

How to access user activity on Linux

If you are using a Linux server, you can use the following commands to check user activity - when system users log in and frequency of use, which group they belong to, and how much disk space they are currently running. consumption, what command are they running, how much disk space is taking up, ...

Finger command

A handy command to track user activity is the **finger**. This command allows you to see who is logged in or just focused on a user's activity information, see when was last logged in, where to log in from, etc. In this example, we will track activity action of user name nemo.

```
$ finger nemo Login: nemo Name: Nemo Demo Directory: /home/nemo Shell: /bin/bash
```

We can see the full name, main directory, and shell of nemo. We can also see nemo's most recent login and email activity. Offices, office phones, and home phones appear only if they are specified in the **/etc/passwd** file with the full name field. For example:

```
nemo:x:1001:1001:Nemo Demo,11,540-222-2222,540-333-3333:/home/nemo:/bin/bash).
```

The output above also indicates that nemo has no "plan" at all, but this just means that this user did not create a .plan file and put some text into it; Nothing out of the ordinary.

Without an argument, the **finger** command will display a list of current logins in the format shown below. You can see when they log in, the IP address they used, the terminal they were using (for example, pts / 1)

```
$ finger Login Name Tty Idle Login Time Office Office Phone nemo Nemo Demo pts/1
```

W command

The **w** command also provides a nicely formatted list of currently active users including free time and which command they last ran. It also displays the time the system has been operating in the top line and provides a load average to know the level of system activity. In this case (0.00 until the end of minutes 1, 5 and 15), the system will largely be zero.

```
$ w 14:23:19 up 1 day, 20:24, 2 users, load average: 0.00, 0.00, 0.00 USER TTY F
```

Command id

With the **id** command , you can view the user ID number and group ID (ID of each member of the group). This information is taken from `/etc/passwd` and `/etc/group files` . There are no arguments, the `id` command reports information for your account.

```
$ id uid=1000(shs) gid=1000(shs) groups=1000(shs),4(adm),11(admin),24(cdrom),27(s
```

The auth.log command

You can get information from the file `/var/log/auth.log` with commands like **grep** . To display the most recent login activity using the **auth.log** data , issue the following command:

```
$ grep "New session" /var/log/auth.log | awk '{print $1,$2,$3,$11}' | tail -5 Jun
```

Last command

The `last` command is the best method to view recent logins of all users or an individual. Remember that the most recent activity will be displayed first because this is the information most admins are most interested in.

```
$ last | head -5 nemo pts/1 192.168.0.6 Fri Jun 19 12:58 still logged in shs pts
```

Du command

The `du` command will report how much space each user's **home directory** is using if running each of the `/home` directories like this:

```
$ sudo du -sk /home/* 289 /home/dorothy 116 /home/dory 88 /home/eel 28 /home/gin
```

By default, sizes are reported in 1024 bytes units.

Ps command and history

For users currently logged in, you can always use commands like `ps -ef | grep ^ nemo` to see which commands and processes the user is currently running. To view the commands that were previously run, you can try viewing the user's history files (e.g. `.bash_history`), but note that the user can set up his account so that certain commands do not appear. in the history file and they can also edit these files if desired.

Count the login number

If you want to see the number of times each of your users has logged in since the `/var/log/wtmp` file was last run, use a command like this:

```
$ for USER in `ls /home` > do > cnt=`last $USER | grep ^$USER | wc -l` # count 1
```

The output will look like this:

```
dorothy: 0 dory: 0 eel: 8 gino: 0 jadep: 102 nemo: 39 shark: 50 shs: 105 tadpole
```

If you want more details, you can order a more complex script, possibly adding some additional information such as login details and formatting.

```
#!/bin/bash sepline="======" for USER in `ls /home` do len=`echo $
```

The above script is limiting the data displayed in the last five logins, but can easily change that if desired. Here's how the data for a user will be formatted:

```
shs === logins: 105 shs pts/0 192.168.0.6 Fri Jun 19 12:57 still logged in shs p
```

You finished reading the article "**How to access user activity on Linux**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.