**Linux troubleshooting**

**Inspecting system and hardware components**

The Linux shell represents an interface to the Linux kernel. By using GNU developed programs in the shell (command interpreter) an administrator can easily inspect the system and perform changes to it.

First of all, when you have connected the Linux system and logged in you will see a shell prompt, like the following:

[root@Oma.lan:/root]#

The first command that you are tempted to type may be the **help** command, which will show the type of the shell you are using, and how to get information for a certain command:

[root@Oma.lan:/root]# **help**

GNU bash, version 3.2.39(1)-release (i486-pc-linux-gnu)

These shell commands are defined internally. Type `help' to see this list.

Type `help name' to find out more about the function `name'.

Use `info bash' to find out more about the shell in general.

Use `man -k' or `info' to find out more about commands not in this list.

Linux operating system includes many commands and you have to get used to learn about them and their paramaters on the fly. The ways to get information about a certain command is by using **man** <command> **; man –k** <keyword-for-your-command>**; <info>** <command> **; whatis** <command> **; apropos** <command>; <command> **--help**. All of these commands are GNU programs that have been compiled and so became binary files. The binary files are usually separated and located as follows:

- *binary files* (ex: ls, cd, pwd, cat, date,…) that can be executed by users

/bin

/usr/bin

/usr/local/bin

- *system binary files* (ex: init, fsck, mkfs,...) that can be executed by root

/sbin

/usr/sbin

/usr/local/sbin

Next you will see some basic commands for inspecting your system (some of then support paramaters):

**dmesg** – print the system’s boot up messages.

This output is placed in /var/log/dmesg at boot time.

**lspci** – list the PCI adapters identified by the kernel

**lsusb** – list the USB adapters identified by the kernel

**dmidecode** – list the description of system hardware components

**uname** – prints system information

[root@Oma.lan:/root]# **uname -a**

Linux Oma.lan 2.6.26-2-686 #1 SMP Thu Nov 25 01:53:57 UTC 2010 i686 GNU/Linux

**uptime** – prints system uptime

[root@Oma.lan:/root]# **uptime**

12:24:00 up 10 days, 11:05, 1 user, load average: 0.02, 0.04, 0.00

**w** – show who is logged in and what they are doing

**who** – show who is logged in

[root@Oma.lan:/root]# **who -uH**

NAME LINE TIME IDLE PID COMMENT

puccu pts/0 2011-02-13 12:00 . 14742 (86.125.248.46)

**runlevel** – show the system’s current runlevel

**id** – who user id

[root@Oma.lan:/root]# **id**

uid=0(root) gid=0(root) groups=0(root)

**date** – show system date and time

**pwd** – print current working directory

**cd** <newdir> – change directory

**ls** <dir> – list directory contents

**mkdir** <newdir> – create a new directory

**rmdir** <dir> – remove a directory

**cp** <source> <dest> - copy files/directories

**mv** <source> <dest> - move/rename files/directories

**rm** <file> - remove a file

**cat** <file> - print the contents of a file

**vmstat** – reports virtual memory statistics

**free** – displays the amount of free/used memory

**top** – displays Linux tasks (similar to Windows Task Manager)

**history** – shows the lines typed in the shell

**du** – show disk usage of a file or directory

**df** – report file system disk space usage

**cat /proc/cpuninfo** – view CPU information

**cat /proc/meminfo** – view Memory information

Now some examples:

[root@Oma.lan:/]# **pwd**

/

[root@Oma.lan:/]# **cd /root**

[root@Oma.lan:/root]# **pwd**

/root

[root@Oma.lan:/root]# **cd /tmp**

[root@Oma.lan:/tmp]# **ls**

hsperfdata\_root testfile vmhsdaemon-0 vmware-root

[root@Oma.lan:/tmp]# **ls -alh**

total 40K

drwxrwxrwt 8 root root 4.0K 2011-02-13 12:28 .

drwxr-xr-x 24 root root 4.0K 2011-02-02 22:16 ..

drwxr-xr-x 2 root root 4.0K 2011-02-03 01:19 hsperfdata\_root

drwxrwxrwt 2 root root 4.0K 2011-02-03 01:18 .ICE-unix

-rw-r--r-- 1 root root 2.5K 2011-02-03 01:19 .license.cfg

-rw-r--r-- 1 root root 14 2011-02-13 12:28 testfile

drwx------ 2 root root 4.0K 2011-02-03 01:21 vmhsdaemon-0

drwx------ 2 root root 4.0K 2011-02-03 01:21 vmware-root

drwxr-xr-x 2 root root 4.0K 2011-02-03 01:19 .webmin

drwxrwxrwt 2 root root 4.0K 2011-02-03 01:18 .X11-unix

[root@Oma.lan:/tmp]# **cat testfile**

Hello world !

[root@Oma.lan:/tmp]# **rm testfile**

[root@Oma.lan:/tmp]#

[root@Oma.lan:/tmp]# **mkdir test**

[root@Oma.lan:/tmp]# **ls**

hsperfdata\_root test vmhsdaemon-0 vmware-root

[root@Oma.lan:/tmp]# **rmdir test**

[root@Oma.lan:/tmp]# **free -m**

total used free shared buffers cached

Mem: 1899 1843 56 0 127 1490

-/+ buffers/cache: 225 1673

Swap: 996 14 981

[root@Oma.lan:/tmp]# **cp /etc/pro**

profile protocols

[root@Oma.lan:/tmp]# **cp /etc/profile /tmp**

[root@Oma.lan:/tmp]# **ls**

hsperfdata\_root profile vmhsdaemon-0 vmware-root

[root@Oma.lan:/tmp]# **mv profile profile\_new**

[root@Oma.lan:/tmp]# **ls**

hsperfdata\_root profile\_new vmhsdaemon-0 vmware-root

[root@Oma.lan:/tmp]# **date**

Sun Feb 13 12:44:04 EET 2011

[root@Oma.lan:/tmp]#

[root@Oma.lan:/tmp]# **runlevel**

N 2

[root@Oma.lan:/tmp]#

It is very important to understand the difference between and absolute and relative path. An absolute path relates to the / (root directory), but an relative path relates to the current directory. Example:

Absolute Path:

[root@Oma.lan:/]# **pwd**

/

[root@Oma.lan:/]# **cd /var/log/apache2/domains/**

[root@Oma.lan:/var/log/apache2/domains]# pwd

/var/log/apache2/domains

Relative Path:

[root@Oma.lan:/var/log]# **pwd**

/var/log

[root@Oma.lan:/var/log]# **cd apache2/domains/**

[root@Oma.lan:/var/log/apache2/domains]# pwd

/var/log/apache2/domains

You may be wondering, how is the kernel made that it can support such a big number of hardware architectures and vendor specific implementation. The response to this question comes from **HAL** (Hardware Abstraction Layer) service. As you can see in the following figure HAL service is an interface between hardware and the kernel, and its purpose is to adapt the hardware to a form that the kernel can interact with.

|  |
| --- |
| KERNEL |
| HAL |
| HARDWARE |

Another wondering question is how does Linux kernel detect and create device files under /dev directory. The response to this question comes from **udevd** service. This service receives notices from the kernel to create/remove/modify a certain device. To list the current devices configured in udev database use **udevinfo** command.

Examples:

root@Oma.lan:/dev]# **ls -alh | grep sd**

lrwxrwxrwx 1 root root 4 2011-02-03 01:18 root -> sda1

brw-rw---- 1 root disk 8, 0 2011-02-03 01:18 sda

brw-rw---- 1 root disk 8, 1 2011-02-03 01:18 sda1

brw-rw---- 1 root disk 8, 2 2011-02-03 01:18 sda2

brw-rw---- 1 root disk 8, 3 2011-02-03 01:18 sda3

brw-rw---- 1 root disk 8, 5 2011-02-03 01:18 sda5

brw-rw---- 1 root disk 8, 16 2011-02-03 01:18 sdb

brw-rw---- 1 root disk 8, 17 2011-02-03 01:18 sdb1

brw-rw---- 1 root disk 8, 18 2011-02-03 01:18 sdb2

brw-rw---- 1 root floppy 8, 32 2011-02-05 23:33 sdc

[root@Oma.lan:/dev]# **udevinfo -e | less**

.........................................................................................

P: /block/sda

N: sda

S: block/8:0

S: disk/by-id/ata-WDC\_WD1600AAJS-00L7A0\_WD-WMAV29582640

S: disk/by-id/scsi-SATA\_WDC\_WD1600AAJS-\_WD-WMAV29582640

S: disk/by-path/pci-0000:00:08.0-scsi-0:0:0:0

E: ID\_VENDOR=ATA

E: ID\_MODEL=WDC\_WD1600AAJS-0

E: ID\_REVISION=01.0

E: ID\_SERIAL=SATA\_WDC\_WD1600AAJS-\_WD-WMAV29582640

E: ID\_SERIAL\_SHORT=WD-WMAV29582640

E: ID\_TYPE=disk

E: ID\_BUS=scsi

E: ID\_ATA\_COMPAT=WDC\_WD1600AAJS-00L7A0\_WD-WMAV29582640

E: ID\_PATH=pci-0000:00:08.0-scsi-0:0:0:0

……………………………………………………………………

From the examples above you can see the character “|” which represents a **PIPE**. A pipe is used to redirect the output from the command to the input of another command. In the first example the output returned by “ls -alh” was redirected to another command, **grep**, which is a filter. In this case grep filters and posts every line that contains the characters “sd”.

**Examples of basic troubleshooting commands within Ubuntu Linux and how to use them**

The following lists provides basic text commands within Ubuntu Linux and provides examples of how they can be used to help troubleshoot specific issues with your computer.

1. Use the command **uname** to show what kernel is being used. The kernel is the first section of the operating system to load into memory, and controls such system areas as disk drive management, memory allocation, system processes, and interrupt handler. In the example below, the kernel being used is **3.11.0-15-generic**:

user@avalon:~$ uname -a  
Linux avalon 3.11.0-15-generic #23-Ubuntu SMP Mon Dec 9 18:17:04 UTC 2013 x86\_64 x86\_64 x86\_64 GNU/Linux  
user@avalon:~$

1. Use the command **dmesg** to show the contents of the boot log. This is a good way to find errors at startup:

user@avalon:~/linux101$ sudo dmesg > dmesg.log

In the example below, the command **cat** can be used with **dmesg** to join together the contents of the log file. The additional command **grep** can be used to show any instance of a particular file name occurring in the log, for example either **Intel** or **error**:

user@avalon:~/linux101$ cat dmesg.log | grep intel  
[ 1.732400] intel\_idle: MWAIT substates: 0x21120  
[ 1.732402] intel\_idle: v0.4 model 0x3A  
[ 1.732405] intel\_idle: lapic\_timer\_reliable\_states 0xffffffff  
[ 10.148411] fbcon: inteldrmfb (fb0) is primary device  
[ 10.922434] i915 0000:00:02.0: fb0 inteldrmfb frame buffer device  
[ 10.926448] snd\_hda\_intel 0000:00:1b.0: irq 50 for MSI/MSI-X  
user@avalon:~/linux101$

user@avalon:~/linux101$ cat dmesg.log | grep error  
[ 7.478502] EXT4-fs (sda2): re-mounted. Opts: errors=remount-ro  
user@avalon:~/linux101$

1. The command **lspci** lists all devices found on the PCI bus:

user@avalon:~/linux101$ sudo lspci > lspci.log

Using the **lspci** command with the additional commands of **cat** and **grep** will help to isolate specific PCI devices, as in the case of **Realtek** below:

user@avalon:~/linux101$ cat lspci.log | grep Realtek  
07:00.0 Ethernet controller: Realtek Semiconductor Co., Ltd. TRL8111/8168/8411 PCI Express Gigabit Ethernet Controller (rev 07)  
09:00.0 Unassigned class [ff00]: Realtek Semiconductor Co., Ltd. RTS5209 PCI Express Card Reader (rev 01)  
09:00.1 SD Host controller: Realtek Semiconductor Co., Ltd. RTS5209 PCI Express Card Reader (rev 01)  
user@avalon~/linux101$

1. Use the command **lsmod** to list all the loaded mod files within the kernel:

user@avalon:~/linux101$ sudo lsmod > lsmod.log  
user@avalon:~linux101$

In Linux, the driver files are loaded as mods. To see which driver/mods are loaded in the kernel, you can use the **cat** and **grep** command with **lsmod** to search for specific mod files, as in the two examples with **Dell** and **hda** below:

user@avalon:~/linux101$ cat lsmod.log | grep dell  
dell\_wmi 12761 0  
sparse\_keymap 13948 1 dell\_wmi  
dell\_laptop 17369 0  
dcdbas 14847 1 dell\_laptop  
wmi 19070 1 dell\_wmi  
user@avalon:~/linux101$

user@avalon:~/linux101$ cat lsmod.log | grep hda  
snd\_hda\_codec\_hdmi 41117 1  
snd\_hda\_codec\_realtek 55704 1  
snd\_hda\_intel 48171 3  
snd\_hda\_codec 188738 3 snd\_hda\_codec\_realtek,snd\_hda\_codec\_hdmi,snd\_hda\_intel  
snd\_hwdep 13602 1 snd\_hda\_codec  
snd\_pcm 102033 3 snd\_hda\_codec\_hdmi,snd\_hda\_codec,snd\_hda\_intel  
snd\_page\_alloc 18710 2 snd\_pcm,snd\_hda\_intel  
snd 69141 17 snd\_hda)codec\_realtek,snd\_hwdep,snd\_timer,snd\_hda\_codec\_hdmi,snd\_pcm,snd\_seq,snd\_rawmidi,snd\_hda\_codec,snd\_hda\_intel,snd\_seq,device,snd\_seq\_midi  
user@avalon:~/linux101$

1. Use the command **lsusb** to list all the USB devices connected to the USB bus:

user@avalon:~/linux101$ sudo lsusb > lsusb.log  
user@avalon:~linux101$

You can use this command with the additional **cat** and **grep** commands to isolate specific USB devices, as in the case of **Intel** below:

user@avalon:~/linux101$ cat lsusb.log | grep Intel  
Bus 002 Device 003: ID 8087:07da Intel Corp.  
Bus 002 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub  
Bus 001 DEvice 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hib  
user@avalon:~/linux101$

1. Use the **ifconfig** command to set and query your computer's network settings:

user@avalon:~/linux101$ ifconfig > ifconfig.log  
user@avalon:~/linux101$

You can use this command with the additional **cat** and **grep** commands to search for **inet**, which will list your computer's IP addresses:

user@avalon:~/linux101$ cat ifconfig.log | grep inet  
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
inet addr:192.168.1.90 Bcast:192.168.1.255 Mask:255.255.255.0  
inet6 addr: fe80::caf7:33ff:fedb:b2bc/64 Scope:Link  
user@avalon:~/linux101$

**A list of basic troubleshooting commands and their function within Ubuntu Linux**

The following table lists some of the more basic commands used within Linux for troubleshooting purposes, as well as giving the function and syntax for each:

| **Command** | **Function** | **Syntax** |
| --- | --- | --- |
| sudo | Used before a command to run as root, or administrator. | sudo apt-get update |
| ls | Same as "dir"; lists the current directory. | ls-ll |
| cp | Copy file. | cp /dir/filename /dir/filename |
| rm | Delete file. | rm /dir/filename /dir/filename |
| mv | Move file. | mv /dir/filename /dir/filename |
| mkdir | Make a directory. | mkdir /dirname |
| df | Report file system disk space usage. | df -h |
| dmesg | Print or control the kernel ring buffer. | dmesg |
| lspci | Lists all PCI devices. | lspci |
| lsusb | Lists all USB devices. | lsusb |
| lsmod | Shows the status of modules in the Linux kernel. | lsmod |
| cat | Concatenate files and print on the standard output. | cat /dir/logfile |
| grep | Print lines of input matching a specified pattern. | grep intel |
| apt-get | Update installer. | apt-get update, or apt-get upgrade |
| sosreport | A utility that collects configuration and diagnostic information about your system. The utility needs to be installed first using the following command line: "sudo apt-get install sosreport". It is recommended you reboot your system after the install before first running the utility. | sosreport |
| "cat" and "grep" together | Use to list all the instances of a specific search item. | cat /dir/logfile | grep intel (the output of this command would print to the default output source any instance of the string "intel") |

Usefull links for system recovery:

Basic System Recovery

<https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/ap-rescuemode.html>

<https://www.centos.org/docs/5/html/Installation_Guide-en-US/ch-rescuemode.html>

Recover from a failed Linux boot

<http://www.ibm.com/developerworks/library/l-GRUB2-features/>

Boot Linux Grub Into Single User Mode

<https://www.cyberciti.biz/faq/grub-boot-into-single-user-mode/>