Basic Linux Command Line Interface Guide

This basic Linux Command-Line Interface (CLI) Guide provides a general explanation of commonly used Bash shell commands for the Barracuda NG Firewall. You can access the command-line interface by connecting to the hostname with a terminal utility (such as PuTTY). It is strongly recommended that the administrator reads the manual (man) page for bash (# man bash) after connecting to the system, and any man pages for commands listed in this article where a further explanation is needed. It is also important to remember that more options are available for commands than what are outlined in the following list.

In this article:

pwd

To verify the directory path that you are currently working in, use the **pwd** command. It stands for "Print name of current Working Directory." From your shell, enter:

 # pwd

ls

To display the contents of the current or specified directory, use the **ls** command. It stands for "list directory contents" and can be thought of as "List to Screen." To display the contents of the current directory, enter:

 # ls

To print the contents of the directory in long list format, add the **-l** option:

 # ls -l

Notice the different outputs between running ls and ls -l.

mount

To attach a file system to a device or several devices, use the **mount** command. You can also run the command by itself to view the mounted partitions:

 # mount

(/dev/sda#   shows the mounted devices.)

df

To view the total and available disk space for the file system, use the **df** command. It stands for "report file system disk space usage" but is commonly referred to as "disk free." To print the output in a "human readable" format, add the **–h** option:

 # df -h

cd

To "change directories" or switch from one directory to another, use the **cd** command.

For example:

1. Change to the /home/remote directory.
   # cd /home/product
2. Print the current working directory to verify that you are now in the /home/product directory.
   # pwd
   You should see the following output:
   /home/product

touch

The **touch** command is "officially" used to change a file's time stamp, but you can use it to create files.

For example, you can create a file named *myfile* in the /mail/tmp directory.

> The /mail/tmp directory is the largest partition and can be used to write files to.

1. Change to the /mail/tmp directory.
   #cd /mail/tmp
2. Create a file named myfile.
   # touch myfile
3. List the directory contents to verify that *myfile* has been created.
   # ls
   In the output, myfile should be listed.

chmod

To change a file's permissions, use the **chmod** command. When changing the file permissions with **chmod**, the permissions are binary counted. Before using the **chmod** command, identify what permissions should be given to which group. For example, you may choose to grant the following permissions:

- Owner — Read, Write, and Execute
- Group — Read and Execute
- Other — Read and Execute

For example, to change the permissions of a file named *myfile*:

1. Allow read, write, and execute permissions for the file.
   # chmod 777 myfile
2. List the directory contents in long list format to view the file permissions.
   # ls -l
   You should see output that is similar to the following:
   -rwxrwxrwx 1 root root   0 2013-03-06 12:17 myfile

vi

To create or edit plain text documents or programs, use the **vi** or **vim** (**V**i **Im**proved) command. The vi editor has two modes:

- **Insert mode** — Lets you use input text into the document or program.
- **Command mode** — Lets you use commands. It is recommended that you read the man page for a comprehensive list of all commands.

You can also learn more about using **vi** by reading the Vimtutor and following the steps outlined in the files. To access the Vimtutor, run:

 # vimtutor

The following table lists common commands that you can use with vi:

Command	Action
:wq	Exit the vi editor and save changes.
:q!	Exit the vi editor without saving changes.
i	Insert before cursor.
Esc	Enter command mode.
x	Delete character under cursor.
d+d	Delete a line in the file.
?	Find a word going backwards.
/	Find a word going forwards.

For example, to edit a file named *myfile* and save your changes:

1. Open the file in the vi editor.
   # vi myfile
2. Press the **I** key to enter the insert mode. When you are in insert mode, --INSERT -- is displayed at the lower left hand corner of the shell.
3. Type the following:
   `#!/bin/bash echo "hello, $USER. I will list the files in the current directory" echo "The current directory is, $PWD" ls # list files`
4. Press the **Esc** key to return to command mode.
5. Save and exit the file. Enter:
   :wq
6. To run this file, enter:
   # ./myfile.

To later remove a line from *myfile* without saving your changes:

1. Open the file in the vi editor:
   # vi myfile
2. Put the cursor in the line that you want to delete and then press **d**+**d**.
3. Exit the file without saving. Enter:
   :q!
4. Run the file.
   # ./myfile.
   The output should not have changed.

wc

You can use the **wc** command with various options to view information about a file:

Command	Output
wc –c <file name>	Prints the number of bytes in the specified file.
wc –m <file name>	Prints the number of characters in the specified file.
wc –w <file name>	Prints the number of words in the specified file.
wc –l <file name>	Prints the number of lines in the specified file.

head

Use the **head** command to print the first 10 lines of a file. For example, to print the first ten lines of a file named *myfile*:

head myfile

To print lines based on other criteria, you can add options to the **head** command.

tail

Use the **tail** command to print the last 10 lines of a file. Usually, the command is used with the **–f** option to view a log file as it is being written. For example, enter:

tail –f /mail/log/debug

To separate the last 10 lines that have been written from the appended output, press the **Enter** key.

To exit out of the tail, press **Ctrl**+**C**.

cat

To "concatenate" a file and print the standard output, use the **cat** command. It is one of

many commands that you can use to view the contents of a file. For example, to view *myfile*:

 # cat myfile

> It is not recommended that you use the **cat** command to view large files.

Using cat to create a file:

You can also use the **cat** command to create a file and write text to that file.

For example, to create a file named *usernamecatfile*:

1. Create the *usernamecatfile* file.
   # cat > usernamecatfile
   The **>** will redirect the output to a file. You can now write to the file.
2. Type the following information:
   This is my example of an output redirect
3. Press the **Enter** key to create a new line, and then press **Ctrl**+**D** to quit the cat.
4. Cat the file you created to verify the file was written to.

Using cat to output to a file:

You can also use the **cat** command to redirect the output to a new file. This is essentially a method of copying a file.

For example:

1. Redirect the output of *usernamecatfile* to *usernamecatredir*.
   # cat usernamecatfile > usernamecatredir
2. Cat the new file and review the output.

less

To view a file incrementally, use the **less** command. This command is especially useful on larger files and is more useful than the **more** command because it allows for forward and backward movement. When viewing the file, using **/pattern** will search forward within the file for the specified regex pattern. When viewing the file, using **?pattern** will search backwards within the file for the specified regex pattern.

For example:

1. View *myfile*.
   # less myfile
2. Review the output. Notice how different it is from the output from using the **cat** command.
3. To exit, press the **Q** key.

grep

To search a file for a given pattern, use the **grep** command. If a line has the requested pattern, the entire line is printed.

For example:

1. Run:
   # grep TIMING /mail/log/info
2. Review the output.

The **grep** command is an excellent tool with tailing for a specific pattern, sending an output to the grep with a pipe, or just greping a file for the pattern. Examples of this include:

- *tail –f /mail/log/info | grep "domain.com"*
- *less /mail/log/info | grep "domain.com"*
- *grep "domain.com" /mail/log/info*

I pipe

To send the output of one command to the input of another, use the **pipe (|)** operator.

For example:

1. Use the **less** command to view the *info* file.
   # less /mail/log/info
2. Press the **Q** key to exit.
3. Press the **up** arrow key to display the previous command (*# less /mail/log/info*) and then add:
   …/info| grep TIMING
4. Review the output. Notice the difference between the output for the **less** command and the **grep** command.

mv

To move a file to another directory, use the **mv** command.

For example:

1. Move myfile from the */mail/tmp* directory to the */home/remote* directory.
   # *mv /mail/tmp/myfile /home/remote/myfile.*
2. List the contents of the */mail/tmp* directory to verify that *myfile* is no longer in it.
   # *ls /mail/tmp*
3. Change to the */home/remote* directory and then list its contents.
   # *cd /home/remote/*
   # *ls*

cp

To make a copy of a file or create a copy of a file in another location, use the **cp** command. It is recommended that you create a copy of any file before modifying it in any way.

For example:

1. Create a copy of *myfile* that is named *myfile2*.
   # *cp myfile myfile2*
2. List the files in the directory to verify that myfile2 was created.

For example, to create a copy of a file in another location:

1. Create a copy of myfile2 in another directory.
   # *cp /home/remote/myfile2 /tmp/myfile2*
2. List the files in the */home/remote* and */tmp/* directories. Both directories should list *myfile2*.

rm

To "remove" a file or directory, use the **rm** command. It is good practice to add the file or directory to be deleted before putting the **rm** command into the string.

For example:

1. Type the following command <u>without</u> pressing **Enter**:
   # /tmp/myfile2
2. Use the left arrow key to go to the beginning of the line and add rm.
   # rm /tmp/myfile2
3. Press **Enter**.
4. List the files in the *tmp* directory to verify that the file has been removed.

ln

To "make links between files," use the **ln** command. Many files and file system locations are already symbolically linked on Barracuda products.

For example:

1. Symbolically link */home/remote/myfile* to */mail/tmp/seemyfile*.
   # ln –s /home/remote/myfile /mail/tmp/seemyfile
2. Run a "long list" for */mail/tmp* to see the symbolic link between *seemyfile* and */home/remote/seemyfile.*
3. Cat both files to see how the content is similar.

find

To search for files in a directory hierarchy, use the **find** command. This command is especially useful when you do not know the name or location of a file.

For example, to search for files in the */home* directory that have *yfile* in their name:

*# find /home –iname '*yfile*'*

The output prints all files in the */home* directory that have the simple pattern of *yfile* in their name.

which

To display the full path of shell commands (executables or scripts), use the **which** command.

For example:

which qm.pl

alias

To create shortcuts by defining aliases, use the **alias** command.

For example:

1. Create an alias named *homepers* for listing the contents of the */home* directory in long list format.
   # alias homepers="ls –l /home"
2. Enter the alias to verify that it works.
   # homepers
   The contents of the */home* directory should be listed in long list format.

The alias is removed when you log out of the shell. To permanently keep the alias, add it to the ~remote/.bashrc file.

uptime

To view how long the system has been running, use the **uptime** command.

ps

To report a snapshot of the current running processes, use the **ps** command. It is useful for getting the process ID to send kill signals. If you are looking for a specific process, add a pipe and the requested process.

For example:

 # ps fax

If you are looking for a specific process, you can pipe the **ps** command to a grep for a pattern.

For example:

 # ps fax | grep mysql

Compare the outputs from both **ps** commands.

top

To display the top processes and memory used, use the **top** command. To sort by memory (%MEM), press **Shift**+**M**.

kill

To send a kill signal to a process ID, use the **kill** command. The **-9** is a common option and is used to "kill all". For example:

 # kill -9 <process ID>

strace

To trace system calls and signals, use the **strace** command. This command essentially intercepts and records the system calls of a process and the signals received by a process. Before using the **strace** command, use the top command to get the required process ID. After getting the process ID, use the following syntax to trace the process:

 # strace –p <process ID>

> Reading a strace can be difficult due to the "noise" that is reported. When a program is run, there are many standard system files that all write to the strace. The **strace** command is especially useful for identifying what a process is doing when a process is "hogging" up CPU resources.

ping

To send an ICMP ECHO_REQUEST to a host and listen for a response, use the **ping** command. This is a good tool for verifying that the host is responding to requests. However, make sure that ICMP has not been disabled on a network or the host; otherwise, no response is provided and the request will time out.

For example:

1. Ping your company website.
   # ping <company website address>
   You should receive responses.
2. Press **Ctrl**+**C** to kill the ping to the website.
3. Ping *www.aol.com*.
   # ping www.aol.com
   This request should not return any responses.
4. Press **Ctrl**+**C** to kill the ping to AOL.
5. Review the output.

ldapsearch

To query an LDAP library, use the **ldapsearch** command.

This command is only available on a Barracuda NG Control Center.

Example 1

Search for a specific LDAP user, running the following command:

 # ldapsearch -h ldap.barracuda.com -p 389 -D 'username@cudanet.local' -b 'dc=cudanet,dc=local' -W "mail=bghodd"*

When prompted, the administrator must provide the password.

Example 2

Create an LDAP Data Interchange Format (LDIF) file, running the following command:

 # ldapsearch -h ldap.barracuda.com -p 389 -D 'username@cudanet.local ' -b 'dc=cudanet,dc=local' -W -LLL > /tmp/thisboxldif.ldif

When prompted, the administrator must provide the password. The LDIF file is written to the *tmp* directory and can be copied and sent to a customer.

dig

To query a DNS server for a record, use the **dig** command. It is a DNS lookup utility. However, the **dig** command relies on DNS and will not reference the */etc/hosts* file to resolve a name to an IP address.

Example:

- # dig barracuda.com (general DNS query for the barracuda.com a record)
- # dig @4.2.2.2 barracuda.com (DNS query for the barracuda.com a record but at a specific server (4.2.2.2))
- # dig mx barracuda.com (DNS query for a Mail eXchange (MX) record)
- # dig –x 64.235.145.81 (DNS query for a reverse lookup of an IP address)

nslookup

To query Internet domain name servers, use the **nslookup** command. It has two modes

- Interactive mode — Allows the user to query a name server for information about various hosts and domains.
- Non-interactive mode — Prints just the name and request information for a host or domain.

For example, to query the MX record for barracuda.com in interactive mode:

1. Start the nslookup.
   # nslookup
2. Specify that you are looking up the MX record.
   > set type=mx
3. Specify the domain.
   > barracuda.com
4. Review the output.
5. To exit nslookup, enter:
   exit

For example, to query bsf01.yourwebsite.com in non-interactive mode:

1. Enter:
   # nslookup bsf01.yourwebsite.com

2. Review the output.

host

To perform DNS lookups, use the **host** command. It is normally used to convert names to IP addresses, and/or IP addresses to names. This command will look in the */etc/host* file.

For example:

1. Look up 64.235.145.81.
   # host 64.235.145.81
   This is the IP address of the network host.
2. Review the output.
3. Look up bsh01.yourwebsite.com.
   # host bsh01.yourwebsite.com
4. Review the output.

telnet

To communicate with another host on a given port, use the **telnet** command.

ifconfig

To view or configure a kernel-resident network interface, use the **ifconfig** command.

traceroute

To print the route that the packets take to get to a network host, use the **traceroute** command. It should be noted that **traceroute** may be unreliable because it is a connectionless (UDP in Unix and ICMP in Windows) connection.

From your shell, enter:

 # traceroute <IP address of the network host>

GET

To send requests to www and local file system servers, use the **GET** command. This tool may be helpful if a customer states that they are not filtering outbound requests "at all". The response will indicate whether or not web traffic is being filtered/blocked; it may be best practice to try the **GET** command with a website that would normally be blocked on a corporate or government network.

For example, enter:

 # GET www.disney.com

hwtool

To read out the currently installed BIOS version number for a running Barracuda NG Firewall without a reboot, use the **hwtool** command.

From your shell, enter:

hwtool -b

fsck

To check the filesystem for errors use the **fsck** command depending on what filesystem you are using (ext2 or ext3).

> If your appliance uses IDE harddisks replace **sdaX** with **hdaX**.

/bin/umount /dev/sda1 /bin/umount /dev/sda2 /bin/umount /dev/sda3 /bin/umount /dev/sda4 /bin/umount /dev/sda5 /bin/umount /dev/sda6

ext2 file system:

/sbin/fsck.ext2 -y /dev/sda1 /sbin/fsck.ext2 -y /dev/sda2 /sbin/fsck.ext2 -y /dev/sda3 /sbin/fsck.ext2 -y /dev/sda4 /sbin/fsck.ext2 -y /dev/sda5 /sbin/fsck.ext2 -y /dev/sda6

ext3 file system:

```
/sbin/fsck.ext3 -y /dev/sda1 /sbin/fsck.ext3 -y /dev/sda2 /sbin/fsck.ext3 -y
/dev/sda3 /sbin/fsck.ext3 -y /dev/sda4 /sbin/fsck.ext3 -y /dev/sda5
/sbin/fsck.ext3 -y /dev/sda6
```

Keyboard Shortcuts

To save time while you are typing at the command line, use the following keyboard shortcuts:

Keys	Action
Ctrl+A	Go to the beginning of the command line.
Ctrl+E	Go to the end of the command line.
Alt+B	Move the cursor backwards by one word.
Alt+F	Move the cursor forward by one word.
Ctrl+W	Remove the previous word, from the cursor to the previous word boundary.
Ctrl+D	Delete the character under the line.
Ctrl+K	Remove the remainder of the line, from the cursor to the end of the line.
Ctrl+U	Clear the line from the cursor to the beginning of the line.
Tab	Autocomplete a line of text.
up/down arrows	Move through previously used commands.