

# Getting Started with Linux Commands

Overview

Common Linux commands

Run multiple commands on the same line

Common configuration files and directories

Additional documentation

## Overview

While cPanel & WHM automates many server administration tasks, familiarity with the Linux® command line can prove useful for both WHM and cPanel users. This documentation gives a brief overview of some basic Linux commands that you may wish to use as you manage your website or server.

**Note:**

For steps to access the command line, read our [How to Access the Command Line](#) documentation.

## Common Linux commands

**Note:**

To execute a command, enter the command with any options or arguments that it requires and press Enter.

The following table lists some basic Linux commands and their functions:

Command	Description
<code>ls</code>	<p>List files and directories that exist within your current directory. This command resembles the <code>dir</code> command in Windows®.</p> <p>To view dotfiles (filenames that begin with a period) and additional file and directory details, add the <code>-al</code> options to the command:</p> <pre>ls -al</pre>
<code>cd location</code>	<p>Navigate between directories.</p> <p>Replace <code>location</code> with the path to the directory that you wish to navigate to. For example, to navigate to the <code>/usr/local/apache/</code> directory, run the following command:</p> <pre>cd /usr/local/apache/</pre>

```
cat filename
```

Print the contents of the specified file to the CLI.

Replace `filename` with the relative path to the file that you wish to view. For example, to print the contents of the `filename.txt` file, run the following command:

```
cat filename.txt
```

**Note:**

To view the data that currently displays on your server's console screen, run the following command:

```
cat /dev/vcs1
```

```
tail filename
```

Print the last 20 lines of a file to the command line interface (CLI).

Replace `filename` with the relative path to the file that you wish to view. For example, to print the last 20 lines of the `filename.txt` file, run the following command:

```
tail filename.txt
```

You can add an argument to change the number of lines that this command prints. For example, to print the last 100 lines of the `filename.txt` file, run the following command:

```
tail -100 filename.txt
```

```
more filename
```

Print the contents of a file to the CLI, one screen at a time.

Replace `filename` with the relative path to the file that you wish to view. For example, to print the contents of the `filename.txt` file one screen at a time, run the following command:

```
more filename.txt
```

```
pico filename
```

Open the specified file in the `pico` text editor.

Replace `filename` with the relative path to the file that you wish to edit. For example, to open the `filename.txt` file in the `pico` editor, run the following command:

```
pico filename.txt
```

<code>vi filename</code>	<p>Open the specified file in the <code>vi</code> text editor.</p> <p>Replace <code>filename</code> with the relative path to the file that you wish to edit. For example, to open the <code>filename.txt</code> file in the <code>vi</code> editor, run the following command:</p> <pre>vi filename.txt</pre>
<code>grep string filename</code>	<p>Search for a string in a specified file, and prints each line that contains a match to the CLI.</p> <p>Replace <code>string</code> with a single word, or multiple words within single quotes ( <code>'</code> ). Replace <code>filename</code> with the relative path to the file that you wish to search. For example, to search for the string <code>coffee filters</code> in the <code>grocerylist.txt</code> file, run the following command:</p> <pre>grep 'coffee filters' grocerylist.txt</pre>
<code>touch filename</code>	<p>Create an empty file in the specified location.</p> <p>Replace <code>filename</code> with the relative path to the file that you wish to create. For example, to create an empty <code>example.txt</code> file, run the following command:</p> <pre>touch example.txt</pre>
<code>ln -s file1 file2</code>	<p>Create a symbolic link between the two specified files.</p> <p>Replace <code>file1</code> with the relative path to the existing file, and <code>file2</code> with the relative path to the new symbolic link file. For example, to create the <code>symlink-file.txt</code> file and point it to the <code>/pointtome/file.txt</code> file, run the following command:</p> <pre>ln -s /pointtome/file.txt symlink-file.txt</pre>
<code>rm filename</code>	<p>Delete the specified file. After you run this command, the system prompts you to confirm the file's deletion.</p> <p>Replace <code>filename</code> with the relative path to the file that you wish to delete. For example, to delete the <code>trash.txt</code> file, run the following command:</p> <pre>rm trash.txt</pre>
<code>last</code>	List which users recently logged in and the timestamp for each login.
<code>w</code>	List currently logged-in users and the location from which they logged in.
<code>netstat</code>	List all of the server's current network connections.

<code>file filename</code>	<p>Guess a file's type, based on the file's contents.</p> <p>Replace <code>filename</code> with the relative path to the file for which you want the system to guess the type. For example, to cause the system to guess the type for the <code>example.txt</code> file, run the following command:</p> <pre>file filename</pre>
<code>du</code>	<p>Show the system's current disk usage for each directory and subdirectory.</p>
<code>wc filename</code>	<p>Display the word count for a specific file.</p> <p>Replace <code>filename</code> with the relative path to the file for which you wish to view a word count. For example, to display a word count for the <code>example.txt</code> file, run the following command:</p> <pre>wc example.txt</pre>
<code>cp file1 file2</code>	<p>Copy a file into a new file.</p> <p>Replace <code>file1</code> with the relative path to the existing file, and <code>file2</code> with the relative path to the new copy file that you wish to create. For example, to copy the contents of the <code>original.txt</code> file to the <code>/copies/duplicate.txt</code> file, run the following command:</p> <pre>cp original.txt /copies/duplicate.txt</pre>
<code>chmod permissions filename</code>	<p>Change a file's octal permissions.</p> <p>Replace <code>permissions</code> with the three-digit octal permissions that you wish to grant to the file, and replace <code>filename</code> with the relative path to the file for which you wish to alter the permissions. For example, to change the permissions of the <code>myfile.txt</code> file to 755, run the following command:</p> <pre>chmod 755 myfile.txt</pre> <p>For more information, read the Wikipedia article about the <code>chmod</code> command.</p>
<code>chown user:group filename</code>	<p>Change a file's user and group ownership.</p> <p>Replace <code>user</code> with the user to whom you wish to grant ownership of the file, <code>group</code> with the group name, and <code>filename</code> with the relative path to the file. For example, to grant the user <code>joe</code> in the group <code>joesgroup</code> ownership of the <code>joesfile.txt</code> file, run the following command:</p> <pre>chown joe:joesgroup joesfile.txt</pre>

<pre>whereis name</pre>	<p>Query applications that match the <code>name</code> value.</p> <p>You can find the most common applications in the following locations:</p> <ul style="list-style-type: none"> <li>• <code>/usr/sbin/sendmail</code></li> <li>• <code>/usr/bin/perl</code></li> <li>• <code>/bin/mail</code></li> <li>• <code>/usr/bin/php</code></li> </ul>
<pre>ps</pre>	<p>Return information about the server's current processes.</p> <p>To view all of the running processes, run one of the following commands:</p> <pre>ps -auxww ps -cef</pre>

## Run multiple commands on the same line

Various command-line tasks may require that you use different commands on the same line. Linux includes easy methods to perform these tasks.

- Use the pipe character (`|`) to retrieve data from one program and "pipe" it to another program.
- Use a single greater-than bracket (`>`) to create a new file if the file does not already exist, or to overwrite any existing content if the file does exist.
- Use a double greater-than bracket (`>>`) to create a new file if the file does not already exist, or to append the new data to the file if the file does exist.
- Use a single less-than bracket (`<`) to send input from a file to a command.

The table below lists examples of how to combine tasks into a single line:

Command	Description
<pre>grep User /usr/local/apache/conf/httpd.conf   more</pre>	<p>This command searches for all of the lines in the <code>httpd.conf</code> file that match the <code>user</code> search term, and then prints the results to your terminal one screen at a time.</p>
<pre>last -a &gt; /root/lastlogins.tmp</pre>	<p>This command prints all of the current login history to the <code>/root/lastlogins.tmp</code> file.</p>
<pre>tail -10000 /var/log/exim_mainlog   grep 'example\.com'   more</pre>	<p>This command finds the last 10,000 lines from the <code>/var/log/exim_mainlog</code> file, searches those lines for all occurrences of the string <code>example.com</code>, and then prints the search results to your terminal one screen at a time.</p> <div data-bbox="808 1633 1458 1785" style="border: 1px solid #f9e79f; padding: 5px;"> <p><b>Note:</b> The system treats periods (<code>.</code>) in a command as wildcard characters. Precede each period with a backslash (<code>\</code>) to instruct <code>grep</code> to interpret the period literally.</p> </div>

<pre>netstat -an   grep :80   wc -l</pre>	<p>This command shows the number of active connections to Apache (Apache's <code>httpd</code> daemon runs on port 80).</p>
<pre>mysql --skip-column-names --batch -e 'show processlist'   wc -l</pre>	<p>This command lists the number of MySQL threads. If subselect expressions start new threads, the output of the <code>show processlist</code> command includes them.</p>

## Common configuration files and directories

You can find common configuration files and directories in the following locations on your server:

Service	Locations
Exim	<pre>/etc/exim.conf /var/log/exim_mainlog /var/log/exim_rejectlog /etc/valiases/ /etc/vfilters/ home/username/.forward</pre>
MySQL®	<pre>/root/.my.cnf /etc/my.cnf /var/lib/mysql/</pre>
Apache	<pre>/usr/local/apache/conf/httpd.conf /usr/local/apache/domlogs/</pre>
System	<pre>/var/log/messages /var/log/dmesg</pre>
ProFTPD	<pre>/etc/proftpd.conf /var/log/xferlog /etc/proftpd/</pre>

SSH

`/etc/ssh/sshd_config`

## Additional documentation

Suggested documentation For cPanel users For WHM users For developers

- [The Linux Knowledge Base and Tutorial](#)
- [LinuxPlanet Tutorials](#)
- [RedHat's Documentation](#)
- [How to Access the Command Line](#)
- [How to Secure SSH](#)
- [How to Access the Command Line](#)
- [Getting Started with Linux Commands](#)
- [How to Disable Prelinking](#)
- [cPanel API 2 Functions - SSH::genkey\\_legacy](#)
- [cPanel API 2 Functions - SSH::delkey](#)
- [cPanel API 2 Functions - SSH::fetchkey](#)
- [cPanel API 1 Modules - SSH](#)
- [cPanel API 2 Functions - SSH::authkey](#)