Tutorial Link https://course.testpad.chitkara.edu.in/tutorials/User Communication Tools and password Aging/6304a65f5611a5348833e42d

TUTORIAL

User Communication Tools and password Aging

Topics

chage for password management

chage command is used to view and change the user password expiry information. This command is used when the login is to be provided for a user for limited amount of time or when it is necessary to change the login password time to time. With the help of this command we can view the aging information of an account, date when the password was previously changed, set the password changing time, lock an account after certain amount of time etc.

In order to view the list of options that can be used with the change command use the help option#

```
# chage -h
```

```
pi@raspberrypi:~ $ chage -h
Usage: chage [options] LOGIN

Options:
  -d, --lastday LAST_DAY        set date of last password change to LAST_DAY
  -E, --expiredate EXPIRE_DATE  set account expiration date to EXPIRE_DATE
  -h, --help                    display this help message and exit
  -I, --inactive INACTIVE       set password inactive after expiration
                                to INACTIVE
  -l, --list                    show account aging information
  -m, --mindays MIN_DAYS        set minimum number of days before password
                                change to MIN_DAYS
  -M, --maxdays MAX_DAYS        set maximim number of days before password
                                change to MAX_DAYS
  -R, --root CHROOT_DIR         directory to chroot into
  -W, --warndays WARN_DAYS      set expiration warning days to WARN_DAYS

pi@raspberrypi:~ $
```

Examples:

1. -l option : *use this option to view the account aging information. In order to view the aging information of the root we have to use keyword sudo or su to become root.*

Input : sudo chage -l root

Output :

```
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Nov 29, 2017
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
pi@raspberrypi:~ $
```

2. -d option : *use this option to set the last password change date to your specified date in the command. In order to change the aging information of the root we are using the -l option to view the changed date.*

Input : sudo chage -d 2018-12-01 root

Output :

```
pi@raspberrypi:~ $ sudo chage -d 2018-12-01 root
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Dec 01, 2018
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
pi@raspberrypi:~ $
```

3. -E option : *use this option to specify the date when the account should expire. We are using the -l option to view the changed date.*

Input : sudo chage -E root

Output :

```
pi@raspberrypi:~ $ sudo chage -E 2018-12-31 root
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Dec 01, 2018
Password expires                                        : never
Password inactive                                       : never
Account expires                                         : Dec 31, 2018
Minimum number of days between password change          : 0
Maximum number of days between password change          : 99999
Number of days of warning before password expires       : 7
pi@raspberrypi:~ $
```

4. -M or -m option : *use this option to specify the maximum and minimum number of days between password change. In order to change the aging information of the root i am using the keyword sudo. Further i am using the -l option to view the changed period.*

Input : sudo chage -M 5 root

Output :

```
pi@raspberrypi:~ $ sudo chage -M 5 root
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Dec 01, 2018
Password expires                                        : Dec 06, 2018
Password inactive                                       : never
Account expires                                         : Dec 31, 2018
Minimum number of days between password change          : 0
Maximum number of days between password change          : 5
Number of days of warning before password expires       : 0
pi@raspberrypi:~ $
```

5. -I option : *use this option to specify the number of days the account should be inactive after its expiry. It is necessary that the user should change the password after it expires, this command is useful when the user does not login after its expiry. Even after this inactivity period if the password is not changed then the account is locked and the user should approach the admin to unlock it. In order to change the aging information of the root i am using the keyword sudo. Further I used the -l option to view the inactivity period.*

Input : sudo chage -I 5 root

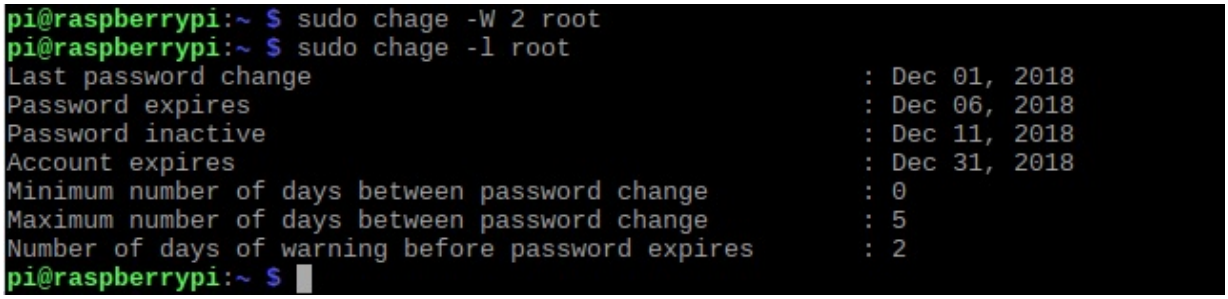
Output :

```
pi@raspberrypi:~ $ sudo chage -I 5 root
pi@raspberrypi:~ $ sudo chage -l root
Last password change                                    : Dec 01, 2018
Password expires                                        : Dec 06, 2018
Password inactive                                       : Dec 11, 2018
Account expires                                         : Dec 31, 2018
Minimum number of days between password change          : 0
Maximum number of days between password change          : 5
Number of days of warning before password expires       : 0
pi@raspberrypi:~ $
```

6. -W option : *use this option to give prior warning before the password expires. The input given in the command is the number of days prior to the expiry date when the warning should be given .In order to change the aging information of the root i am using the keyword sudo. Further i am using the -l option to view the warning period.*

Input : sudo chage -W 2 root

Output :



Talking to the Users

1 wall tool

The **wall** command (as in "write all") allows you to send a message to all users who are currently logged into the system. This implies that the system is likely a server and that users are working on the command line. While the wall command is generally used by sysadmins to send out notices to users to let send out information (e.g., that the server is going down for maintenance), it can be used by any user.

A sysadmin might send out a message like this:

```
$ wall The system will be going down in 15 minutes to address a
serious problem
```

Everyone logged into the system will see something like this:

```
Broadcast message from admin@dragonfly (pts/0) (Thu Mar  5 08:56:42
2020):
The system is going down in 15 minutes to address a serious problem
```

If you want to use single quote marks in your message, enclose the message in double quote marks like this:

```
$ wall “Don’t forget to save your work before logging off”
```

2 mesg

If, for some reason, you don’t want to accept messages from another user, you can stop them from arriving with the **mesg** command. This command can be used with a “n” argument to refuse mail from the user or a “y” argument to allow the messages to arrive.

```
$ mesg n doug
$ mesg y doug
```

The blocked user will not be notified that their messages have been blocked. You can also block or allow all messages with a **mesg** command like one of these:

```
$ mesg y
$ mesg n
```

3 write

Another command for sending text without reverting to email is **write**. This command can be used to communicate with a specific user.

```
$ write nemo
Are you still at your desk?
I need to talk with you right away.
^C
```

Enter your text and use **^C** to exit when you're done. The command allows you to send text, but doesn't start a two-way conversation. It just sends the text. If the user is logged in on more than one terminal, you can specify which terminal you want to send the message to or you can rely on the system to choose the one with the shortest idle time.

```
$ write nemo#1
```

If the user you are trying to write to has messages blocked, you should see something like this:

```
$ write nemo
write: nemo has messages disabled
```

4 users

users command in Linux system is used to show the user names of users currently logged in to the current host. It will display who is currently logged in according to *FILE*.

Example: *users* command without any option will print the users currently logged in.