**Searching by Name**

Of course, the most common method to look for a file is using its name. To run a simple search query using the name of the file, use the **find** command like this:

**find . -name my-file**

We used the **-name** option, and searched for a file called **my-file**. Note that we started the search in our current directory by using the **. (dot)** argument.

Keep in mind that the **-name** argument looks for case-sensitive terms in Linux. If you know the name of the file, but are not sure about its case-sensitivity, use the following **find** command:

**find . -iname my-file**

You can also search for all files without a certain keyword in their name. There are two ways to do this. The first method involves using the **–not** keyword in the following manner:

**find . -not -name my-file**

Second, we can use the exclamation symbol (**!**). However, it has to be preceded by the escape identifier (\) to let Linux know that this is the part of the **find** command.

**find . \! -name my-file**

You can look for multiple files with a common format like **.txt** as well:

**find . -name “\*.txt”**

This will list down all the text files starting with the current folder.

Lastly, if you want to find a certain file by name and remove it, use the**-delete** argument after the file name:

**find . -name my-file -delete**

**Searching by Type**

Linux allows users to list all information based on their types. There are several filters that you can use:

* **d** – directory or folder
* **f** – normal file
* **l** – symbolic link
* **c** – character devices
* **b** – block devices

A simple example of using a file type can be seen below:

**find / -type d**

This will list all of the current directories in your system since we searched from our root directory with the / (slash) symbol.

You can also combine the –**type** and**-name** options to narrow down your searches further:

**find / -type f -name my-file**

This will look for files named **my-file**, excluding directories or links.

**Searching by Time**

If you want to search for files based on when they were accessed and modification time footprints. Linux keeps track of the files using these three timestamps.

* **Access Time** (**-atime**) – when the file was either read or written into.
* **Modification Time** (**-mtime**) – when the file was modified.
* **Change Time** (**-ctime**) – when the file’s meta-data was updated.

This option has to be used with a number that specifies how many days passed since the file was accessed, modified or changed:

**find / -atime 1**

This command will show all files that were accessed a day ago starting from your current time.

We can narrow down our queries even more by adding **plus** (**+**) and **minus** (–) signs preceding the number of days. For instance:

**find / -mtime +2**

It lists down all the files that have a modification time of more than two days ago.

To find all files whose meta-data was updated less than a day ago, run the following:

**find / -ctime -1**

While not often used, there are some additional arguments that are also related to timed-searches. The **-mmin** argument looks for modified files on a minute basis. It can be used like this:

find / -mmin -1

Also, we have the**-newer** argument, which can be used to compare the age of two or more files and display the newer one.

find / -newer my-file

What you’ll get are all of the files that are more recently modified than your file.

**Searching by Size**

Linux lets you search for files based on their sizes. The syntax for searching files by size is:

**find <startingdirectory> -size <size-magnitude> <size-unit>**

You can specify the following size units:

* **c** – bytes
* **k** – kilobytes
* **M** – megabytes
* **G** – gigabytes
* **b** – 512-byte chunks

A simple example of how to use the **find** command for file sizes is as follows:

* **find / -size 10M**

Here we search for all of the files in your system that are exactly 10 megabytes. Just like when searching based on time, you can filter your searches further using the plus and minus signs:

find / -size +5G

It will display all the files that are more than five gigabytes in size.

**Searching by Ownership**

Linux gives you the ability to narrow down your searches based on file ownership. To find files of a certain owner, the following command should be executed:

* **find / -user john**

The script will return a list of all files that the user named **john** owns. Similar to usernames, we can also find files through group names:

* **find / -group classroom**

**locate**

### The Basic Syntax

You can now use the command to search for files using this syntax:

**locate [my-file]**

#### Count the Number of Files

In order to tell how many files appear on your search result, insert **-c** after the locate command.

**locate -c my-file**

Instead of listing all the files, it will give you the total number of them.

#### Ignore Case Sensitive

Use **-i** on your linux**locate** command to ignore case sensitive files. For instance:

**locate -i my-file**

All of the files with this name will be shown, regardless of any uppercase or lowercase symbols found.