst0`, `nst1`, etc
  - The standard IDE tape devices are named `ht0`, `ht1`, etc., and `nht0`, `nht1`, etc

Controlling Tape Drives cont'd

- Tape drive normally rewinds the media after the tape operation has completed. If you don't want the tape to rewind, you can access the device by its no rewind name.

- Examples

```
# mt -f /dev/nst0 fsf 50   // Positions the tape (don't forget to use "n").
# mt -f /dev/st0 rewind    // Rewinds the tape.
# mt -f /dev/st0 offline   // Ejects the tape (but doesn't rewind it first).
# mt -f /dev/st0 rewoff    // Rewinds and ejects the tape.
# mt -f /dev/st0 erase     // Erases the tape.
```

Using `tar/star` Commands

- Archive to tapes or other media or files
- `star` command backups `SELinux` contexts and `ACL` attributes.
- Options
  - `c`: To create new archive
  - `t`: To list the content of existing archive
  - `x`: To extract existing archive
  - `v`: verbose
  - `z`: gzip compress
  - `j`: bzip2 compress

Examples

```
# tar cf /dev/st0 fname
# tar zcf /dev/st0 fname
# tar zxf /dev/st0
```

* Where does `tar` extract the files ???

Incremental and Full Back Ups

- A full backup is a complete file system backup.
- An incremental backup copies only files in the file system that have been added or modified since a previous lower-level backup.
- Backup increment = dump level

Level	Definition
0	Full Backup
1-9	The backup copies new or modified files since the last lower-level backup

Using dump Command

- It can backs up filesystems. By providing the mount point of the filesystem to back up.

- Example

```
# dump -0u -f /dev/nst1 /home
# dump -4u -f /dev/nst1 /home
```

Recovering dump Data

- To recover an entire filesystem

```
# restore -rf /dev/st0
```

- To restore individual files and directories

```
# restore -xf /dev/st0 file1 file2
# restore -if /dev/st0
```

Name: Sherine Bahader
Email: sbahader@gmail.com