If password policy is already implemented in your organization, then you don’t need to look for this options.

However, if you had set up lock period for 24 hours you might need to unlock the user’s account manually.

This tutorial will help you to manually lock and unlock users account in Linux.

This can be done using the following two Linux Commands in three ways.

* **passwd:**This is used to update user’s authentication tokens. This is achieved by calling the Linux-PAM and Libuser API
* **usermod:**This is used to modify/update given user’s account information. It used to add a user to a specific group, etc.,

To experiment this, we are taking daygeek user account. Let’s see the steps.

To check the given user account is available or not in system  use id Command. Yes, my account is available in the system.

# id daygeek

uid=2240(daygeek) gid=2243(daygeek) groups=2243(daygeek),2244(ladmin)

Method-1: How to lock, unlock and check status of the given user account in Linux using passwd Command?

The **passwd** command is one of the frequently used command by Linux administrator very often to update user’s authentication tokens in the /etc/shadow file.

Run the **passwd** command with the -l switch to lock the given user account.

# passwd -l daygeek

Locking password for user daygeek.

passwd: Success

You can check the locked account status either by using passwd command or grep the given user name from /etc/shadow file.

Checking the user account locked status using passwd command.

# passwd -S daygeek

or

# passwd --status daygeek

daygeek LK 2019-05-30 7 90 7 -1 (Password locked.)

This will show few short information about the status of the password of given account.

* **LK:** Password locked
* **NP:** No password
* **PS:** Password set

Checking the locked user account status using /etc/shadow file. Two exclamation mark will be added in front of the password, if the account is already locked.

# grep daygeek /etc/shadow

daygeek:!!$6$tGvVUhEY$PIkpI43HPaEoRrNJSRpM3H0YWOsqTqXCxtER6rak5PMaAoyQohrXNB0YoFCmAuh406n8XOvBBldvMy9trmIV00:18047:7:90:7:::

Run the **passwd** command with the -u switch to unlock the given user account.

# passwd -u daygeek

Unlocking password for user daygeek.

passwd: Success

Method-2: using usermod

Run the usermod command with the -L switch to lock the given user account.

# usermod --lock daygeek

or

# usermod -L daygeek

You can check the locked account status by using either passwd command or grep the given user name from /etc/shadow file.

Checking the user account locked status using passwd command.

# passwd -S daygeek

or

# passwd --status daygeek

daygeek LK 2019-05-30 7 90 7 -1 (Password locked.)

This will show a short information about the status of the password for a given account.

* **LK:** Password locked
* **NP:** No password
* **PS:** Password set

Checking the locked user account status using /etc/shadow file. Two exclamation mark will be added in front of the password, if the account is already locked.

# grep daygeek /etc/shadow

daygeek:!!$6$tGvVUhEY$PIkpI43HPaEoRrNJSRpM3H0YWOsqTqXCxtER6rak5PMaAoyQohrXNB0YoFCmAuh406n8XOvBBldvMy9trmIV00:18047:7:90:7:::

Run the usermod command with the -U switch to unlock the given user account.

# usermod --unlock daygeek

or

# usermod -U daygeek

Method-3: How To disable /enable SSH access to the given user account in Linux using usermod command?

This can be done by assigning the nologin shell to the given user. To do so, run the below command.

# usermod -s /sbin/nologin daygeek

You can check the locked user account details by greping the given user name from /etc/passwd file.

# grep daygeek /etc/passwd

daygeek:x:2240:2243::/home/daygeek:/sbin/nologin

We can enable the user ssh access by assigning back to the old shell.

# usermod -s /bin/bash daygeek

How to lock, unlock and check status of multiple user account in Linux using Shell script?

Create The Users list. Each user should be in separate line.

$ cat user-lists.txt

u1

u2

u3

u4

u5

Use the following shell script to lock multiple users account in Linux.

# user-lock.sh

#!/bin/bash

for user in `cat user-lists.txt`

do

passwd -l $user

done

Set an executable permission to user-lock.sh file.

# chmod + user-lock.sh

Finally run the script to achieve this.

# sh user-lock.sh

Locking password for user u1.

passwd: Success

Locking password for user u2.

passwd: Success

Locking password for user u3.

passwd: Success

Locking password for user u4.

passwd: Success

Locking password for user u5.

passwd: Success

Use the following shell script to check locked users account in Linux.

# vi user-lock-status.sh

#!/bin/bash

for user in `cat user-lists.txt`

do

passwd -S $user

done

Set an executable permission to user-lock-status.sh file.

# chmod + user-lock-status.sh

Finally run the script to achieve this.

# sh user-lock-status.sh

u1 LK 2019-06-10 0 99999 7 -1 (Password locked.)

u2 LK 2019-06-10 0 99999 7 -1 (Password locked.)

u3 LK 2019-06-10 0 99999 7 -1 (Password locked.)

u4 LK 2019-06-10 0 99999 7 -1 (Password locked.)

u5 LK 2019-06-10 0 99999 7 -1 (Password locked.)

Use the following shell script to unlock multiple users account in Linux.

# user-unlock.sh

#!/bin/bash

for user in `cat user-lists.txt`

do

passwd -u $user

done

Set an executable permission to user-unlock.sh file.

# chmod + user-unlock.sh

Finally run the script to achieve this.

# sh user-unlock.sh

Unlocking password for user u1.

passwd: Success

Unlocking password for user u2.

passwd: Success

Unlocking password for user u3.

passwd: Success

Unlocking password for user u4.

passwd: Success

Unlocking password for user u5.

passwd: Success

Run the same shell script user-lock-status.sh to check whether the locked user accounts got unlocked in Linux.

# sh user-lock-status.sh

u1 PS 2019-06-10 0 99999 7 -1 (Password set, SHA512 crypt.)

u2 PS 2019-06-10 0 99999 7 -1 (Password set, SHA512 crypt.)

u3 PS 2019-06-10 0 99999 7 -1 (Password set, SHA512 crypt.)

u4 PS 2019-06-10 0 99999 7 -1 (Password set, SHA512 crypt.)

u5 PS 2019-06-10 0 99999 7 -1 (Password set, SHA512 crypt.)