

1.105.1

Manage/Query kernel and kernel modules at runtime

Weight 4

Linux Professional Institute Certification — 102

Angus Lees gus@inodes.org

Nick Urbanik nicku@nicku.org

This document Licensed under GPL—see section 8

2005 July

Outline

Contents

1	Context	2
2	Objectives	2
3	Documentation	2
4	Kernel Modules	3
5	Utilities	3
5.1	uname	3
5.2	lsmod	3
5.3	modprobe	4
5.4	modinfo	4
6	/etc/modules.conf	4
7	Low level commands	5
8	License	5

1 Context

Topic 105 Kernel [7]

1.105.1 Manage/Query kernel and kernel modules at runtime [4]

1.105.2 Reconfigure, build, and install a custom kernel and kernel [3]

2 Objectives

Description of Objective

Candidates should be able to manage and/or query a kernel and kernel loadable modules. This objective includes using command-line utilities to get information about the currently running kernel and kernel modules. It also includes manually loading and unloading modules as appropriate. It also includes being able to determine when modules can be unloaded and what parameters a module accepts. Candidates should be able to configure the system to load modules by names other than their file name.

Key files, terms, and utilities include:

/lib/modules/kernel-version/modules.dep — the file built by `depmod` showing which module depends on what other modules

/etc/modules.conf & /etc/conf.modules — configuration file for `modprobe`

depmod — determines module dependencies and rebuilds `modules.dep`

insmod — low level command to insert a module into running kernel

lsmod — list modules loaded into kernel

rmmod — low level command to remove a module from kernel

modinfo — shows details about a particular module

modprobe — the tool for inserting and removing modules into/from the running kernel

uname — shows information about the system, including the kernel version info.

3 Documentation

Manage/Query Kernel and kernel modules at runtime

- `<linux-source>/Documentation/`

4 Kernel Modules

Kernel Module Basics

- Kernel modules are chunks of kernel code which can get loaded and unloaded at run-time.
- Some modules depend on code in other modules.
- Good for shipping pre-built kernels – but not loading all available code; good for changing between conflicting modules (eg. OSS and ALSA), etc.

5 Utilities

5.1 uname

uname – Who the kernel thinks it is

Modules are stored in `/lib/modules/$(uname -r)/kernel/`

- a print all information
- s kernel name (“Linux”)
- n “node name” (hostname)
- r kernel release (“2.4.20-3-686”)
- v kernel version (compile time)
- m machine hardware name (“i686”)
- o operating system (“GNU/Linux”)

5.2 lsmod

lsmod – Currently loaded modules

lsmod uses `/proc/modules` to show you which kernel modules are currently loaded.

Module	Size	Used by
bsd_comp	5888	1
ppp_async	10624	1
ppp_generic	29072	6 bsd_comp, ppp_async
slhc	7040	1 ppp_generic
ipv6	230624	10
ds	14596	0
pcmcia_core	62688	1 ds

irda	174400	0
parport_pc	34088	1
lp	10560	0
parport	40552	2 parport_pc, lp

5.3 modprobe

modprobe – Loading and removing modules

Load a module and its dependencies:

```
$ sudo modprobe <modulename> [<module options>] ←
eg: $ sudo modprobe ftape ft_fdc_base=0x123 ←
```

Unload a module if it's unused:

```
$ sudo modprobe -r <modulename> ←
eg: $ sudo modprobe -r ftape ←
```

5.4 modinfo

modinfo – Module options

“modinfo ftape” gives:

```
parm:          ft_fdc_base:Base address of FDC controller.
parm:          ft_fdc_irq:IRQ (interrupt channel) to use.
parm:          ft_fdc_dma:DMA channel to use.
parm:          ft_fdc_threshold:Threshold of the FDC Fifo.
:
author:        (c) 1993-1996 Bas Laarhoven (bas@vimec.nl), (c) 1995-1996 Kai Harreki
description:   QIC-117 driver for QIC-40/80/3010/3020 floppy tape drives.
license:       GPL
```

6 /etc/modules.conf

modprobe maintenance

Configuration in `/etc/modules.conf` (aka `/etc/conf.modules`).

```
# bogus example
options ftape ft_fdc_base=0x123 fg_fdc_irq=5 ft_fdc_dma=1
alias char-major-27 ftape
```

depmod builds `modules.dep`, describing module dependencies.

7 Low level commands

Loading modules - low-level

Low level commands to actually insert or remove a module:

```
insmod <filename> <module options>  
rmmod <modulename>
```

You will probably never use these directly; use `modprobe` or `modprobe -r` instead.

8 License

License Of This Document

Copyright © 2005, 2003 Angus Lees <gus@inodes.org> and Nick Urbanik <nicku@nicku.org>.

Permission is granted to make and distribute verbatim copies or modified versions of this document provided that this copyright notice and this permission notice are preserved on all copies under the terms of the GNU General Public License as published by the Free Software Foundation—either version 2 of the License or (at your option) any later version.