Monitor User Commands:

Following are the basic user monitor commands

- who
- last
- w
- id

who

As a Linux user, sometimes it is required to know some basic information like :

- Time of last system boot
- List of users logged-in
- Current run level etc

Though this type of information can be obtained from various files in the Linux system but there is a command line utility 'who' that does exactly the same for you. In this article, we will discuss the capabilities and features provided by the 'who' command.

The basic syntax of the who command is :

```
who [OPTION]... [ FILE | ARG1 ARG2 ]
```

Examples of 'who' command

1. Get the information on currently logged in users

This is done by simply running the 'who' command (without any options). Consider the following example:

```
$ who
iafzal tty7         2012-08-07 05:33 (:0)
iafzal pts/0        2012-08-07 06:47 (:0.0)
iafzal pts/1        2012-08-07 07:58 (:0.0)
```

2. Get the time of last system boot

The is done using the -b option. Consider the following example:

```
$ who -b
         system boot  2012-08-07 05:32
```

So we see that the above output gives the exact date and time of last system boot.

3. Get information on system login processes

This is done using the -l option. Consider the following example:

```
$ who -l
LOGIN    tty4         2012-08-07 05:32            1309 id=4
LOGIN    tty5         2012-08-07 05:32            1313 id=5
LOGIN    tty2         2012-08-07 05:32            1322 id=2
LOGIN    tty3         2012-08-07 05:32            1324 id=3
LOGIN    tty6         2012-08-07 05:32            1327 id=6
LOGIN    tty1         2012-08-07 05:32            1492 id=1
```

So we see that information related to system login processes was displayed in the output.

4. Get the hostname and user associated with stdin

This is done using the -m option. Consider the following example:

```
$ who -m
iafzal pts/1         2012-08-07 07:58 (:0.0)
```

So we see that the relevant information was produced in the output.

5. Get the current run level

This is done using the -r option. Consider the following example:

```
$ who -r
         run-level 2  2012-08-07 05:32
```

So we see that the information related to current run level (which is 2) was produced in the output.

6. Get the list of user logged in

This is done using the -u option. Consider the following example:

```
$ who -u
iafzal tty7          2012-08-07 05:33  old        1619 (:0)
iafzal pts/0         2012-08-07 06:47 00:31       2336 (:0.0)
iafzal pts/1         2012-08-07 07:58  .          2336 (:0.0)
```

So we see that a list of logged-in users was produced in the output.

7. Get number of users logged-in and their user names

This is done using the -q option. Consider the following example:

```
$ who -q
iafzal iafzal iafzal
# users=3
```

So we see that information related to number of logged-in users and their user names was produced in the output.

8. Get all the information

This is done using the -a option. Consider the following example:

```
$ who -a
           system boot   2012-08-07 05:32
           run-level 2   2012-08-07 05:32
LOGIN      tty4          2012-08-07 05:32             1309 id=4
LOGIN      tty5          2012-08-07 05:32             1313 id=5
LOGIN      tty2          2012-08-07 05:32             1322 id=2
LOGIN      tty3          2012-08-07 05:32             1324 id=3
LOGIN      tty6          2012-08-07 05:32             1327 id=6
LOGIN      tty1          2012-08-07 05:32             1492 id=1
iafzal + tty7            2012-08-07 05:33   old       1619 (:0)
iafzal + pts/0           2012-08-07 06:47   .         2336 (:0.0)
iafzal + pts/1           2012-08-07 07:58   .         2336 (:0.0)
```

So we see that all the information that 'who' can print is produced in output.

---

last command:

To find out when a **particular user last logged in** to the Linux or Unix server.

Syntax

The basic syntax is:

```
last
last [userNameHere]
last [tty]
last [options] [userNameHere]
```

If no options provided last command displays a list of all users logged in (and out). You can filter out results by supplying names of users or terminal to show only those entries matching the username/tty.

last command examples

To find out who has recently logged in and out on your server, type:

```
$ last
```

Sample outputs:

```
root     pts/1           10.1.6.120        Tue Jan 28 05:59   still logged in
root     pts/0           10.1.6.120        Tue Jan 28 04:08   still logged in
root     pts/0           10.1.6.120        Sat Jan 25 06:33 - 08:55  (02:22)
root     pts/1           10.1.6.120        Thu Jan 23 14:47 - 14:51  (00:03)
root     pts/0           10.1.6.120        Thu Jan 23 13:02 - 14:51  (01:48)
```

```
root       pts/0          10.1.6.120         Tue Jan  7 12:02 - 12:38  (00:35)

wtmp begins Tue Jan  7 12:02:54 2014
```

List all users last logged in/out time

last command searches back through the file /var/log/wtmp file and the output may go back to
several months. Just use the less command or more command as follows to display output one
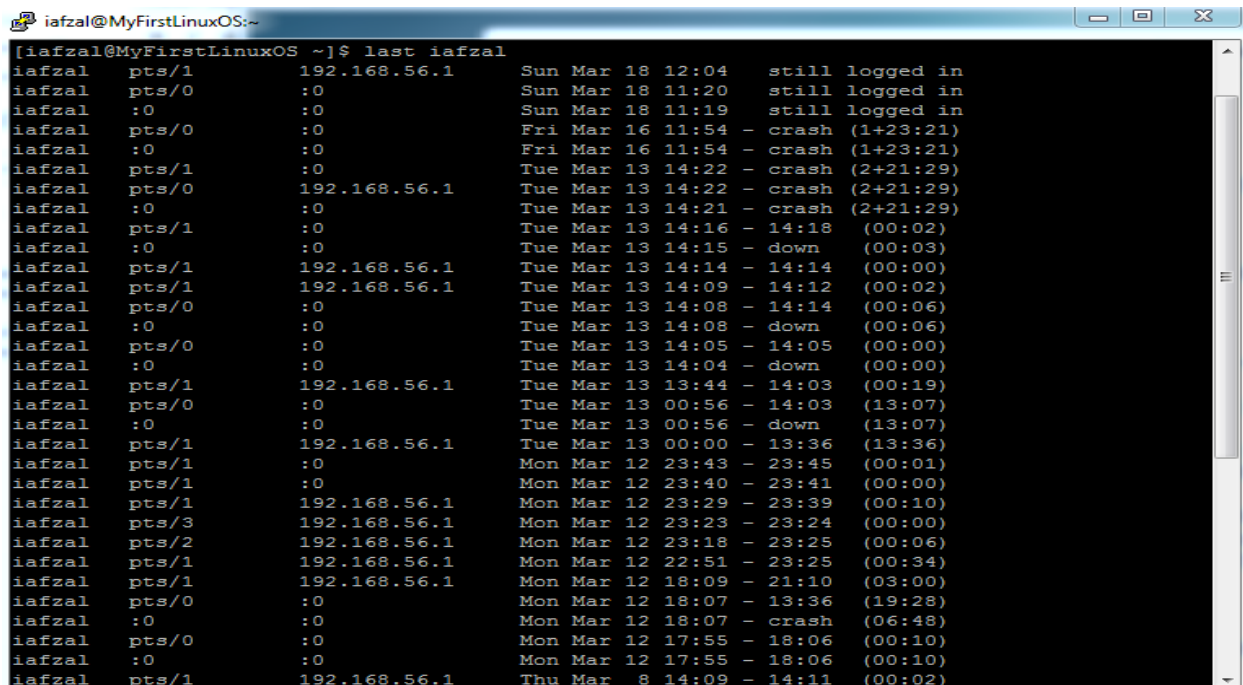screen at a time:
```
$ last | more
last | less
```

List a particular user last logged in

To find out when user iafzal last logged in, type:
```
$ last iafzal
$ last iafzal | less
$ last iafzal | grep 'Thu Jan 23'
```

Sample outputs:

Hide hostnames (Linux only)

To hide the display of the hostname field pass `-R` option:
```
$ last -R
last -R iafzal
```
Sample outputs:

```
iafzal@MyFirstLinuxOS:~

[iafzal@MyFirstLinuxOS ~]$ last -R iafzal
iafzal    pts/1        Sun Mar 18 12:04    still logged in
iafzal    pts/0        Sun Mar 18 11:20    still logged in
iafzal    :0           Sun Mar 18 11:19    still logged in
iafzal    pts/0        Fri Mar 16 11:54 - crash (1+23:21)
iafzal    :0           Fri Mar 16 11:54 - crash (1+23:21)
iafzal    pts/1        Tue Mar 13 14:22 - crash (2+21:29)
iafzal    pts/0        Tue Mar 13 14:22 - crash (2+21:29)
iafzal    :0           Tue Mar 13 14:21 - crash (2+21:29)
iafzal    pts/1        Tue Mar 13 14:16 - 14:18  (00:02)
iafzal    :0           Tue Mar 13 14:15 - down   (00:03)
iafzal    pts/1        Tue Mar 13 14:14 - 14:14  (00:00)
iafzal    pts/1        Tue Mar 13 14:09 - 14:12  (00:02)
iafzal    pts/0        Tue Mar 13 14:08 - 14:14  (00:06)
iafzal    :0           Tue Mar 13 14:08 - down   (00:06)
iafzal    pts/0        Tue Mar 13 14:05 - 14:05  (00:00)
iafzal    :0           Tue Mar 13 14:04 - down   (00:00)
iafzal    pts/1        Tue Mar 13 13:44 - 14:03  (00:19)
iafzal    pts/0        Tue Mar 13 00:56 - 14:03  (13:07)
iafzal    :0           Tue Mar 13 00:56 - down   (13:07)
iafzal    pts/1        Tue Mar 13 00:00 - 13:36  (13:36)
iafzal    pts/1        Mon Mar 12 23:43 - 23:45  (00:01)
iafzal    pts/1        Mon Mar 12 23:40 - 23:41  (00:00)
iafzal    pts/1        Mon Mar 12 23:29 - 23:39  (00:10)
iafzal    pts/3        Mon Mar 12 23:23 - 23:24  (00:00)
iafzal    pts/2        Mon Mar 12 23:18 - 23:25  (00:06)
```

Display complete login & logout times

By default year is now displayed by last command. You can force last command to display full
login and logout times and dates by passing `-F` option:
```
$ last -F
```
Sample outputs:

```
iafzal@MyFirstLinuxOS:~

[iafzal@MyFirstLinuxOS ~]$ last -F
iafzal    pts/1     192.168.56.1       Sun Mar 18 12:04:26 2018    still logged in
iafzal    pts/0     :0                 Sun Mar 18 11:20:44 2018    still logged in
iafzal    :0        :0                 Sun Mar 18 11:19:11 2018    still logged in
reboot    system boot 3.10.0-693.el7.x Sun Mar 18 11:16:17 2018 - Sun Mar 18 12:07:42 2018  (00:51)
iafzal    pts/0     :0                 Fri Mar 16 11:54:45 2018 - crash                      (1+23:21)
iafzal    :0        :0                 Fri Mar 16 11:54:32 2018 - crash                      (1+23:21)
reboot    system boot 3.10.0-693.el7.x Fri Mar 16 11:51:26 2018 - Sun Mar 18 12:07:42 2018 (2+00:16)
iafzal    pts/1     :0                 Tue Mar 13 14:22:22 2018 - crash                      (2+21:29)
iafzal    pts/0     192.168.56.1       Tue Mar 13 14:22:16 2018 - crash                      (2+21:29)
iafzal    :0        :0                 Tue Mar 13 14:21:53 2018 - crash                      (2+21:29)
reboot    system boot 3.10.0-693.el7.x Tue Mar 13 14:19:18 2018 - Sun Mar 18 12:07:42 2018 (4+21:48)
spiderma  pts/2     192.168.56.1       Tue Mar 13 14:18:11 2018 - Tue Mar 13 14:18:50 2018  (00:00)
iafzal    pts/1     :0                 Tue Mar 13 14:16:10 2018 - Tue Mar 13 14:18:50 2018  (00:02)
spiderma  pts/0     192.168.56.1       Tue Mar 13 14:16:00 2018 - Tue Mar 13 14:18:14 2018  (00:02)
iafzal    :0        :0                 Tue Mar 13 14:15:44 2018 - down                       (00:03)
reboot    system boot 3.10.0-693.el7.x Tue Mar 13 14:15:08 2018 - Tue Mar 13 14:18:55 2018  (00:03)
iafzal    pts/1     192.168.56.1       Tue Mar 13 14:14:30 2018 - Tue Mar 13 14:14:50 2018  (00:00)
```

Display full user/domain names

```
$ last -w
```

Display last reboot time

The user reboot logs in each time the system is rebooted. Thus following command will show a log of all reboots since the log file was created:
```
$ last reboot
$ last -x reboot
```

Display last shutdown time

Find out the system shutdown entries and run level changes:
```
$ last -x
$ last -x shutdown
```

Find out who was logged in at a particular time

The syntax is as follows to see the state of logins as of the specified time:
```
$ last -t YYYYMMDDHHMMSS
$ last -t YYYYMMDDHHMMSS userNameHere
```

w command:
```
Options:
 -h, --no-header      do not print header
 -u, --no-current     ignore current process username
 -s, --short          short format
 -f, --from           show remote hostname field
 -o, --old-style      old style output
 -i, --ip-addr        display IP address instead of hostname (if possible)

     --help     display this help and exit
 -V, --version  output version information and exit
```

id command:
```
Print user and group information for the specified USER,
or (when USER omitted) for the current user.
```

```
-a                 ignore, for compatibility with other versions
-Z, --context      print only the security context of the current user
-g, --group        print only the effective group ID
-G, --groups       print all group IDs
-n, --name         print a name instead of a number, for -ugG
-r, --real         print the real ID instead of the effective ID, with -ugG
-u, --user         print only the effective user ID
-z, --zero         delimit entries with NUL characters, not whitespace;
                     not permitted in default format
    --help         display this help and exit
    --version      output version information and exit
```