**5 Practical Linux fuser Command Examples**

The fuser utility in Linux is a powerful tool. As the name suggests it gives information about file user or the process that is currently using the file or directory.

But fuser functionality is not just limited to giving information about the process. The article explains how to use fuser utility with 5 practical examples.

**1. Who is Using a File or Directory?**

This is the basic use of fuser command. i.e to Identify which processes are using a particular file or directory.

$ fuser  .

./:                   3965c  4175c  4281c  4334c  4337c

In the above example we used the fuser utility to find all the processes using the current directory ‘./’ .

We see that the output consists of process IDs of the processes using fuser but all the PIDs are followed by a character ‘c’. This indicates the type of access. The type of access can be any one of the following:

* c      current directory
* e      executable being run
* f      open file. f is omitted in default display mode
* F      open file for writing. F is omitted in default display mode
* r      root directory
* m      mmap’ed file or shared library

So ‘c’ in the output would mean that these processes are using this directory as their current directory.

Use Option -v to display detailed information in the output:

$ fuser -v ./

USER        PID ACCESS COMMAND

./:       himanshu   3965 ..c..    bash

himanshu   4175 ..c..    gedit

himanshu   4281 ..c..    bash

himanshu   4334 ..c..  socket\_serv

himanshu   4337 ..c..    bash

So we see above that running fuser on the current directory gives the information on all the processes that are using this directory.

**2. fuser on an Executable**

socket\_serv is a C program executable which is a TCP server that listens on a particular port.

$ ./socket\_serv

When you execute fuser on this executable, you’ll see the following:

$ fuser -v socket\_serv

USER        PID    ACCESS       COMMAND

socket\_serv:         himanshu   4334    ...e.        socket\_serv

The access specifier in this example is ‘e’. Its different from the access specifier we saw in above examples. It conveys that the file is an executable.

**3. Check Processes Using TCP/UDP Sockets**

Using fuser we can also check the processes using TCP/UDP sockets. Since the above stated socket\_serv sample C program executable is running on TCP port 5000, lets use fuser utility on this socket.

$ fuser -v -n tcp 5000

USER        PID ACCESS COMMAND

5000/tcp:            himanshu   4334   F....  socket\_serv

So we see that fuser gives all detailed information of the process running on TCP port 5000.

Other than the examples above, we can use the ‘-m’ flag with this utility to display processes using a mounted file system like a USB drive.

**4. Kill Processes that are Using a particular Program**

Till now we learned that fuser provides information on the processes using files, directories, sockets etc. But the power of this utility is not restricted to providing information only. You can also kill processes using  this utility.

We saw that a TCP server is running on the system which is accessing the binary file ‘socket\_serv’. Now, lets try to kill the process using this binary file using fuser.

$ fuser -v -k socket\_serv

USER        PID ACCESS COMMAND

socket\_serv:         himanshu   4334 ...e.   socket\_serv

Notice that we have used the ‘-k’ flag to kill the process using file ‘socket\_serv’. Lets see on the other terminal where the server was running.

$ ./socket\_serv

Killed

We already explained you how you can [kill a process using 4 different methods](http://www.thegeekstuff.com/2009/12/4-ways-to-kill-a-process-kill-killall-pkill-xkill/). Now you know one more method to kill a process.

**5. Interactively Kill Processes using fuser**

In the above example we saw that the flag ‘-k’ is used when we want to kill the processes using a particular file but to avoid the killing of processes accidentally, another option ‘-i’ exists. If we use this option then ‘fuser’ will run in interactive mode and will ask before killing the process. See the example below.

$ fuser -v -k -i socket\_serv

USER        PID ACCESS COMMAND

socket\_serv:         himanshu   5643 ...e.   socket\_serv

Kill process 5643 ? (y/N) y

So we see that using ‘-k’ and ‘-i’ we can selectively kill processes using a particular file.

Now, this is a very powerful use of ‘fuser’ command.

Suppose you want to delete a file forcefully but it is being used by many processes then the processes won’t let you delete the file. In that case, you can use fuser utility to kill all the processes (or selected processes) that are using that file.

$ fuser -v -k -i ./

USER        PID ACCESS COMMAND

./:       himanshu   3965 ..c..    bash

himanshu   4175 ..c..    gedit

himanshu   4281 ..c..    bash

Kill process 3965 ? (y/N) y

Kill process 4175 ? (y/N) y

Kill process 4281 ? (y/N) y

Note that the use of ‘-k’ sends a SIGKILL signal to the processes using that particular file. We can use the option -SIGNAL to send any other signal. The list of signals supported by fuser is given by: $ fuser -l

HUP INT QUIT ILL TRAP ABRT IOT BUS FPE KILL USR1 SEGV USR2 PIPE ALRM TERM

STKFLT CHLD CONT STOP TSTP TTIN TTOU URG XCPU XFSZ VTALRM PROF WINCH IO PWR

For example, the following sends SIGHUP instead of SIGKILL, when you use the -k option.

$ fuser -v -k -HUP -i ./

# Linux: HowTo Check User Password Expiration Date and Time

[/etc/shadow](http://www.cyberciti.biz/faq/understanding-etcshadow-file/) files stores actual password in encrypted format for user’s account. How do I check users password expiry information under Linux operating systems?  
  
You need to use the chage command. It can display password expiry information as well as changes the number of days between password changes and the date of the last password change. This information is used by the system to determine when a user must change his/her password.

## View account again information:

To see account aging information such as expiry date and time, enter:

chage -l userNameHere

To see account aging info for vivek, enter:  
$ chage -l vivek  
Sample outputs:

Last password change : Jan 10, 2012

Password expires : never

Password inactive : never

Account expires : never

Minimum number of days between password change : 0

Maximum number of days between password change : 99999

Number of days of warning before password expires : 7

# chage username  
OR  
# chage -I -1 -m 0 -M 99999 -E -1 username

# Linux: Force Users To Change Their Passwords Upon First Login

Q. How can I force my Linux users to change their passwords upon the first login under CentOS / Debian Linux?  
  
A. You can use any one of the following command to change user passwords upon the first login:  
[a] **usermod command** - Modify various user account properties including user password expiry information.  
Where,

[b] **chage command** - Change user password expiry information

## Task: Use chage command to force users to chage their password upon first login

Use the following syntax:  
chage -d 0 {user-name}  
# chage -d 0 tom

* **-d 0** : Set the number of days since January 1st, 1970 when the password was last changed. The date may also be expressed in the format YYYY-MM-DD. By setting it to zero, you are going to force user to change password upon first login.

## #1: top - Process Activity Command

The top program provides a dynamic real-time view of a running system i.e. actual process activity. By default, it displays the most CPU-intensive tasks running on the server and updates the list every five seconds.

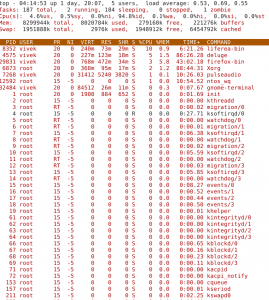
[](http://www.cyberciti.biz/tips/top-linux-monitoring-tools.html/top-output)

Fig.01: Linux top command

### Commonly Used Hot Keys

The top command provides several useful hot keys:

|  |  |
| --- | --- |
| **Hot Key** | **Usage** |
| t | Displays summary information off and on. |
| m | Displays memory information off and on. |
| A | Sorts the display by top consumers of various system resources. Useful for quick identification of performance-hungry tasks on a system. |
| f | Enters an interactive configuration screen for top. Helpful for setting up top for a specific task. |
| o | Enables you to interactively select the ordering within top. |
| r | Issues renice command. |
| k | Issues kill command. |
| z | Turn on or off color/mono |

**=> Related:** [How do I Find Out Linux CPU Utilization?](http://www.cyberciti.biz/tips/how-do-i-find-out-linux-cpu-utilization.html)

## #2: vmstat - System Activity, Hardware and System Information

The command vmstat reports information about processes, memory, paging, block IO, traps, and cpu activity.  
# vmstat 3  
Sample Outputs:

procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu------

r b swpd free buff cache si so bi bo in cs us sy id wa st

0 0 0 2540988 522188 5130400 0 0 2 32 4 2 4 1 96 0 0

1 0 0 2540988 522188 5130400 0 0 0 720 1199 665 1 0 99 0 0

0 0 0 2540956 522188 5130400 0 0 0 0 1151 1569 4 1 95 0 0

0 0 0 2540956 522188 5130500 0 0 0 6 1117 439 1 0 99 0 0

0 0 0 2540940 522188 5130512 0 0 0 536 1189 932 1 0 98 0 0

0 0 0 2538444 522188 5130588 0 0 0 0 1187 1417 4 1 96 0 0

0 0 0 2490060 522188 5130640 0 0 0 18 1253 1123 5 1 94 0 0

### Display Memory Utilization Slabinfo

# vmstat -m

### Get Information About Active / Inactive Memory Pages

# vmstat -a  
**=> Related:** [How do I find out Linux Resource utilization to detect system bottlenecks?](http://www.cyberciti.biz/tips/linux-resource-utilization-to-detect-system-bottlenecks.html)

## #3: w - Find Out Who Is Logged on And What They Are Doing

w command displays information about the users currently on the machine, and their processes.  
# w username  
# w vivek  
Sample Outputs:

17:58:47 up 5 days, 20:28, 2 users, load average: 0.36, 0.26, 0.24

USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT

root pts/0 10.1.3.145 14:55 5.00s 0.04s 0.02s vim /etc/resolv.conf

root pts/1 10.1.3.145 17:43 0.00s 0.03s 0.00s w

### #4: uptime - Tell How Long The System Has Been Running

The uptime command can be used to see how long the server has been running. The current time, how long the system has been running, how many users are currently logged on, and the system load averages for the past 1, 5, and 15 minutes.  
# uptime  
Output:

18:02:41 up 41 days, 23:42, 1 user, load average: 0.00, 0.00, 0.00

1 can be considered as optimal load value. The load can change from system to system. For a single CPU system 1 - 3 and SMP systems 6-10 load value might be acceptable.

## #5: ps - Displays The Processes

ps command will report a snapshot of the current processes. To select all processes use the -A or -e option:  
# ps -A  
Sample Outputs:

PID TTY TIME CMD

1 ? 00:00:02 init

2 ? 00:00:02 migration/0

3 ? 00:00:01 ksoftirqd/0

4 ? 00:00:00 watchdog/0

5 ? 00:00:00 migration/1

6 ? 00:00:15 ksoftirqd/1

4881 ? 00:53:28 java

4885 tty1 00:00:00 mingetty

4886 tty2 00:00:00 mingetty

4887 tty3 00:00:00 mingetty

4888 tty4 00:00:00 mingetty

4891 tty5 00:00:00 mingetty

4892 tty6 00:00:00 mingetty

4893 ttyS1 00:00:00 agetty

12853 ? 00:00:00 cifsoplockd

12854 ? 00:00:00 cifsdnotifyd

14231 ? 00:10:34 lighttpd

14232 ? 00:00:00 php-cgi

54981 pts/0 00:00:00 vim

55465 ? 00:00:00 php-cgi

55546 ? 00:00:00 bind9-snmp-stat

55704 pts/1 00:00:00 ps

ps is just like top but provides more information.

### Show Long Format Output

# ps -Al  
To turn on extra full mode (it will show command line arguments passed to process):  
# ps -AlF

### To See Threads ( LWP and NLWP)

# ps -AlFH

### To See Threads After Processes

# ps -AlLm

### Print All Process On The Server

# ps ax  
# ps axu

### Print A Process Tree

# ps -ejH  
# ps axjf  
# pstree

### Print Security Information

# ps -eo euser,ruser,suser,fuser,f,comm,label  
# ps axZ  
# ps -eM

### See Every Process Running As User Vivek

# ps -U vivek -u vivek u

### Set Output In a User-Defined Format

# ps -eo pid,tid,class,rtprio,ni,pri,psr,pcpu,stat,wchan:14,comm  
# ps axo stat,euid,ruid,tty,tpgid,sess,pgrp,ppid,pid,pcpu,comm  
# ps -eopid,tt,user,fname,tmout,f,wchan

### Display Only The Process IDs of Lighttpd

# ps -C lighttpd -o pid=  
OR  
# pgrep lighttpd  
OR  
# pgrep -u vivek php-cgi

### Display The Name of PID 55977

# ps -p 55977 -o comm=

### Find Out The Top 10 Memory Consuming Process

# ps -auxf | sort -nr -k 4 | head -10

### Find Out top 10 CPU Consuming Process

# ps -auxf | sort -nr -k 3 | head -10

## #6: free - Memory Usage

The command free displays the total amount of free and used physical and swap memory in the system, as well as the buffers used by the kernel.  
# free   
Sample Output:

total used free shared buffers cached

Mem: 12302896 9739664 2563232 0 523124 5154740

-/+ buffers/cache: 4061800 8241096

Swap: 1052248 0 1052248

**=> Related:** :

1. [Linux Find Out Virtual Memory PAGESIZE](http://www.cyberciti.biz/faq/linux-check-the-size-of-pagesize/)
2. [Linux Limit CPU Usage Per Process](http://www.cyberciti.biz/faq/cpu-usage-limiter-for-linux/)
3. [How much RAM does my Ubuntu / Fedora Linux desktop PC have?](http://www.cyberciti.biz/tips/how-much-ram-does-my-linux-system.html)

## #7: iostat - Average CPU Load, Disk Activity

The command iostat report Central Processing Unit (CPU) statistics and input/output statistics for devices, partitions and network filesystems (NFS).  
# iostat   
Sample Outputs:

Linux 2.6.18-128.1.14.el5 (www03.nixcraft.in) 06/26/2009

avg-cpu: %user %nice %system %iowait %steal %idle

3.50 0.09 0.51 0.03 0.00 95.86

Device: tps Blk\_read/s Blk\_wrtn/s Blk\_read Blk\_wrtn

sda 22.04 31.88 512.03 16193351 260102868

sda1 0.00 0.00 0.00 2166 180

sda2 22.04 31.87 512.03 16189010 260102688

sda3 0.00 0.00 0.00 1615 0

**=> Related:** : [Linux Track NFS Directory / Disk I/O Stats](http://www.cyberciti.biz/faq/howto-linux-track-nfs-client-disk-metrics/)

## #8: sar - Collect and Report System Activity

The sar command is used to collect, report, and save system activity information. To see network counter, enter:  
# sar -n DEV | more  
To display the network counters from the 24th:  
# sar -n DEV -f /var/log/sa/sa24 | more  
You can also display real time usage using sar:  
# sar 4 5  
Sample Outputs:

Linux 2.6.18-128.1.14.el5 (www03.nixcraft.in) 06/26/2009

06:45:12 PM CPU %user %nice %system %iowait %steal %idle

06:45:16 PM all 2.00 0.00 0.22 0.00 0.00 97.78

06:45:20 PM all 2.07 0.00 0.38 0.03 0.00 97.52

06:45:24 PM all 0.94 0.00 0.28 0.00 0.00 98.78

06:45:28 PM all 1.56 0.00 0.22 0.00 0.00 98.22

06:45:32 PM all 3.53 0.00 0.25 0.03 0.00 96.19

Average: all 2.02 0.00 0.27 0.01 0.00 97.70

**=> Related:** : [How to collect Linux system utilization data into a file](http://www.cyberciti.biz/tips/howto-write-system-utilization-data-to-file.html)

## #9: mpstat - Multiprocessor Usage

The mpstat command displays activities for each available processor, processor 0 being the first one. mpstat -P ALL to display average CPU utilization per processor:  
# mpstat -P ALL  
Sample Output:

Linux 2.6.18-128.1.14.el5 (www03.nixcraft.in) 06/26/2009

06:48:11 PM CPU %user %nice %sys %iowait %irq %soft %steal %idle intr/s

06:48:11 PM all 3.50 0.09 0.34 0.03 0.01 0.17 0.00 95.86 1218.04

06:48:11 PM 0 3.44 0.08 0.31 0.02 0.00 0.12 0.00 96.04 1000.31

06:48:11 PM 1 3.10 0.08 0.32 0.09 0.02 0.11 0.00 96.28 34.93

06:48:11 PM 2 4.16 0.11 0.36 0.02 0.00 0.11 0.00 95.25 0.00

06:48:11 PM 3 3.77 0.11 0.38 0.03 0.01 0.24 0.00 95.46 44.80

06:48:11 PM 4 2.96 0.07 0.29 0.04 0.02 0.10 0.00 96.52 25.91

06:48:11 PM 5 3.26 0.08 0.28 0.03 0.01 0.10 0.00 96.23 14.98

06:48:11 PM 6 4.00 0.10 0.34 0.01 0.00 0.13 0.00 95.42 3.75

06:48:11 PM 7 3.30 0.11 0.39 0.03 0.01 0.46 0.00 95.69 76.89

**=> Related:** : [Linux display each multiple SMP CPU processors utilization individually](http://www.cyberciti.biz/faq/linux-mpstat-command-report-processors-related-statistics/).

## #10: pmap - Process Memory Usage

The command pmap report memory map of a process. Use this command to find out causes of memory bottlenecks.  
# pmap -d PID  
To display process memory information for pid # 47394, enter:  
# pmap -d 47394  
Sample Outputs:

47394: /usr/bin/php-cgi

Address Kbytes Mode Offset Device Mapping

0000000000400000 2584 r-x-- 0000000000000000 008:00002 php-cgi

0000000000886000 140 rw--- 0000000000286000 008:00002 php-cgi

00000000008a9000 52 rw--- 00000000008a9000 000:00000 [ anon ]

0000000000aa8000 76 rw--- 00000000002a8000 008:00002 php-cgi

000000000f678000 1980 rw--- 000000000f678000 000:00000 [ anon ]

000000314a600000 112 r-x-- 0000000000000000 008:00002 ld-2.5.so

000000314a81b000 4 r---- 000000000001b000 008:00002 ld-2.5.so

000000314a81c000 4 rw--- 000000000001c000 008:00002 ld-2.5.so

000000314aa00000 1328 r-x-- 0000000000000000 008:00002 libc-2.5.so

000000314ab4c000 2048 ----- 000000000014c000 008:00002 libc-2.5.so

.....

......

..

00002af8d48fd000 4 rw--- 0000000000006000 008:00002 xsl.so

00002af8d490c000 40 r-x-- 0000000000000000 008:00002 libnss\_files-2.5.so

00002af8d4916000 2044 ----- 000000000000a000 008:00002 libnss\_files-2.5.so

00002af8d4b15000 4 r---- 0000000000009000 008:00002 libnss\_files-2.5.so

00002af8d4b16000 4 rw--- 000000000000a000 008:00002 libnss\_files-2.5.so

00002af8d4b17000 768000 rw-s- 0000000000000000 000:00009 zero (deleted)

00007fffc95fe000 84 rw--- 00007ffffffea000 000:00000 [ stack ]

ffffffffff600000 8192 ----- 0000000000000000 000:00000 [ anon ]

mapped: 933712K writeable/private: 4304K shared: 768000K

The last line is very important:

* **mapped: 933712K** total amount of memory mapped to files
* **writeable/private: 4304K** the amount of private address space
* **shared: 768000K** the amount of address space this process is sharing with others

**=> Related:** : [Linux find the memory used by a program / process using pmap command](http://www.cyberciti.biz/tips/howto-find-memory-used-by-program.html)

## #11 and #12: netstat and ss - Network Statistics

The command netstat displays network connections, routing tables, interface statistics, masquerade connections, and multicast memberships. ss command is used to dump socket statistics. It allows showing information similar to netstat. See the following resources about ss and netstat commands:

* [ss: Display Linux TCP / UDP Network and Socket Information](http://www.cyberciti.biz/tips/linux-investigate-sockets-network-connections.html)
* [Get Detailed Information About Particular IP address Connections Using netstat Command](http://www.cyberciti.biz/tips/netstat-command-tutorial-examples.html)

## #13: iptraf - Real-time Network Statistics

The iptraf command is interactive colorful IP LAN monitor. It is an ncurses-based IP LAN monitor that generates various network statistics including TCP info, UDP counts, ICMP and OSPF information, Ethernet load info, node stats, IP checksum errors, and others. It can provide the following info in easy to read format:

* Network traffic statistics by TCP connection
* IP traffic statistics by network interface
* Network traffic statistics by protocol
* Network traffic statistics by TCP/UDP port and by packet size
* Network traffic statistics by Layer2 address

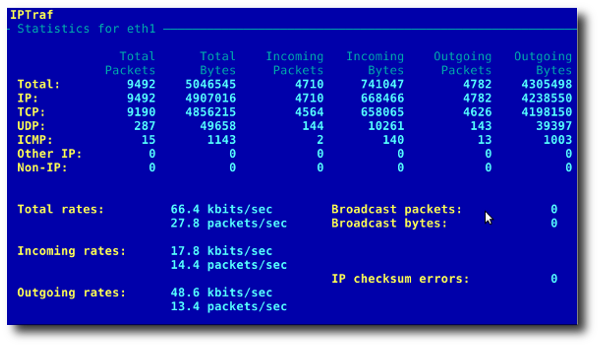
[](http://www.cyberciti.biz/tips/top-linux-monitoring-tools.html/iptraf3)

Fig.02: General interface statistics: IP traffic statistics by network interface

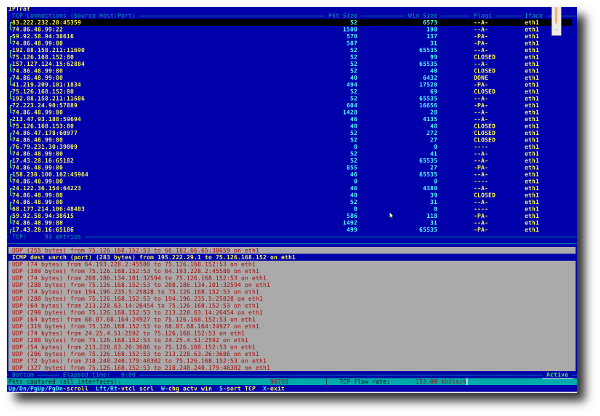
[](http://www.cyberciti.biz/tips/top-linux-monitoring-tools.html/iptraf2)

Fig.03 Network traffic statistics by TCP connection

## #14: tcpdump - Detailed Network Traffic Analysis

The tcpdump is simple command that dump traffic on a network. However, you need good understanding of TCP/IP protocol to utilize this tool. For.e.g to display traffic info about DNS, enter:  
# tcpdump -i eth1 'udp port 53'  
To display all IPv4 HTTP packets to and from port 80, i.e. print only packets that contain data, not, for example, SYN and FIN packets and ACK-only packets, enter:  
# tcpdump 'tcp port 80 and (((ip[2:2] - ((ip[0]&0xf)<<2)) - ((tcp[12]&0xf0)>>2)) != 0)'  
To display all FTP session to 202.54.1.5, enter:  
# tcpdump -i eth1 'dst 202.54.1.5 and (port 21 or 20'  
To display all HTTP session to 192.168.1.5:  
# tcpdump -ni eth0 'dst 192.168.1.5 and tcp and port http'  
Use [wireshark to view detailed](http://www.cyberciti.biz/faq/linux-unix-bsd-apache-tcpdump-http-packets-sniffing/) information about files, enter:  
# tcpdump -n -i eth1 -s 0 -w output.txt src or dst port 80

## #15: strace - System Calls

Trace system calls and signals. This is useful for debugging webserver and other server problems. See how to use to [trace the process and](http://www.cyberciti.biz/tips/linux-strace-command-examples.html) see What it is doing.

## #16: /Proc file system - Various Kernel Statistics

/proc file system provides detailed information about various hardware devices and other Linux kernel information. See [Linux kernel /proc](http://www.cyberciti.biz/files/linux-kernel/Documentation/filesystems/proc.txt) documentations for further details. Common /proc examples:  
# cat /proc/cpuinfo  
# cat /proc/meminfo  
# cat /proc/zoneinfo  
# cat /proc/mounts

## 17#: Nagios - Server And Network Monitoring

[Nagios](http://www.nagios.org/) is a popular open source computer system and network monitoring application software. You can easily monitor all your hosts, network equipment and services. It can send alert when things go wrong and again when they get better. [FAN is](http://fannagioscd.sourceforge.net/drupal/) "Fully Automated Nagios". FAN goals are to provide a Nagios installation including most tools provided by the Nagios Community. FAN provides a CDRom image in the standard ISO format, making it easy to easilly install a Nagios server. Added to this, a wide bunch of tools are including to the distribution, in order to improve the user experience around Nagios.

## 18#: Cacti - Web-based Monitoring Tool

Cacti is a complete network graphing solution designed to harness the power of RRDTool's data storage and graphing functionality. Cacti provides a fast poller, advanced graph templating, multiple data acquisition methods, and user management features out of the box. All of this is wrapped in an intuitive, easy to use interface that makes sense for LAN-sized installations up to complex networks with hundreds of devices. It can provide data about network, CPU, memory, logged in users, Apache, DNS servers and much more. See how [to install and configure Cacti network graphing](http://www.cyberciti.biz/faq/fedora-rhel-install-cacti-monitoring-rrd-software/) tool under CentOS / RHEL.

# 2. Bonding Driver Options

Options for the bonding driver are supplied as parameters to the bonding module at load time. They may be given as command line arguments to the insmod or modprobe command, but are usually specified in either the /etc/modules.conf or /etc/modprobe.conf configuration file, or in a distro-specific configuration file (some of which are detailed in the next section).

The available bonding driver parameters are listed below. If a parameter is not specified the default value is used. When initially configuring a bond, it is recommended "tail -f /var/log/messages" be run in a separate window to watch for bonding driver error messages.

It is critical that either the miimon or arp\_interval and arp\_ip\_target parameters be specified, otherwise serious network degradation will occur during link failures. Very few devices do not support at least miimon, so there is really no reason not to use it.

Options with textual values will accept either the text name or, for backwards compatibility, the option value. E.g., "mode=802.3ad" and "mode=4" set the same mode.

The parameters are as follows:

arp\_interval

Specifies the ARP link monitoring frequency in milliseconds.

If ARP monitoring is used in an etherchannel compatible mode

(modes 0 and 2), the switch should be configured in a mode

that evenly distributes packets across all links. If the

switch is configured to distribute the packets in an XOR

fashion, all replies from the ARP targets will be received on

the same link which could cause the other team members to

fail. ARP monitoring should not be used in conjunction with

miimon. A value of 0 disables ARP monitoring. The default

value is 0.

arp\_ip\_target

Specifies the IP addresses to use as ARP monitoring peers when

arp\_interval is > 0. These are the targets of the ARP request

sent to determine the health of the link to the targets.

Specify these values in ddd.ddd.ddd.ddd format. Multiple IP

addresses must be separated by a comma. At least one IP

address must be given for ARP monitoring to function. The

maximum number of targets that can be specified is 16. The

default value is no IP addresses.

downdelay

Specifies the time, in milliseconds, to wait before disabling

a slave after a link failure has been detected. This option

is only valid for the miimon link monitor. The downdelay

value should be a multiple of the miimon value; if not, it

will be rounded down to the nearest multiple. The default

value is 0.

lacp\_rate

Option specifying the rate in which we'll ask our link partner

to transmit LACPDU packets in 802.3ad mode. Possible values

are:

slow or 0

Request partner to transmit LACPDUs every 30 seconds

fast or 1

Request partner to transmit LACPDUs every 1 second

The default is slow.

max\_bonds

Specifies the number of bonding devices to create for this

instance of the bonding driver. E.g., if max\_bonds is 3, and

the bonding driver is not already loaded, then bond0, bond1

and bond2 will be created. The default value is 1.

miimon

Specifies the MII link monitoring frequency in milliseconds.

This determines how often the link state of each slave is

inspected for link failures. A value of zero disables MII

link monitoring. A value of 100 is a good starting point.

The use\_carrier option, below, affects how the link state is

determined. See the High Availability section for additional

information. The default value is 0.

mode

Specifies one of the bonding policies. The default is

balance-rr (round robin). Possible values are:

balance-rr or 0

Round-robin policy: Transmit packets in sequential

order from the first available slave through the

last. This mode provides load balancing and fault

tolerance.

active-backup or 1

Active-backup policy: Only one slave in the bond is

active. A different slave becomes active if, and only

if, the active slave fails. The bond's MAC address is

externally visible on only one port (network adapter)

to avoid confusing the switch.

In bonding version 2.6.2 or later, when a failover

occurs in active-backup mode, bonding will issue one

or more gratuitous ARPs on the newly active slave.

One gratutious ARP is issued for the bonding master

interface and each VLAN interfaces configured above

it, provided that the interface has at least one IP

address configured. Gratuitous ARPs issued for VLAN

interfaces are tagged with the appropriate VLAN id.

This mode provides fault tolerance. The primary

option, documented below, affects the behavior of this

mode.

balance-xor or 2

XOR policy: Transmit based on the selected transmit

hash policy. The default policy is a simple [(source

MAC address XOR'd with destination MAC address) modulo

slave count]. Alternate transmit policies may be

selected via the xmit\_hash\_policy option, described

below.

This mode provides load balancing and fault tolerance.

broadcast or 3

Broadcast policy: transmits everything on all slave

interfaces. This mode provides fault tolerance.

802.3ad or 4

IEEE 802.3ad Dynamic link aggregation. Creates

aggregation groups that share the same speed and

duplex settings. Utilizes all slaves in the active

aggregator according to the 802.3ad specification.

Slave selection for outgoing traffic is done according

to the transmit hash policy, which may be changed from

the default simple XOR policy via the xmit\_hash\_policy

option, documented below. Note that not all transmit

policies may be 802.3ad compliant, particularly in

regards to the packet mis-ordering requirements of

section 43.2.4 of the 802.3ad standard. Differing

peer implementations will have varying tolerances for

noncompliance.

Prerequisites:

1. Ethtool support in the base drivers for retrieving

the speed and duplex of each slave.

2. A switch that supports IEEE 802.3ad Dynamic link

aggregation.

Most switches will require some type of configuration

to enable 802.3ad mode.

balance-tlb or 5

Adaptive transmit load balancing: channel bonding that

does not require any special switch support. The

outgoing traffic is distributed according to the

current load (computed relative to the speed) on each

slave. Incoming traffic is received by the current

slave. If the receiving slave fails, another slave

takes over the MAC address of the failed receiving

slave.

Prerequisite:

Ethtool support in the base drivers for retrieving the

speed of each slave.

balance-alb or 6

Adaptive load balancing: includes balance-tlb plus

receive load balancing (rlb) for IPV4 traffic, and

does not require any special switch support. The

receive load balancing is achieved by ARP negotiation.

The bonding driver intercepts the ARP Replies sent by

the local system on their way out and overwrites the

source hardware address with the unique hardware

address of one of the slaves in the bond such that

different peers use different hardware addresses for

the server.

Receive traffic from connections created by the server

is also balanced. When the local system sends an ARP

Request the bonding driver copies and saves the peer's

IP information from the ARP packet. When the ARP

Reply arrives from the peer, its hardware address is

retrieved and the bonding driver initiates an ARP

reply to this peer assigning it to one of the slaves

in the bond. A problematic outcome of using ARP

negotiation for balancing is that each time that an

ARP request is broadcast it uses the hardware address

of the bond. Hence, peers learn the hardware address

of the bond and the balancing of receive traffic

collapses to the current slave. This is handled by

sending updates (ARP Replies) to all the peers with

their individually assigned hardware address such that

the traffic is redistributed. Receive traffic is also

redistributed when a new slave is added to the bond

and when an inactive slave is re-activated. The

receive load is distributed sequentially (round robin)

among the group of highest speed slaves in the bond.

When a link is reconnected or a new slave joins the

bond the receive traffic is redistributed among all

active slaves in the bond by initiating ARP Replies

with the selected mac address to each of the

clients. The updelay parameter (detailed below) must

be set to a value equal or greater than the switch's

forwarding delay so that the ARP Replies sent to the

peers will not be blocked by the switch.

Prerequisites:

1. Ethtool support in the base drivers for retrieving

the speed of each slave.

2. Base driver support for setting the hardware

address of a device while it is open. This is

required so that there will always be one slave in the

team using the bond hardware address (the

curr\_active\_slave) while having a unique hardware

address for each slave in the bond. If the

curr\_active\_slave fails its hardware address is

swapped with the new curr\_active\_slave that was

chosen.

primary

A string (eth0, eth2, etc) specifying which slave is the

primary device. The specified device will always be the

active slave while it is available. Only when the primary is

off-line will alternate devices be used. This is useful when

one slave is preferred over another, e.g., when one slave has

higher throughput than another.

The primary option is only valid for active-backup mode.

updelay

Specifies the time, in milliseconds, to wait before enabling a

slave after a link recovery has been detected. This option is

only valid for the miimon link monitor. The updelay value

should be a multiple of the miimon value; if not, it will be

rounded down to the nearest multiple. The default value is 0.

use\_carrier

Specifies whether or not miimon should use MII or ETHTOOL

ioctls vs. netif\_carrier\_ok() to determine the link

status. The MII or ETHTOOL ioctls are less efficient and

utilize a deprecated calling sequence within the kernel. The

netif\_carrier\_ok() relies on the device driver to maintain its

state with netif\_carrier\_on/off; at this writing, most, but

not all, device drivers support this facility.

Squid is most popular open-source software that brings this to us. It also has some excellent features for doing something else such as web access controlling, bandwidth controlling, restriction policies, and content caching and filtering. Actually people install SQUID to pursuit 2 goals: first reduce the bandwidth charges by content caching and second for restricting access to particular contents.

The following guide explains advantages of using Squid and will show you how to install, configure, control, and maintain the Squid Proxy Server on RHEL5 and CentOS Linux.

\*\*\* Notice : This guide or tutorial or whatever you like to call it is based on my personal exprience and I guarantee to you 100% that it's working like a charm for me. So if you install this software and for any reason you have technical difficulties just post the comment and I'll be with you to solve that. \*\*\*

Just something that you should know:   
'#' hash sign before rule line in config file will disable the rule.  
If you need to use your proxy server you need to modify the browser settings on your client computer, for example: IE > Internet Option > Lan Setting > enable proxy server checkbox > set IP Address of your Squid proxy server and port (default is 3128).

Before anything: If you are not sure Squid was installed, type the following command:

# rpm -q squid

squid-2.6.STABLE6-5.el5\_1.3     //this means you have squid installed on your box and do not need to install, so prepare your self for the configuration.

To install on RHEL5/CentOS type this command:

# yum install squid

And if you cannot use yum then try this way:

First download the latest version of squid from <http://www.squid-cache.org/> (official Squid website) and move it to /tmp:

# cd /tmp  
# rpm -ivh squid-2.6.STABLE.rpm

This will install squid configurations and binaries in their directories. After that use this command to run the program automatically when the system boots:

# chkconfig --level 35 squid on     // runlevel 3 is for running squid on text based and 5 is for running on x environments

Ok, now it's time to start the service so:

# service squid start

For the configuration you need to open the config file depending on your version of Linux, for RHEL5/CentOS do like this:

# vi /etc/squid/squid.conf

That's it, you can define most parameters in here, remember on start or restart of the service or viewing the log files you may see this error:

WARNING: Could not determine this machines public hostname. Please configure one or set 'visible\_hostname'.

It means the hostname isn't correctly defined and you need to change the visible\_hostname in the config file. It needs to change for identity of the cache server or troubleshooting or viewing the logs. So change it before anything else like this:

visible\_hostname HowtoForge

As you can see http\_port 3128, it means Squid listens for requests from HTTP clients on this port.

### Access Control Lists (ACL)

ACLs are used to restrict usage, limit web access of host(s); each ACL line defines a particular type of activity, such as an access time or source network, after that we need to link the ACL to an http\_access statement that tells Squid whether or not to deny or allow traffic that matches the ACL.

When you install Squid for the first time, you need to add some acls to allow your network to use the internet because squid by default denies web access.

The syntax of an ACL is like this:

acl aclname acltype value

aclname = rulename (it could be some desire name like mynetwork)

acltype = type of acl like : src, dst (src:source ip | dst:destination ip)

value = it could be ip address, networks, URLs ,...

This example will allow localhost to access the internet:

acl localhost src 127.0.0.1/32  
http\_access allow localhost

We are allowing the computer that matches the ip address range contained in the localhost ACL to access the internet. There are other ACLs and ACL-operators available for Squid, but this is good for practice.

So with this syntax, you can now tell squid how to work. Suppose you want to allow your 192.168.1 network range to access the internet, you can do this but first open the config file and find these lines:

http\_access allow localhost

http\_access deny all

Replace them with:

acl mynetwork src 192.168.1.0/24

http\_access allow localhost

http\_access allow mynetwork

http\_access deny all

Note: Specify the rules before the line http\_access deny all. After that change save your file and restart the squid service.

(If you use vi editor use this to save and quit > 1-press ESC key 2-type ':x' without quotation and hit enter.)

# service squid restart

Remember you may see an error after restarting the squid service for using "/24" in your config, if so don't panic you can easily change /24 to /255.255.255.0 and again restart the squid service, after restarting your entire network which uses the IP addresses 192.168.1.1 to 192.168.1.254 have access to the internet.

You may ask yourself about allowing internet to everyone except particular ip addresses, actually it's a good start and brings some fun :) . Ok, to do this open the config file and do like this:

acl bad\_employee src 192.168.1.18

http\_access deny bad\_employee

acl mynetwork src 192.168.1.0/24

http access allow mynetwork

In the above example the entire network will be allowed to use the internet except the blocked person (bad\_employee). Remember Squid interprets the rules from top to bottom, so you need to be careful.

You can create a restricting rule by times for your company and assign that to your created mynetwork acl like this:

acl mynetwork src 192.168.1.0/24

acl business\_hours time M T W H F 9:00-17:00

acl bad\_employee src 192.168.1.18

http\_access deny bad\_employee

http\_access allow mynetwork business\_hours

Day-abbrevs: S - Sunday M - Monday T – Tuesday W - Wednesday H - Thursday F - Friday A - Saturday

You can also block a particular URL like this:

acl block\_site dst www.yahoo.com

http\_access deny block\_site

www.yahoo.com will be filtered BUT mail.yahoo.com is open because we block yahoo.com, so if you want to block a single url and its subdomains we do it like this:

acl block\_domain dstdomain .yahoo.com

http\_access deny block\_domain

And you can do more than blocking one URL, if you want to block more than a single domain we need to create a file to hold the particular URLs and give this file read permissions like this:

# touch /etc/squid/block\_list.txt  
# chmod 444 /etc/squid/block\_list.txt   
# vi /etc/squid/block\_list.txt

Enter some URLs to block like this:

www.sxx.com

www.yahoo.com

www.hotmail.com

And then save and quit, it's time to create rules. Open the config file and put these parameters in it:

acl block\_list url\_regex "/etc/squid/block\_list.txt"

http\_access deny block\_list

You can block the URLs that contain unexpected words like this:

acl blockword url\_regex sxx

http\_access deny blockword

(You can block case insensitive words like this : -i sxx)

You can block downloads of .exe files like this:

acl block\_exe url\_regex .\*\.exe$

http\_access deny block\_exe

If you want block more extensions to download you can specify all in a file as described before (exact like some URL to block section).

You can block TLDs (.br .eu) like this:

acl block\_tld dstdom\_regex \.br$

http\_access deny block\_tld

You can configure squid to prompt for username and password from users with ncsa\_auth that reads an NCSA-compliant encrypted password file, so:

# htpasswd -c /etc/squid/squid\_passwd your\_username

enter pass : your\_password

# chmod o+r /etc/squid/squid\_passwd

Open the config file and put these lines in it and change to your own configuration:

auth\_param basic program /usr/lib/squid/ncsa\_auth /etc/squid/squid\_passwd

acl ncsa\_user proxy\_auth REQUIRED

http\_access allow ncsa\_user

If you don't want to modify the browser for using a proxy there is a method that is called "Transparent Proxy"; to use this you need to do like this:

Prior to Squid Version 2.6:

httpd\_accel\_host virtual

httpd\_accel\_port 80

httpd\_accel\_with\_proxy on

httpd\_accel\_uses\_host\_header on

Squid Version 2.6 to 3.0:

http\_port 3128 transparent

Squid Version 3.1+ :

http\_port 3128 intercept

# How To Set Up an NFS Mount on CentOS 6

Tagged In: [Linux Basics](https://www.digitalocean.com/community/tags/linux-basics), [CentOS](https://www.digitalocean.com/community/tags/centos)

### About NFS (Network File System) Mounts

NFS mounts work to share a directory between several servers. This has the advantage of saving disk space, as the home directory is only kept on one server, and others can connect to it over the network. When setting up mounts, NFS is most effective for permanent fixtures that should always be accessible.

## Setup

An NFS mount is set up between at least two servers. The machine hosting the shared network is called the server, while the ones that connect to it are called ‘clients’.

This tutorial requires 2 servers: one acting as the server and one as the client. We will set up the server machine first, followed by the client. The following IP addresses will refer to each one:

**Master: 12.34.56.789**

**Client: 12.33.44.555**

The system should be set up as root. You can access the root user by typing

sudo su

## Setting Up the NFS Server

### Step One—Download the Required Software

Start off by using apt-get to install the nfs programs.

yum install nfs-utils nfs-utils-lib

Subsequently, run several startup scripts for the NFS server:

chkconfig nfs on

service rpcbind start

service nfs start

### Step Two—Export the Shared Directory

The next step is to decide which directory we want to share with the client server. The chosen directory should then be added to the /etc/exports file, which specifies both the directory to be shared and the details of how it is shared.

Suppose we wanted to share the directory, /home.

We need to export the directory:

vi /etc/exports

Add the following lines to the bottom of the file, sharing the directory with the client:

/home 12.33.44.555(rw,sync,no\_root\_squash,no\_subtree\_check)

These settings accomplish several tasks:

* **rw:** This option allows the client server to both read and write within the shared directory
* **sync:** Sync confirms requests to the shared directory only once the changes have been committed.
* **no\_subtree\_check:** This option prevents the subtree checking. When a shared directory is the subdirectory of a larger filesystem, nfs performs scans of every directory above it, in order to verify its permissions and details. Disabling the subtree check may increase the reliability of NFS, but reduce security.
* **no\_root\_squash:** This phrase allows root to connect to the designated directory

Once you have entered in the settings for each directory, run the following command to export them:

exportfs -a

## Setting Up the NFS Client

### Step One—Download the Required Software

Start off by using apt-get to install the nfs programs.

yum install nfs-utils nfs-utils-lib

### Step Two—Mount the Directories

Once the programs have been downloaded to the the client server, create the directory that will contain the NFS shared files

mkdir -p /mnt/nfs/home

Then go ahead and mount it

mount 12.34.56.789:/home /mnt/nfs/home

You can use the df -h command to check that the directory has been mounted. You will see it last on the list.

df -h

Filesystem Size Used Avail Use% Mounted on

/dev/sda 20G 783M 18G 5% /

12.34.56.789:/home 20G 785M 18G 5% /mnt/nfs/home

Additionally, use the mount command to see the entire list of mounted file systems.

mount

Your list should look something like this:

/dev/sda on / type ext4 (rw,errors=remount-ro)

none on /proc/sys/fs/binfmt\_misc type binfmt\_misc (rw)

sunrpc on /var/lib/nfs/rpc\_pipefs type rpc\_pipefs (rw)

nfsd on /proc/fs/nfsd type nfsd (rw)

12.34.56.789:/home on /mnt/nfs/home type nfs (rw,noatime,nolock,bg,nfsvers=2,intr,tcp,actimeo=1800,addr=12.34.56.789)

## Testing the NFS Mount

Once you have successfully mounted your NFS directory, you can test that it works by creating a file on the Client and checking its availability on the Server.

Create a file in the directory to try it out:

touch /mnt/nfs/home/example

You should then be able to find the files on the Server in the /home.

ls /home

You can ensure that the mount is always active by adding the directory to the fstab file on the client. This will ensure that the mount starts up after the server reboots.

vi /etc/fstab

12.34.56.789:/home /mnt/nfs/home nfs auto,noatime,nolock,bg,nfsvers=3,intr,tcp,actimeo=1800 0 0

You can learn more about the fstab options by typing in:

man nfs

After any subsequent server reboots, you can use a single command to mount directories specified in the fstab file:

mount -a

You can check the mounted directories with the two earlier commands:

df -h

mount

## Removing the NFS Mount

Should you decide to remove a directory, you can unmount it using the umount command:

cd

sudo umount /*directory name*

You can see that the mounts were removed by then looking at the filesystem again.

df -h

You should find your selected mounted directory gone.

# NFS Stale File Handle error and solution

by [NIX Craft](http://www.cyberciti.biz/tips/about-us) on October 9, 2006 · [36 comments](http://www.cyberciti.biz/tips/nfs-stale-file-handle-error-and-solution.html#comments)· LAST UPDATED May 14, 2008

in [Linux](http://www.cyberciti.biz/tips/category/linux), [Tips](http://www.cyberciti.biz/tips/category/tips), [Troubleshooting](http://www.cyberciti.biz/tips/category/troubleshooting)

Sometime NFS can result in to weird problems. For example NFS mounted directories sometimes contain stale file handles. If you run command such as ls or vi you will see an error:  
$ ls  
.: Stale File Handle

First let us try to understand the concept of Stale File Handle. [Managing NFS and NIS](http://www.amazon.com/gp/redirect.html?ie=UTF8&location=http%3A%2F%2Fwww.amazon.com%2FManaging-NIS-2nd-Hal-Stern%2Fdp%2F1565925106%2Fsr%3D1-1%2Fqid%3D1160421456%3Fie%3DUTF8%26s%3Dbooks&tag=cyberciti-20&linkCode=ur2&camp=1789&creative=9325), 2nd Edition book defines filehandles as follows (a good book if you would like to master NFS and NIS):  
A filehandle becomes stale whenever the file or directory referenced by the handle is removed by another host, while your client still holds an active reference to the object. A typical example occurs when the current directory of a process, running on your client, is removed on the server (either by a process running on the server or on another client).  
So this can occur if the directory is modified on the NFS server, but the directories modification time is not updated.

### How do I fix this problem?

a) The best solution is to remount directory from the NFS client using [mount command](http://www.cyberciti.biz/tips/ubuntu-linux-nfs-client-configuration-to-mount-nfs-share.html):  
# [umount -f /mnt/local](http://www.cyberciti.biz/tips/how-do-i-forcefully-unmount-a-disk-partition.html)  
# mount -t nfs nfsserver:/path/to/share /mnt/local

First [command](http://www.cyberciti.biz/tips/how-do-i-forcefully-unmount-a-disk-partition.html) (umount) forcefully unmount a disk partition /mnt/local (NFS).

(b) Or try to mount NFS directory with the noac option. However I don't recommend using noac option because of performance issue and Checking files on NFS filesystem referenced by file descriptors (i.e. the fcntl and ioctl families of functions) may lead to inconsistent result due to the lack of consistency check in kernel even if noac is used.

# What are the different RAID levels for Linux / UNIX and Windows Server?

by [Nix Craft](http://www.cyberciti.biz/tips/about-us) on July 5, 2007 · [5 comments](http://www.cyberciti.biz/faq/different-raid-levels/#comments)· LAST UPDATED October 22, 2008

in [File system](http://www.cyberciti.biz/faq/category/file-system/), [FreeBSD](http://www.cyberciti.biz/faq/category/freebsd/), [Hardware](http://www.cyberciti.biz/faq/category/hardware/)

[](http://www.cyberciti.biz/faq/category/file-system/)

**Q**. What are the different RAID levels? Which one is recommended for file server and database server?  
  
**A**. A [Redundant Array of Independent Drives](http://en.wikipedia.org/wiki/RAID) (or Disks), also known as Redundant Array of Inexpensive Drives (or Disks) (RAID) is an term for data storage schemes that divide and/or replicate data among multiple hard drives. RAID can be designed to provide increased data reliability or increased I/O performance, though one goal may compromise the other.

There are total 10 types of RAID levels:

* RAID level 0
* RAID level RAID level 1
* RAID level 2
* RAID level 3
* RAID level 4
* RAID level 5
* RAID level 6
* RAID level 10
* RAID level 50
* RAID level 0+1

### Commonly used RAID levels for UNIX / Linux and Windows server

Following are commonly used RAID levels :

|  |  |  |  |
| --- | --- | --- | --- |
| **RAID level** | **Minimum hard disks** | **Suggested application** | **Notes** |
| RAID 0 - Striped Set without parity | 2 Hard disks | 1. Video Production and Editing 2. Image Editing 3. Any application requiring high bandwidth | Provides improved performance and additional storage but no fault tolerance from disk errors or disk failure. Any disk failure destroys the array, which becomes more likely with more disks in the array. |
| RAID 1 - **Mirrored** Set (2 disks minimum) without parity. | 2 Hard disks | 1. Office application 2. Financial application 3. Payroll application etc | Provides fault tolerance from disk errors and single disk failure. Increased read performance occurs when using a multi-threaded operating system that supports split seeks, very small performance reduction when writing. Array continues to operate so long as at least one drive is functioning |
| RAID 5 | 3 Hard disks | 1. File and Application servers 2. Internet Web, E-mail servers 3. Intranet servers | Highest Read data transaction rate, Medium Write data transaction rate, Overall good (aggregate) transfer rate. drive failure requires replacement, but the array is not destroyed by a single drive failure. Upon drive failure, any subsequent reads can be calculated from the distributed parity such that the drive failure is masked from the end user. The array will have data loss in the event of a second drive failure and is vulnerable until the data that was on the failed drive is rebuilt onto a replacement drive |
| RAID 10 (nested RAID 1+0) | 4 Hard disks | 1. Database server (such as Oracle / MySQL / MS-SQL) which requiring high performance and fault tolerance | Provides fault tolerance and improved performance but increases complexity. |

Mike Chirico (mchirico@users.sourceforge.net) or (mchirico@comcast.net)  
Copyright (c) 2005 (GNU Free Documentation License)  
Last Updated: Sun Jan 15 08:23:29 EST 2006   
[<http://souptonuts.sourceforge.net/quota_tutorial.html>]

**Implementing Disk Quotas on Linux**

This tutorial walks you through implementing disk quotas for both users and groups on Linux, using a virtual filesystem, which is a filesystem created from a disk file. Since quotas work on a per-filesystem basis, this is a way to implement quotas on a sub-section, or even multiple subsections of your drive, without reformatting. This tutorial also covers quotactl, or quota's C interface, by way of an example program that can store disk usage in a SQLite database for monitoring data usage over time.

This tutorial was tested on Fedora Core 2, 3, and 4. I'm assuming you have the quota tools installed. If you're not sure, try the following test, which will return 3.12-6 or 3.12-5 depending on which version of Fedora Core you are using.

$ rpm -q quota

quota-3.12-6

**Creating a Virtual Filesystem**

The following steps walk you through creating a ext3 virtual filesystem mounted on "/quota" with a size of 20 MB. Again, since quotas are installed on a filesystem, we're going to create an isolated filesystem.

The mount point for this filesystem will be "quota". As root, first create the mount point quota, which at this point is just a directory.

# mkdir -p /quota

Next, create a 20M file (disk image) in a suitable location. What I did below is create the file disk-quota.ext3 in the directory /usr/disk-img.

# mkdir -p /usr/disk-img

# dd if=/dev/zero of=/usr/disk-img/disk-quota.ext3 count=40960

The dd command above created a 20MB file because, by default, dd uses a block size of 512 bytes. That makes this size: 40960\*512=20971520. For kicks, we'll confirm this size.

# ls -lh /usr/disk-img/disk-quota.ext3

-rw-r--r-- 1 root root 20M Jul 19 14:34 /usr/disk-img/disk-quota.ext3

Next, format this as an ext3 filesystem.

# /sbin/mkfs -t ext3 -q /usr/disk-img/disk-quota.ext3 -F

The "-t" gives it the type. You're not limited to ext3. In fact, you could use ext2 or other filesystems installed on your system. The "-q" is for the device, and "-F" is to force the creation without warning us that this is a file and not a block device.

Add the following line to "/etc/fstab". This will make the filesystem always available on reboot, plus it's easier to mount and unmout when testing.

/usr/disk-img/disk-quota.ext3 /quota ext3 rw,loop,usrquota,grpquota 0 0

Now, mount this filesystem.

# mount /quota

**Or if you didn't edit /etc/fstab above**

# mount -o loop,rw,usrquota,grpquota /usr/disk-img/disk-quota.ext3 /quota

Now take a look at /quota. You should see the "lost+found" directory. Plus, you can take a look at /proc/mounts to see that you have an "ext3" type filesystem. At this point you can create and add files if you want.

# ls

lost+found

# grep 'quota' /proc/mounts

/dev/loop0 /quota ext3 rw 0 0

**The mount command below shows us usrquota and grpquota options have been added.**

# mount | grep '/quota'

/usr/disk-img/disk-quota.ext3 on /quota type ext3 (rw,loop=/dev/loop0,usrquota,grpquota)

**Sharing a Directory amoung Several Users**

This step creates a group and implements group rights on a directory within the quota filesystem. Specifically, this step creates the group, "quotagrp" and adds the two existing users "chirico" and "sporkey" into this group. The direcory "/quota/share" is setup so that any files created in this directory by these two users will be sharable by default for members of this group. This is done by setting the setgid bit on the directory.

First create the group and add any existing users.

# groupadd quotagrp

# usermod -G quotagrp chirico

# usermod -G quotagrp sporkey

Create the directory /quota/share and set the access rights so that files created in this directory can be edited by any group members.

# mkdir -p /quota/share

# chown -R root.quotagrp /quota/share

# chmod 2775 /quota/share

Above the command "2755" sets the "setgid" bit. You can see this with the "ls" command below.

# ls -ld /quota/share

drwxrwsr-x 2 root quotagrp 1024 Jul 19 15:16 /quota/share/

**^---------- Note the s, setgid bit, from chmod 2775**

An important note here. If the users above "chirico" and "sporkey" are currently logged in when they were added to the group, they will not get access to the group. These users need to login again. Having these users run "newgrp" or even "newgrp -" (Fedora core 4) will give them access to the group; however, this will not correctly set group file permissions. To avoid trouble in a production environment have users login again or execute "su - " to corrently initilize their environment.

Execute from chirico account.

$ groups

chirico

$ su - chirico

Password:

$ groups

quotagrp chirico

**Quotas**

Run quotecheck. The first time you run this command, use the "-c" option to create the necessary database files. The following should be run as root.

# quotacheck -cug /quota

Note that two files have been created "aquota.group" and "aquota.user".

# ls -l /quota

aquota.group aquota.user lost+found share

Use "edquota" to grant the user "chirico" the desired quota.

# edquota -f /quota chirico

Executing the command above brings up a text file in your default editor. You will change entries in this file.

Disk quotas for user chirico (uid 500):

Filesystem blocks soft hard inodes soft hard

/dev/loop0 0 0 0 0 0 0

Above for user chirico there have been no blocks or inodes used on this filesystem. Note that an inode is used for each file and directory. We'll change the settings as follows:

Disk quotas for user chirico (uid 500):

Filesystem blocks soft hard inodes soft hard

/dev/loop0 0 100 200 0 10 15

Note that the numbers under "blocks" and "inodes" indicated the current blocks and inodes in use by this user. Those values should not be edited, since they are only used for reference.

**setquota - command line quota editor**

You can also use the setquota command, which has the advantage of not using an editor making it ideal for implementing in a script. For example, to set the soft block limit to 100, a hard block limit of 200, a soft inode to 10 and a hard inode to 15 as we did above, execute the following command.

# setquota -u chirico 100 200 10 15 -a /dev/loop0

Turn quotas on with the following command.

# quotaon /quota

From the root user you can "su" into the chirico account to see the changes.

# su - chirico

$ touch /quota/share/t1

$ quota

Disk quotas for user chirico (uid 500):

Filesystem blocks quota limit grace files quota limit grace

/dev/loop0 1 100 200 1 10 20

As an interesting test, still under this user, create a hard link as follows with the ln command. Then execute the quota command to see how many inodes are taken.

$ ln /quota/share/t1 /quota/share/t2

$ quota

Disk quotas for user chirico (uid 500):

Filesystem blocks quota limit grace files quota limit grace

/dev/loop0 1 100 200 1 10 20

Note that the number of files has not changed. However, if you create a symbolic link, sometimes called a soft link, with the "ln -s" command, the number will increse to 2, because an additional inode is created with a soft link.

$ ln -s /quota/share/t1 /quota/share/t3

$ quota

Disk quotas for user chirico (uid 500):

Filesystem blocks quota limit grace files quota limit grace

/dev/loop0 2 100 200 2 10 20

**Quotas for Groups**

To set quotas for the group "quotagrp", use the following command.

# edquota -g quotagrp

Disk quotas for group quotagrp (gid 619):

Filesystem blocks soft hard inodes soft hard

/dev/loop0 6 0 0 4 0 0

Now make the following changes.

Disk quotas for group quotagrp (gid 619):

Filesystem blocks soft hard inodes soft hard

/dev/loop0 6 5 100 4 6 10

Or, use the "setquota" command as follows:

# setquota -g quotagrp 5 100 6 10 -a /dev/loop0

Now run the following command under the user account that has group access, which will attempt to create 15 files.

$ for i in $(seq 15); do touch "/quota/share/file\_$i"; done

loop0: warning, group file quota exceeded.

loop0: write failed, group file limit reached.

touch: cannot touch `/quota/share/file\_7': Disk quota exceeded

**Reports**

The "repquota" command prints a summarized report. It should be run with root.

# repquota /quota

\*\*\* Report for user quotas on device /dev/loop0

Block grace time: 7days; Inode grace time: 7days

Block limits File limits

User used soft hard grace used soft hard grace

----------------------------------------------------------------------

root -- 1204 0 0 5 0 0

chirico -- 10 100 200 9 10 20

To get a report by group, use the -g option as follows.

# repquota -g /quota

\*\*\* Report for group quotas on device /dev/loop0

Block grace time: 7days; Inode grace time: 7days

Block limits File limits

Group used soft hard grace used soft hard grace

----------------------------------------------------------------------

root -- 1202 0 0 4 0 0

quotagrp ++ 12 5 100 6days 10 6 10 6days

Note the "++" above for quotagrp indicating that both the block limit and inode limits have been exceeded.

Or to get everything, run repquota with the -a option as follows.

# repquota -a

**warnquota - send mail to users over quota**

Running warnquota without any options will email users that go over the limit.

# warnquota

However in this case no mail message will be sent, because the group limit was exceeded. The file "/etc/quotagrpadmins" needs to contain a username responsible for the group. Here will put in the user "sporkey", so that the file looks as follows:

#

# This is a sample groupadmins file (/etc/quotagrpadmins)

#

# Comments begin with hash in the beginning of the line

# In this file you specify users responsible for space used by the group

users: root

mygroup: chief

quotagrp: sporkey

Now if warnquota is execute with the -g option, mail will be send to user "sporkey".

# warnquota -g /quota

Messages can be customized by editing the "/etc/warnquota.conf" file.

**quotactl - C API**

You may want to create your own quota tools, especially if you forsee a need to monitor quota data over time, across multiple computers. My database tool of choice for this is [SQLite](http://www.sqlite.org/) because it is very fast, the complete database fits into one file, there is no setup or administration needed, and databases can be combine. For detailed information on using SQLite take a look at the (["SQLite Tutorial"](http://souptonuts.sourceforge.net/readme_sqlite_tutorial.html)).

**Goal - Create quotadb**

The goal is to create the program "quotadb" that will automatically create the necessary database ( a default database quota\_database) and tables, for all filesystems and all users on the system. There will also be a "-f <database>" option to place the database file in a particular location, or just place it in the default directory as "quota\_database", if no options are specified.

**Create tables in /root/quota\_database**

# quotadb -f /root/quota\_database

Now, after the initial commad is run, quotadb can be run on a peridoic basis to update the entries in the table. Update is specified with the -u option.

**-u option updates history table**

# quotadb -f /root/quota\_database -u

After the tables are created by running "quotadb -f /root/quota\_database", quotadb is placed in a cron job and run nightly. To get into the cron editor as root type "cronjob -e".

#root cronjob. Type cronjob -e

# This updates history database

#MINUTE(0-59) HOUR(0-23) DAYOFMONTH(1-31) MONTHOFYEAR(1-12) DAYOFWEEK(0-6) Note 0=Sun

2 1 \* \* \* /usr/local/bin/quotadb -f /root/quota\_database -u

Running this once a day, over a period of time, will populate the database table history, defined below.

CREATE TABLE history (filesystem varchar(50),

name varchar(50),

uid int,

b\_curr long,

b\_slimit long,

b\_hlimit long,

i\_curr long,

i\_slimit long,

i\_hlimit long,

timeEnter DATE );

CREATE TRIGGER insert\_history\_timeEnter

AFTER INSERT ON history

BEGIN UPDATE history

SET timeEnter = DATETIME('NOW','localtime')

WHERE rowid = new.rowid;

The trigger definition above updates the timeEnter field with the current date and time. By the way this is localtime, hence the option 'localtime'. The default would give you [UTC](http://ftp1.sourceforge.net/souptonuts/README_Working_With_Time.html) time.

**Structure - if\_dqblk**

One important structure that gets passed to quotactl is **if\_dqblk**. This is defined in the file "quota.h", included in the [quota\_examples.tar.gz](http://sourceforge.net/project/showfiles.php?group_id=79320&package_id=158621). You will not find this structure in **#include <linux/quota.h>** or "/usr/include/linux/quota.h" on the Fedora distros. Instead, there is a copy of this structure in the "quota-3.12-5.src.rpm". You can get this source rpm as follows:

$ wget http://download.fedora.redhat.com/pub/fedora/linux/core/3/SRPMS/quota-3.12-5.src.rpm

$ rpm -K quota-3.12-5.src.rpm

$ rpm -ivh quota-3.12-5.src.rpm

$ su

Password:

# rpm -ivh quota-3.12-5.src.rpm

1:quota ########################################### [100%]

If you're curious, the "rpm -K" command above checks the signature on the package.

After this installation, the source can be found under "/usr/src/redhat/SOURCES".

# ls -l /usr/src/redhat/SOURCES/quota-3.12.tar.gz

If you "tar -xzf" this source, you'll see "quota.h", which contains the if\_dqblk structure shown below. Note the block hard limit "dqb\_bhardlimit", block soft limit "dqb\_bsoftlimit", current space taken "dqb\_curspace", and similar all 64 bit values for the inode variables as well.

struct if\_dqblk {

u\_int64\_t dqb\_bhardlimit;

u\_int64\_t dqb\_bsoftlimit;

u\_int64\_t dqb\_curspace;

u\_int64\_t dqb\_ihardlimit;

u\_int64\_t dqb\_isoftlimit;

u\_int64\_t dqb\_curinodes;

u\_int64\_t dqb\_btime;

u\_int64\_t dqb\_itime;

u\_int32\_t dqb\_valid;

};

By the way, the Linux kernel source contains a quota.h file as well "/lib/modules/2.6.11-1.35\_FC3smp/build/include/linux/quota.h", which has an identical structure. It cannot be used in a user-land program. Kernel values for 64 bits are defined as \_\_u65 and not u\_int64\_t; but, you can certainly see these values get passed directly to the kernel.

/\* This is from the kernel source \*/

struct if\_dqblk {

\_\_u64 dqb\_bhardlimit;

\_\_u64 dqb\_bsoftlimit;

\_\_u64 dqb\_curspace;

\_\_u64 dqb\_ihardlimit;

\_\_u64 dqb\_isoftlimit;

\_\_u64 dqb\_curinodes;

\_\_u64 dqb\_btime;

\_\_u64 dqb\_itime;

\_\_u32 dqb\_valid;

};

**Example Call to quotactl**

Below is an example call to quotactl for getting quota spaced used for uid. The variable **dq** is defined as type **struct if\_dqblk** and it's passed as the last parameter to quotactl. Note the macro **QCMD(Q\_GETQUOTA, USRQUOTA)**, passed as the first parameter. Q\_GETQUOT, USRQUOTA are defined in "quota.h". The block device, as defined by block\_device is the second parameter. And the user id value is defined in uid. The function quotactl returns 0 on success and -1 on failure.

...

struct if\_dqblk dq;

if (quotactl(QCMD(Q\_GETQUOTA, USRQUOTA), block\_device, uid, (caddr\_t) & dq)) {

perror("quotactl");

return 1;

} else {

printf("Device: %s\n", block\_device);

printf

("Current space: %7.1qu \tSoft limit: %7.1qu \tHard limit: %7.1qu \tGrace period: %qu\n",

dq.dqb\_curspace, dq.dqb\_bsoftlimit, dq.dqb\_bhardlimit,

dq.dqb\_btime);

...

**Set Quota Options**

The section above was an example of getting the quota settings. Let's move on to changing or setting the quota options for the user defined in pwd->pw\_uid. It helps to have a specific example. Below we will set a block soft limit of 78, block hard limit of 96, inode soft limit of 50, and an inode hard limit of 100. This program starts with a call to getpwname which takes the user name as a string and passes back a passwd structure. The passwd structure contains the user name, password, user id, group id, real name, home directory and shell program for that particular user.

...

const struct passwd \*pwd;

struct if\_dqblk dq;

if ((pwd = getpwnam(user)) == NULL) {

printf(stderr, "Invalid -u option \n");

exit(1);

}

dq.dqb\_bsoftlimit = 78;

dq.dqb\_bhardlimit = 96;

dq.dqb\_isoftlimit = 50;

dq.dqb\_ihardlimit = 100;

if (quotactl(QCMD(Q\_SETQUOTA, USRQUOTA), block\_device, pwd->pw\_uid, (caddr\_t) & dq)) {

perror("quotactl");

return 1;

}

Again, the full examples can be downloaded from [quota\_examples.tar.gz](http://sourceforge.net/project/showfiles.php?group_id=79320&package_id=158621)

### [Contents](https://www.digitalocean.com/community/tutorials/how-to-enable-user-and-group-quotas)

[Sign Up](https://www.digitalocean.com/community/auth/digitalocean?display=sessionless+register) [Log In](https://www.digitalocean.com/community/auth/digitalocean)

* [Tutorials](https://www.digitalocean.com/community/tutorials)
* [Questions](https://www.digitalocean.com/community/questions)
* [Projects](https://www.digitalocean.com/community/projects)
* [Tags](https://www.digitalocean.com/community/tags)

Top of Form



Bottom of Form

Published Jan 15, 2014

Updated May 30, 2014

Beginner

* [SHARE](https://www.digitalocean.com/community/tutorials/how-to-enable-user-and-group-quotas)
* [TWEET](http://twitter.com/share?text=How%20To%20Enable%20User%20and%20Group%20Quotas&url=https://www.digitalocean.com/community/tutorials/how-to-enable-user-and-group-quotas)
* [SHARE](https://www.facebook.com/sharer/sharer.php?u=https://www.digitalocean.com/community/tutorials/how-to-enable-user-and-group-quotas)
* [SHARE](https://plus.google.com/share?url=https://www.digitalocean.com/community/tutorials/how-to-enable-user-and-group-quotas)
* [SHARE](https://news.ycombinator.com/submitlink?u=https://www.digitalocean.com/community/tutorials/how-to-enable-user-and-group-quotas&t=How%20To%20Enable%20User%20and%20Group%20Quotas)

# How To Enable User and Group Quotas

Tagged In: [Linux Basics](https://www.digitalocean.com/community/tags/linux-basics), [Miscellaneous](https://www.digitalocean.com/community/tags/miscellaneous), [Getting Started](https://www.digitalocean.com/community/tags/getting-started), [Ubuntu](https://www.digitalocean.com/community/tags/ubuntu), [Debian](https://www.digitalocean.com/community/tags/debian)

## Introduction

This guide assumes that a droplet with [user accounts](https://www.digitalocean.com/community/articles/how-to-add-and-delete-users-on-ubuntu-12-04-and-centos-6) has already been created. Most of the time, user quotas are applied to [FTP](https://www.digitalocean.com/community/articles/how-to-set-up-vsftpd-on-ubuntu-12-04) or [SFTP](https://www.digitalocean.com/community/articles/how-to-use-sftp-to-securely-transfer-files-with-a-remote-server) users, but it's possible to apply this to any system user. Notice that it's not possible to use VSFTP's virtual user feature with quotas – the users should exist on the system!

Quotas are used to limit the amount of disk space a user or group can use on the VPS. There are generally two different methods of managing quotas: first, an empty file system can be created and mounted for a specific user. An advantage of this method is that [no additional package is required](http://infofreund.de/user_quotas_vsftpd_en/). Second, the quota tool can be used to dynamically manage the disk quota of a user or a group. A big advantage is the possibility to change quotas on the fly without any complicated disk resizing.

This guide mainly focuses on the creation of user specific quotas; however, all the commands are exactly the same for group quotas. Instead of a user name, a group name can be used in the described commands.

## Installing Quota

This guide starts by installing the quota program using the following command:

apt-get install quota

The mount options of the file system need to be edited before user specific quotas can be used. The mount file fstab needs to be opened for editing using the following command:

sudo nano /etc/fstab

The quotas are enabled by adding a usrquota and/or grpquota to the mounting options of the main hard disk. When using ursquota, the quotas are only enabled on specific users. The grpquota option allows for quotas on user groups.

Both options can be independently added depending on the desired result. The fstab file should be edited as follows for enabling user quotas (for group quotas add grpquota).

LABEL=DOROOT / ext4 errors=remount-ro,usrquota 0 1

Save the file and enable the new mount options by remounting the file system as follows:

mount -o remount /

The following command will create a new quotas file in the root directory of the file system. This is an index file used by the quota tool for keeping track of the user's disk size. It also contains the user limits and configured options.

quotacheck -cum /

The command consists of the following three parameters:

1. The c parameter indicates the creation of a new file, overwriting any previous files.
2. The u parameter indicates that a new user index file should be created. To also create a group index file, add the g command in the previous command.
3. The m parameter indicates that no read-only mount of the complete file system is required to generate the different index files.

Because the m parameter is used, it's possible that a small mismatch happens in the actual specific user disk size and the calculated disk size by the quota program. Make sure that no user is currently uploading files to the server when running the previous command to minimize a possible mismatch.

The following command announces to the system that disk quotas should be enabled on the desired file system.

quotaon /

A similar command can be used to turn off disk quota checking, thus disabling the quotas for the different users and groups.

quotaoff

## Configuring Quotas For Different Users

The user quotas are configured using the edquota command, followed by the desired user name or group name. The command will open the default configured text editor. In this guide, we assume that the user ftpuser should receive a quota of 10Mb. The command used is as follows:

edquota ftpuser

Which opens the quota file for editing

Disk quotas for user ftpuser (uid 1001):

Filesystem blocks soft hard inodes soft hard

/dev/disk/by-label/DOROOT 8 10000 10240 2 0 0

The text editor shows 7 different columns:

1. Indicates the name of the file system that has a quota enabled
2. Indicates the amount of blocks currently used by the user
3. Indicates the **soft block** limit for the user on the file system
4. Indicates the **hard block** limit for the user on the file system
5. Indicates the amount of inodes currently used by the user
6. Indicates the **soft inode** limit for the user on the file system
7. Indicates the **hard inode** limit for the user on the file system

The blocks refer to the amount of disk space, while the inodes refer to the number of files/folders that can be used. Most of the time the block amount will be used in the quota.

The hard block limit is the absolute maximum amount of disk space that a user or group can use. Once this limit is reached, no further disk space can be used. The soft block limit defines the maximum amount of disk space that can be used. However, unlike the hard limit, the soft limit can be exceeded for a certain amount of time. This time is known as the grace period. More information about the grace period later in the guide.

In the example above, a soft limit off 9,785Mb and hard limit of 10Mb are used. To see the quota in action an FTP/SFTP transfer can be started, where multiple files will be uploaded with a total size of 12 Mb for example (as long as its larger than the hard limit). The FTP/SFTP client will indicate a transfer error, meaning that the user will be unable to upload any files. Of course, 10Mb isn't a meaningful quota. In this guide every user will get a soft limit of 976 Mb and a hard limit of 1Gb. The configuration looks as follows:

Disk quotas for user ftpuser (uid 1001):

Filesystem blocks soft hard inodes soft hard

/dev/disk/by-label/DOROOT 8 1000000 1048576 2 0 0

For checking the quota of a specific user, the quota command can be used followed by the user or group

quota ftpuser

Which gives the following output

Disk quotas for user ftpuser (uid 1001):

Filesystem blocks soft hard inodes soft hard

/dev/disk/by-label/DOROOT 8 1000000 1048576 2 0 0

## Generating Reports

It is possible to generate a report from the different quotas. The following command is used:

repquota -a

Which produces the following output

\*\*\* Report for user quotas on device /dev/disk/by-label/DOROOT

Block grace time: 7days; Inode grace time: 7days

Block limits File limits

User used soft hard grace used soft hard grace

------------------------------------------------------------------------------------

root -- 1118708 0 0 37093 0 0

daemon -- 68 0 0 4 0 0

man -- 9568 0 0 139 0 0

www-data -- 2908 0 0 15 0 0

nobody -- 0 0 0 1 0 0

libuuid -- 24 0 0 2 0 0

Debian-exim -- 44 0 0 10 0 0

mysql -- 30116 0 0 141 0 0

ftpuser -- 8 1000000 1048576 2 0 0

## Optional: Specify A Grace Period

To give current users some time to reduce their files on the droplet, a grace period can be configured. This is the allowed time a user can exceed their soft limit, while still staying under the hard limit. The grace time is configured using the following command [notice that this is system wide; no user specific configuration is possible]. The grace period can be expressed in seconds, minutes, hours, days, weeks or months.

edquota -t

The command gives the following output and specifies the different time unites that could be used. For this guide, a grace period of 7 days is used.

Grace period before enforcing soft limits for users:

Time units may be: days, hours, minutes, or seconds

Filesystem Block grace period Inode grace period

/dev/disk/by-label/DOROOT 7days 7days

### Conclusion

The quotas will be automatically updated and enforced when a user transfers/creates/moves/deletes a file/folder. Remember that the quota program works by looking at the owner or group of a specific file/folder. SSH users could escape the quotas by changing the owner or group of their files.

setenforce(1) SELinux Command Line documentation setenforce(1)

NAME

setenforce - modify the mode SELinux is running in.

SYNOPSIS

setenforce [ Enforcing | Permissive | 1 | 0 ]

DESCRIPTION

Use Enforcing or 1 to put SELinux in enforcing mode. Use Permissive or

0 to put SELinux in permissive mode. You need to modify /etc/grub.conf

or /etc/selinux/config to disable SELinux.

AUTHOR

Dan Walsh, <dwalsh@redhat.com>

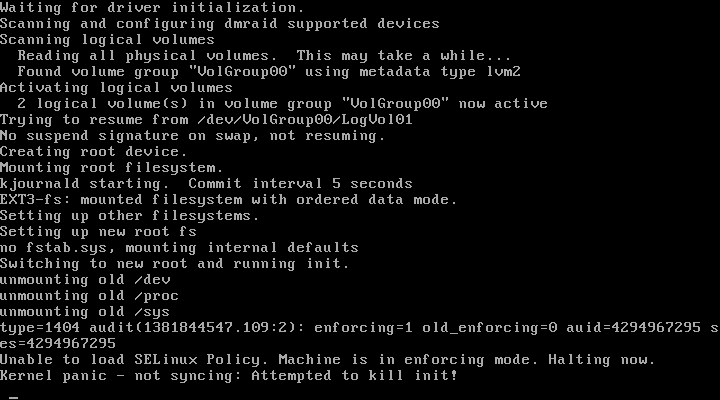
SEE ALSO

getenforce(8), selinuxenabled(8)

FILES

/etc/grub.conf, /etc/selinux/config

dwalsh@redhat.com 7 April 2004 setenforce(1)

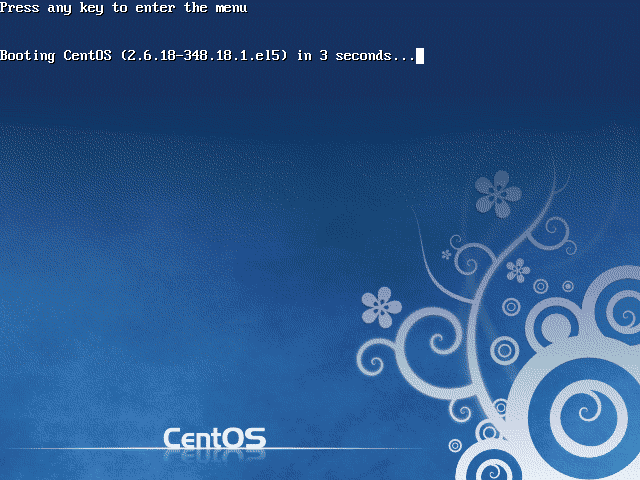
[](http://www.deep-silver.com/wp-content/uploads/2013/10/00.png)

# Kernel panic – not syncing Attempted to kill init Fixed

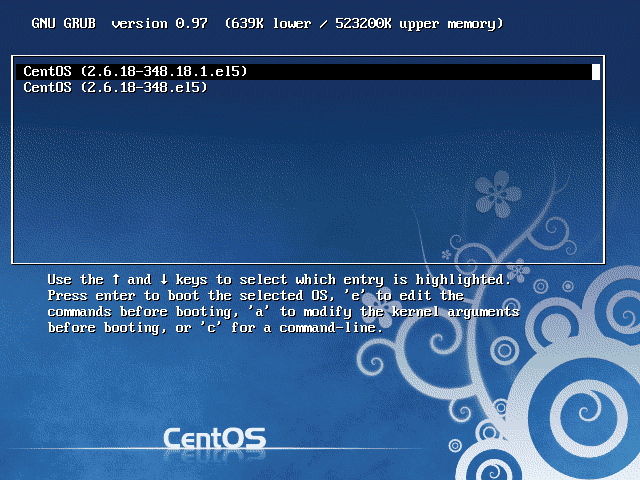
* [Linux](http://www.deep-silver.com/category/linux/)
* [CentOS](http://www.deep-silver.com/tag/centos/), [Linux](http://www.deep-silver.com/tag/linux/)
* [Leave a comment](http://www.deep-silver.com/kernel-panic-syncing-attempted-to-kill-init/#respond)

Kernel panic – not syncing Attempted to kill init Oh Boy we Hate Them….  
There are several reasons for the kernel to hang at boot with the line….  
**Fixing Kernel panic – not syncing: Attempted to kill init!**  
In most cases this cause by a faulty SELinux setting most likely on system file label.  
This can happened if you just upgraded your Linux box or played with SELinux while trying to remove or disable it.  
Before you can actually try to fix the issue with SELiunx that causing the system to hang at boot, you need to get your system booting up again.  
Here’s how to do it.  
In this example I will be using [CentOS](http://www.centos.org) 5.9 but these steps should apply to any recent CentOS/RHEL.

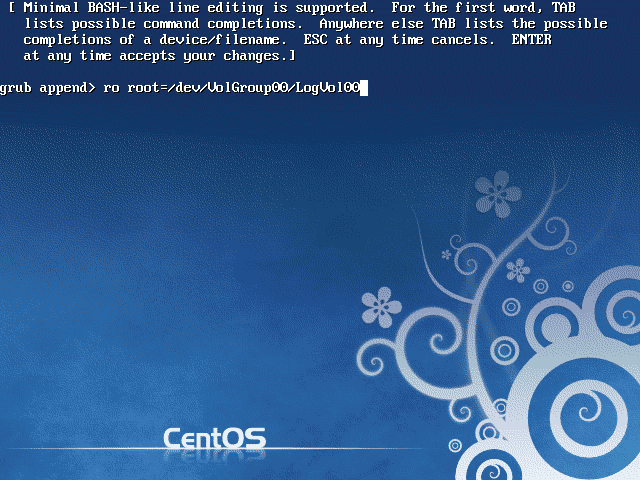
**STEP 01:**  
Restart the server an as soon as you see the boot screen as in the picture below (or similar depend on your distribution), press **Esc** button on the server keyboard.(you can actually press any key but we want to be safe)



Doing so will take you into Boot option menu as in the picture below.



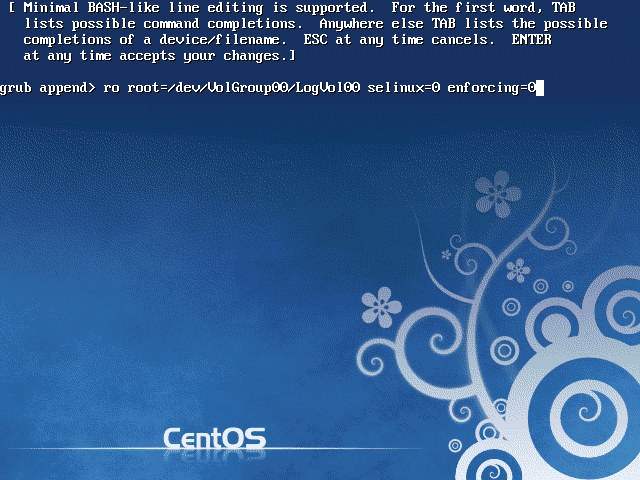
**STEP 02:**  
Using the ↑and ↓keys choose the first boot menu in this case CenOS (2.6.18.348.18.1.el5) and press **A** in the server keyboard to get the next screen as below.



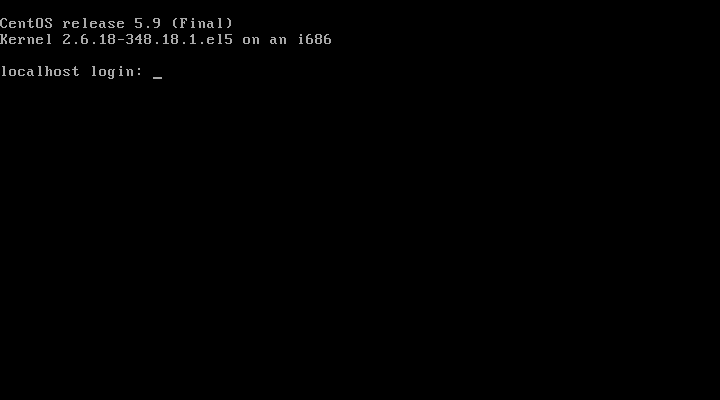
**STEP 03:**  
In this screen menu append this to the end of the line selinux=0 enforcing=0.  
So the whole line will look like this…  
**grub append> ro root=/dev/VolGroup00/LogVo100 selinux=0 enforcing=0**

Your line might look other then the example but the idea is to append the arguments to the end of the line.

Note that we disable both, SELinux and enforcing, in most cases you will only need one of them however by using both arguments we make sure the system boot even if SELinux is disabled but left enforcing or the targeted policy is missing or corrupted.



Once you done appending the arguments to the Kernel boot line, hit **Enter** and everything went well the system will continue to boot and you should get the terminal login screen as below.



Once you have managed to get into the terminal, is now time to look at the issue that may cause the kernel panic.  
Bare in mind that until now you have fixed noting but disabling the SELinux from causing the system to hang at boot. You now have to look into the cause of it.

Make sure you are logged in as root and run this command

# getenforce

Enforcing

The getenforce command returns Permissive when SELinux is enabled, but SELinux policy rules are not enforced or returns Disabled if SELinux is disabled.  
Running The **sestatus** command returns the SELinux status and the policy being used

# sestatus

SELinux status: enabled

SELinuxfs mount: /selinux

Current mode: enforcing

Mode from config file: enforcing

Policy version: 23

Policy from config file: targeted

In our case, the SELinux is enabled and running in enforcing mode using targeted policy.  
As we already figure it up, in most cases kernel hang cause by a faulty SELinux setting most likely on system file label, you should make sure that system label are corrected.  
The easiest way to do so is by relabeling the whole system using the SELinux its self, but in order for it to be able to do, you have to make sure that SELinux is running and in permissive mode.  
First use the following command to confirm that the SELinux packages are installed.

# rpm -qa | grep selinux

libselinux-python-1.33.4-5.7.el5

libselinux-utils-1.33.4-5.7.el5

libselinux-devel-1.33.4-5.7.el5

selinux-policy-targeted-2.4.6-338.el5

libselinux-1.33.4-5.7.el5

selinux-policy-2.4.6-338.el5

Assuming the following packages are installed: selinux-policy-targeted, selinux-policy, libselinux, libselinux-python, libselinux-utils, policycoreutils  
If one of the above is missing or corrupted install it or reinstall it using Yum command. if you want to be safe just reinstall using this command

# yum -y install libselinux libselinux-python libselinux-utils policycoreutils selinux-policy selinux-policy-targeted

Once you confirmed that all required packages are installed, run this command and make sure that SELinux is configured to run in permissive mode

# cat /etc/selinux/config

# This file controls the state of SELinux on the system.

# SELINUX= can take one of these three values:

# enforcing - SELinux security policy is enforced.

# permissive - SELinux prints warnings instead of enforcing.

# disabled - No SELinux policy is loaded.

SELINUX=permissive

# SELINUXTYPE= can take one of these two values:

# targeted - Targeted processes are protected,

# mls - Multi Level Security protection.

SELINUXTYPE=targeted

**VERY IMPORTANT:**  
If in the above the line show **SELINUX=enforcing** or **SELINUX=disabled**, edit the file using your favorite editor make sure sure it say **SELINUX=permissive** as above and then reboot the system before you do the next step.

Run the below command and make sure SELinux is runing and in permissive mode.

# setenforce 0

# getinforce

permissive

With the above done, run this command to create a file on root system named .autorelabel and force the system to reboot.

# touch /.autorelabel

# reboot

During the next boot, if SELinux find the file ./autorelable it will relabel all system files and then delete the autorelable file.

\*\*\* Warning -- SELinux targeted policy relabel is required.

\*\*\* Relabeling could take a very long time, depending on file

\*\*\* system size and speed of hard drives.

\*\*\*\*

Once the system is up and running again, you should run this command to make sure that SELinux did not deny actions during the last boot.

# grep "SELinux is preventing" /var/log/messages

If SELinux did not deny actions during the last boot, this command does not return any output and it is now safe to go ahead and enforce SELinux.  
Edit the file **/etc/selinux/config** and this time append **SELINUX=enforcing**.

Reboot your system and make sure SELinux is running in enforcing mode

# getenforce

Enforcing

DONE!

### Subscribe

Subscribe to our e-mail newsletter to receive updates.

#### Related Posts:

* [Install Ruby 1.8.7 On CentOS 5.9 Using Yum](http://www.deep-silver.com/install-ruby-1-8-7-on-centos-5-9-using-yum-repository/)

[← Scammer Aljrabli Michael – WARNING](http://www.deep-silver.com/scammer-aljrabli-michael-warning/)

##### No comments yet.

### Leave a Reply

You must be [logged in](http://www.deep-silver.com/login?redirect_to=http%3A%2F%2Fwww.deep-silver.com%2Fkernel-panic-syncing-attempted-to-kill-init%2F) to post a comment.

### Categories

* [Handcraft](http://www.deep-silver.com/category/handcraft/)
* [Linux](http://www.deep-silver.com/category/linux/)
* [Others](http://www.deep-silver.com/category/others/)

### Recent Posts

* [Kernel panic – not syncing Attempted to kill init Fixed](http://www.deep-silver.com/kernel-panic-syncing-attempted-to-kill-init/)
* [Scammer Aljrabli Michael – WARNING](http://www.deep-silver.com/scammer-aljrabli-michael-warning/)
* [Install Ruby 1.8.7 On CentOS 5.9 Using Yum](http://www.deep-silver.com/install-ruby-1-8-7-on-centos-5-9-using-yum-repository/)
* [Silver Craft Heirlooms](http://www.deep-silver.com/silver-craft-heirlooms/)
* [Handcrafted jewelry design](http://www.deep-silver.com/handcrafted-jewelry-design/)

### Recent Comments

* ray ban on [Scammer Aljrabli Michael – WARNING](http://www.deep-silver.com/scammer-aljrabli-michael-warning/#comment-188)
* maillot de football pas cher on [Scammer Aljrabli Michael – WARNING](http://www.deep-silver.com/scammer-aljrabli-michael-warning/#comment-181)
* wholesale silver jewellery on [Install Ruby 1.8.7 On CentOS 5.9 Using Yum](http://www.deep-silver.com/install-ruby-1-8-7-on-centos-5-9-using-yum-repository/#comment-139)
* r4 3ds on [Handcrafted jewelry design](http://www.deep-silver.com/handcrafted-jewelry-design/#comment-98)
* Carol on [Handcrafted jewelry design](http://www.deep-silver.com/handcrafted-jewelry-design/#comment-74)

### Archives

* [October 2013](http://www.deep-silver.com/2013/10/)
* [March 2013](http://www.deep-silver.com/2013/03/)
* [February 2013](http://www.deep-silver.com/2013/02/)
* [January 2013](http://www.deep-silver.com/2013/01/)

# kernel panic not syncing attempted to kill init vmware centos 6.2

I recently came across an issue where I had a vmware ESXI server that had a fresh CentOS 6.2 install that had been rebooted and gave the cryptic error:

kernel panic not syncing attempted to kill init!

Most sites that you are googling will talk about either changing the permissions on file, changing the Disk controller to LSI or that you need to edit your vmx file?!

In my case, pressing "Esc" during the boot process showed what was really hanging up the system loading. SeLinux was angry and needed to be disabled.

To do this, just boot off the same CD that you used to install the OS, choose Rescue and answer the first few questions normally, then when it drops you into a shell use:

vi /mnt/sysimage/etc/selinux/config

Set Selinux to 0

|  |  |
| --- | --- |
| down vote [favorite](http://serverfault.com/questions/603875/server-wont-boot-kernel-panic-not-syncing) | After a reboot of a CentOS 6.2 server we are getting a kernel panic with the following error:  Kernel panic - not syncing: Attempted to kill init!  Panic occurred, switching back to text console  When passing **init=option selinux=0** to the kernel at boot the server will boot without a problem. I have tried searching for what init=option does but I have not been able to find any recourses explaining it.   * Could someone explain what this parameter does? * Also, as this is a production server are there any negatives/risks in running with the init=option?   I have tried to remake the initramfs via dracut which did not work and received the above error upon boot.  As this is a clustered server and the primary server should be identical I have copied the /boot drive from the other server but this did not solve the problem either. (The other server does not have this behavior)  Restoring /boot from backup did not work either.  Could you please help me investigate this problem   * What causes this kernel panic? * How would you go about resolving this issue? * What logs (if any) would tell you more information? * As this is an older install, would an update/upgrade potentially fix this issue?   I would also like to add that the only other possible related issue was that the root disk which was raid mirrored failed and was replaced before this happened.  Thanks for your help! |

## [Kernel panic – not syncing: Attempted to kill init! ” Error](http://alexhunt86.wordpress.com/2013/11/08/kernel-panic-not-syncing-attempted-to-kill-init-error/)

Posted on [November 8, 2013](http://alexhunt86.wordpress.com/2013/11/08/kernel-panic-not-syncing-attempted-to-kill-init-error/) by [Alex Hunt](http://alexhunt86.wordpress.com/author/alexhunt86/)

Rate This

If you get kernel panic after system reboot…dont panic just follow the steps as shown below :  
1. Once you are at the grub screen, press c for command line.  
2. On command line execute the following commands.  
**find /grub/stage2**  
**find /etc/hosts**

This will provide you the boot partition & the root partition nos. in the format (hd0,2) etc  
If the root partition & boot partitions are same then execute the following commands.  
**root (hd0,2)**  
**kernel /boot/vmlinuz-kernel version ro root=/dev/hda3**  
**initrd /boot/initrd-version**  
**boot**

if the boot & root partitions are different then replace the root=/dev/hda3 with the value you get for root partition i.e. if you get (hd0,5) then replace the value with /dev/hda6.  
If you still get the kernel panic message, recreate the initrd image using the mkinitrd command.

Bottom of Form