

[hostinger.es](https://www.hostinger.es)

Comandos Locate y Find de Linux: Guía Detallada

11-14 minutos

[VPS](#)

access_time

noviembre 20, 2020

hourglass_empty

7min de lectura

person_outline

Gustavo B.

En este tutorial, veremos cómo utilizar las capacidades de búsqueda de archivos de Linux. Cubriremos dos comandos Linux diferentes para buscar archivos: **find** (que traduce encontrar) y **locate** (que traduce ubicar). Se mostrarán en detalle la forma de uso y ejemplos para ambos comandos para que puedas elegir el mejor para ti. Además, estamos utilizando Ubuntu VPS, pero esto también funcionará para Debian, CentOS o cualquier otra distribución de Linux que estés usando.

Las personas que recién comienzan con Linux a menudo afirman que se confunden al tratar de buscar archivos. Esto se debe a que la mayoría migra sus archivos desde Windows, que tiene un diseño

de directorio más claro. Y aunque hay algo de cierto en esto, Linux ofrece a los usuarios muchas más opciones para buscar archivos que Windows. Una vez que te has familiarizado con estos comandos, la búsqueda de archivos en su plataforma Linux será extremadamente fácil. En resumen, podrás ejecutar comandos y buscar archivos por nombre, formato, tamaño, fecha de modificación y mucho más.

Usando el comando Find en Linux

Comencemos con el comando **find** y cómo usarlo en toda su capacidad.

La sintaxis básica

El comando más común utilizado para encontrar y filtrar archivos en Linux es a través del comando **find**. El diseño básico de este comando es el siguiente:

```
find <startingdirectory> <options> <search term>
```

Comienza con la palabra clave *find*, que alerta a Linux de que lo que sigue se refiere a la búsqueda de un archivo. El argumento **<startingdirectory>** es el punto de origen de donde deseas iniciar la búsqueda. Esto es útil si tienes una idea aproximada de dónde podría estar ubicado el archivo deseado, ya que hace más específica la búsqueda. La mayoría de las veces, sin embargo, querrás buscar el archivo en todo el sistema. Puedes hacer esto reemplazando tu ruta con una barra “/”, que es el símbolo del directorio raíz. A veces es posible que quieras iniciar la búsqueda desde el directorio de trabajo actual, es decir, el directorio donde está abierto el terminal. Esto se puede hacer con el argumento

punto “ . “. Para averiguar tu directorio actual, usa el comando **pwd**. Finalmente, para comenzar la búsqueda de archivos desde tu carpeta de inicio, usa el símbolo “ ~ “.

El segundo argumento es el filtro que deseas usar para buscar tu archivo. Este podría ser el nombre, tipo, fecha de creación o de modificación del archivo, etc. El tercer argumento es una continuación del segundo, donde se especificará el término de búsqueda relevante. Echemos un vistazo a las diversas opciones que Linux proporciona a los usuarios cuando buscan sus archivos:

Búsqueda por nombre

Por supuesto, el método más común y obvio para buscar un archivo es usar su nombre. Para ejecutar una consulta de búsqueda simple usando el nombre del archivo, usa el comando **find** de la siguiente manera:

```
find . -name "my-file"
```

En el comando anterior, usamos la opción **-name** y buscamos un archivo llamado **my-file**. Ten presente que comenzamos la búsqueda en nuestro directorio actual.

Una cosa importante para recordar cuando se utiliza el estándar es el argumento **-name**, que busca términos distinguiendo entre mayúsculas y minúsculas en Linux. Por lo tanto, si conoces el nombre del archivo, pero no estás seguro de si las mayúsculas y minúsculas, usa el comando find de esta manera:

```
find . -iname "my-file"
```

Otra forma de utilizar la opción *name* es buscar todos los archivos sin una palabra clave en el nombre. En Linux, puedes hacer esto

de dos maneras. El primer método implica el uso de la palabra clave **-not** de la siguiente manera:

```
find . -not -name "my-file"
```

También podemos usar “!”, aunque debe estar precedido por el identificador de escape para que Linux sepa que esta es parte del comando de búsqueda y no una independiente.

```
find . \! -name "my-file"
```

También puede buscar varios archivos con un formato común como .txt, lo cual podría ser útil en algunos casos:

```
find . -name "*.txt"
```

Esto listará todos los archivos de texto comenzando con la carpeta actual.

Finalmente, si deseas buscar un determinado archivo por nombre y eliminarlo, usa el argumento **-delete** después del nombre del archivo:

```
find . -name "my-file" -delete
```

Búsqueda por tipo

Para la mayoría de los usuarios, basta con saber cómo encontrar archivos por sus nombres. Sin embargo, siempre es útil conocer todas las herramientas que se ofrecen para aprovechar Linux al máximo.

Aquí es donde entra en juego el argumento **-type**. Linux ofrece a los usuarios las siguientes opciones para buscar archivos por tipo:

- **f** – archivo normal
- **d** – directorio o carpeta

- **l** – enlace simbólico
- **c** – dispositivos de caracteres
- **b** – dispositivos de bloque

Un ejemplo simple del uso del tipo de archivo para la búsqueda se puede ver a continuación:

```
find / -type d
```

Esto mostrará una lista de todos los directorios presentes en tu sistema de archivos, al haber comenzado la búsqueda desde nuestro directorio raíz con el símbolo de barra inclinada.

También puedes concatenar las opciones `-type` y `-name` para hacer tus búsquedas aún más específicas:

```
find / -type f -name "my-file"
```

Esto buscará archivos llamados **my-file**, excluyendo directorios o enlaces.

Búsqueda por fecha

Si quieres buscar archivos en función de su fecha de acceso y los registros de fecha de modificación, Linux te ofrece las herramientas para hacerlo. Hay 3 registros de tiempo de los cuales Linux realiza seguimiento en los archivos:

- **Tiempo de acceso (-atime)** – Fecha más reciente en que el archivo fue leído o escrito.
- **Tiempo de modificación (-mtime)** – Fecha más reciente en que se modificó el archivo.
- **Hora de cambio (-ctime)** – Fecha más reciente en que se actualizaron los metadatos del archivo.

Esta opción debe usarse con un número. Este número especifica el número de días desde que se accedió, modificó o cambió el archivo. La forma más sencilla de buscar archivos por tiempo es:

```
find / -atime 1
```

Esto encontrará todos los archivos a los que se accedió hace un día desde el momento actual. Esto significa que se listarán todos los archivos que fueron leídos y/o escritos desde el día anterior.

Podemos hacer que nuestras consultas sean más precisas con los signos más (+) y menos (-) precediendo al número de días. Por ejemplo:

```
find / -mtime +2
```

Esto listará todos los archivos que tienen un tiempo de modificación de más de dos días.

Para buscar todos los archivos cuyos metadatos de inodo se actualizaron hace menos de un día, ejecuta lo siguiente:

```
find / -ctime -1
```

Finalmente, hay algunos argumentos adicionales además de estos 3 que también están relacionados con las búsquedas por fecha. El argumento **-<time-descriptor>min** busca cuántos minutos han pasado. Se puede usar así:

```
find / -mmin -1
```

Esto buscará archivos que se modificaron hace menos de un minuto. Además, el argumento **-newer** se puede usar para comparar la antigüedad de dos archivos y encontrar el más reciente.

Búsqueda por tamaño

Al igual que Linux te brinda la opción de buscar archivos basados en registros de tiempo, también te permite hacer lo mismo con los tamaños. La sintaxis básica para buscar archivos por tamaño es:

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

Puedes especificar las siguientes unidades de tamaño:

- **c** – bytes
- **k** – kilobytes
- **M** – megabytes
- **G** – gigabytes
- **b** – trozos de 512 bytes

Un ejemplo simple de cómo usar el comando **find** para los tamaños de archivo es el siguiente:

```
find / -size 10M
```

Esto buscará en tu sistema archivos que tengan exactamente 10 megabytes de tamaño. Al igual que cuando buscaste en función del tiempo, puedes filtrar aún más tus búsquedas con los signos más y menos:

```
find / -size +5G
```

El comando anterior listará todos los archivos de tu disco que tengan más de 5 Gigabytes de tamaño. De manera similar, puede lograr esto con el signo menos para indicar “menor que” en tus consultas.

Búsqueda por propiedad

La jerarquía de privilegios de Linux también se puede utilizar al buscar archivos. Linux te da la capacidad de especificar tus búsquedas según la propiedad del archivo, así como los permisos otorgados a diferentes usuarios.

Para buscar archivos de un determinado propietario, se debe ejecutar el siguiente comando:

```
find / -user dummy
```

Esto devolverá una lista de todos los archivos que posee el usuario llamado **dummy**. Similar a los nombres de usuario, también podemos buscar archivos a través de nombres de grupo:

```
find / -group classroom
```

Esto buscará aquellos archivos que son propiedad del grupo llamado **classroom**.

Búsqueda por permisos

Los usuarios que desean buscar archivos basados en los permisos de los archivos pueden hacerlo usando la opción **-perm** del comando **find**. Por ejemplo:

```
find / -perm 644
```

En Linux, 644 corresponde a permisos de lectura y escritura. Lo que significa que este comando buscará todos los archivos que solo tienen permisos de lectura y escritura. Puedes jugar con esta opción un poco más, así:

```
find / -perm -644
```

Esto mostrará todos los archivos que tengan al menos el permiso 644.

Para leer más sobre los permisos y los diversos códigos correspondientes a otros permisos de Linux, visita [esta página web](#).

Otras opciones útiles

Además de todos estos métodos de búsqueda de archivos, hay otras opciones útiles que uno debería recordar.

Por ejemplo, para buscar archivos y carpetas vacíos en tu sistema, usa lo siguiente:

```
find / -empty
```

Del mismo modo, para buscar todos los ejecutables guardados en tu disco, utiliza la opción **-exec**:

```
find / -exec
```

Para buscar archivos legibles, puedes ejecutar el siguiente comando:

```
find / -read
```

Y eso es todo lo relacionado con el comando **find** en Linux. Como puedes ver, hay un montón de opciones disponibles para que los usuarios puedan adaptar sus consultas perfectamente de acuerdo a sus necesidades. Veamos ahora el otro comando que se puede usar en lugar de **find** para buscar archivos en Linux.

Usando el comando Locate en Linux

En este punto, podrías cuestionar la necesidad de tener una alternativa para el comando **find**. Después de todo, hace todo lo necesario para buscar archivos. Sin embargo, **locate** es una alternativa útil, ya que es más rápido que **find** para buscar en todo

el sistema.

Por defecto, Linux no viene con el comando **locate** preinstalado.

Para obtener el paquete, ejecuta los siguientes comandos en tu terminal:

```
sudo apt-get update
```

```
sudo apt-get install mlocate
```

Ahora que has adquirido **locate**, puedes usarlo para buscar archivos así:

```
locate my-file
```

Puedes usar el argumento **-b** para una búsqueda más específica. Solo buscará el “nombre base” del archivo, enumerando efectivamente solo los archivos que tienen el término de búsqueda en lugar de devolver los directorios que conducen a los archivos.

```
locate -b my-file
```

Otros argumentos disponibles incluyen:

- **-e** muestra entradas de archivos existentes en el momento en que se ejecuta el comando **locate**.
- **-q** inhabilita la visualización de errores encontrados en el proceso de búsqueda.
- **-c** muestra la cantidad de archivos que coinciden, en lugar de los nombres de los archivos

El comando **locate** estándar a veces puede mostrar archivos que han sido eliminados. Esto se debe a que el comando `locate` busca en la base de datos principal del sistema operativo Linux. Si esa base de datos no se actualiza, incluso los archivos eliminados pueden aparecer en los resultados de búsqueda. Puedes actualizar

manualmente la base de datos ejecutando lo siguiente:

```
sudo updatedb
```

Lectura adicional

- [Comando cat de Linux – Con ejemplos de uso](#)
- [Cómo usar el comando Unzip de Linux](#)
- [Cómo usar el comando Shutdown de Linux en Ubuntu 18.04 y CentOS 7](#)

Conclusión

En este tutorial, hemos aprendido a:

- Utilizar el comando **find** en Linux para buscar archivos por nombre, tipo, fecha, tamaño, propiedad y permisos, además de algunas otras opciones útiles.
- Instalar y usar **locate** para realizar búsquedas de archivos más rápidas en todo el sistema.

Los usuarios de Linux reciben dos herramientas extremadamente potentes para recorrer su disco y buscar archivos. Depende completamente de la necesidad individual del usuario elegir cuál método es más adecuado. Recomendamos dar un vistazo a ambos para ver cuál es el adecuado para ti.

Si tienes algún consejo o idea interesante, ¡compártela con la comunidad en los comentarios!