☰ 📖 **Documentation**   🔍

What are inodes in Linux?

What are inodes in Linux?   ✏️

This article provides an overview of the concept of *inode* and some helpful commands.

What is an inode?

Linux® must allocate an index node (inode) for every file and directory in the filesystem. Inodes do not store actual data. Instead, they store the metadata where you can find the storage blocks of each file's data.

Metadata in an inode

The following metadata exists in an inode:

- File type
- Permissions
- Owner ID
- Group ID
- Size of file
- Time last accessed
- Time last modified
- Soft/Hard Links
- Access Control List (ACLs)

Check the inode number in a specific file

There are different ways to check the inode number. The following example shows the creation of a file named **mytestfile**. The command `stat` displays the file statistics, including the unique inode number:

```
[root@Rackspace-Server /]# touch mytestfile
[root@Rackspace-Server /]# stat mytestfile
File: mytestfile
Size: 0              Blocks: 0          IO Block: 4096    regular empty file
Device: ca01h/51713d     Inode: 13          Links: 1
Access: (0644/-rw-r--r--)  Uid: (    0/    root)   Gid: (    0/    root)
Context: unconfined_u:object_r:etc_runtime_t:s0
Access: 2021-03-26 15:51:27.036124392 -0500
```

```
 Modify: 2021-03-26 15:51:27.036124392 -0500
 Change: 2021-03-26 15:51:27.036124392 -0500
 Birth: -
```

You can also check the inode number of **mytestfile** by listing the contents of the directory. You can run a combination of commands in the directory by using `ls` or `grep`, as shown in the following examples:

```
[root@Rackspace-Server /]# ls -lhi | grep mytestfile
    13 -rw-r--r--.   1 root root     0 Mar 26 15:51 mytestfile


[root@Rackspace-Server /]# ls -i mytestfile
13 mytestfile
```

Check the inode usage on filesystems

The following example checks the inodes on all the mounted filesystems, focusing on **/dev/xvda1**, which has a maximum inode allocation of 1,310,720:

```
[root@Rackspace-Server ~]# df -i
Filesystem       Inodes IUsed    IFree IUse% Mounted on
devtmpfs          99906   318    99588    1% /dev
tmpfs            103934     1   103933    1% /dev/shm
tmpfs            103934   500   103434    1% /run
tmpfs            103934    17   103917    1% /sys/fs/cgroup
/dev/xvda1      1310720 47034 1263686    4% /
```

Adding the flag `-h` to the preceding `df` command does not give you an exact number, but it provides a more readable output:

```
[root@Rackspace-Server ~]# df -ih
Filesystem       Inodes IUsed IFree IUse% Mounted on
devtmpfs           98K   318   98K    1% /dev
tmpfs             102K     1  102K    1% /dev/shm
tmpfs             102K   500  102K    1% /run
tmpfs             102K    17  102K    1% /sys/fs/cgroup
/dev/xvda1        1.3M   46K  1.3M    4% /
```

Count inodes under a certain directory

To check the number of inodes in a specific directory, run the following command:

```
find <DIRECTORY> | wc -l
```

The following example checks the file count in the **/root** directory.

```
[root@Rackspace-Server ~]# pwd
/root
[root@Rackspace-Server ~]# find . | wc -l
11
```

In this case, it shows 11 files created under **/root**.

What happens to the inode assigned when moving or copying a file?

When you copy a file, Linux assigns a different inode to the new file, as shown in the following example:

```
[root@Rackspace-Server inodes]# touch file1
[root@Rackspace-Server inodes]# ls -lhi
total 0
262446 -rw-r--r--. 1 root root 0 Mar 26 19:30 file1

[root@Rackspace-Server inodes]# cp file1 file2

[root@Rackspace-Server inodes]# ls -lhi
total 0
262446 -rw-r--r--. 1 root root 0 Mar 26 19:30 file1
262440 -rw-r--r--. 1 root root 0 Mar 26 19:31 file2
```

Things are different when moving a file. As long as the file does not change filesystems, the inode remains the same:

```
[root@Rackspace-Server inodes]# ls -lhi directory1/file1
262440 -rw-r--r--. 1 root root 0 Mar 26 19:34 directory1/file1

[root@Rackspace-Server inodes]# mv directory1/file1 directory2/

[root@Rackspace-Server inodes]# ls -lhi directory2/file1
262440 -rw-r--r--. 1 root root 0 Mar 26 19:34 directory2/file1
```

The following example moves **file1** from **/dev/xvda1** to the **/dev/xvdb1** filesystem:

```
[root@Rackspace-Server inodes]# df -hP {/,/backups}
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda1       20G  2.4G   17G  13% /
/dev/xvdb1      391M     0  391M   0% /backups

[root@Rackspace-Server inodes]# pwd
/inodes

[root@Rackspace-Server inodes]# mv test /backups
```

```
[root@Rackspace-Server inodes]# ls -lhi /backups/test
117329 -rw-r--r--. 1 root root 0 Mar 26 19:34 /backups/test
```

When the system moved the file to another filesystem, it assigned a different inode.

Best practices to keep the inode usage low

You should check your inode usage because excessive usage can lead to issues when creating newer files. Perform the following steps to keep your usage low:

1. Delete unnecessary files and directories.
2. Delete cache files.
3. Delete old email files.
4. Delete temporary files.

What could happen if you never attend to inodes?

Even though the server has free disk space, the server can run out of inodes, which can result in the following consequences when the server does not have enough inodes when creating more files:

- You might lose data.
- The applications might crash.
- The server might restart.
- Processes might not restart.
- Scheduled tasks might not run.

🕐 **Updated 11 months ago**

← **Viewing System Users in Linux**   **What does Rackspace do in the cloud?** →

Did this page help you?   👍 Yes   👎 No

Developer Network   **Blogs**

          Developer Center   **Expert Insights Tech Blog**
          API Documentation and User Guides   **Rackspace Blog**
                                          Solve: Thought Leadership