**Chapter 6.11: Managing Local Users and Groups**

Marko Shaffer

Information Technology, Franklin University

ITEC 200: Linux Fundamentals

Professor Kagan Ulucay

6/11/2023

Red Hat System Administration I 8.2

Lab 04 CH 6 – Section 6.11Managing Local Users and Groups

**Performance Checklist:**

In this lab you will set a default local password policy, create a supplementary group for three users, allow that group to use sudo to run commands as root, and modify the password policy for one use

**Outcomes:**

You should be able to:

* Set a default password aging policy of the local user's password.
* Create a group and use the group as a supplementary group for new users.
* Create three new users with the new group as their supplementary group.
* Configure the group members of the supplementary group to run any command as any user using **sudo**.
* Set a user-specific password aging policy.

Log in to workstation as student using student as the password.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Franklin VM: | Standard User Account: | The Student's Root Account: |
| Username | kiosk | student | root |
| Password | redhat | student | redhat |

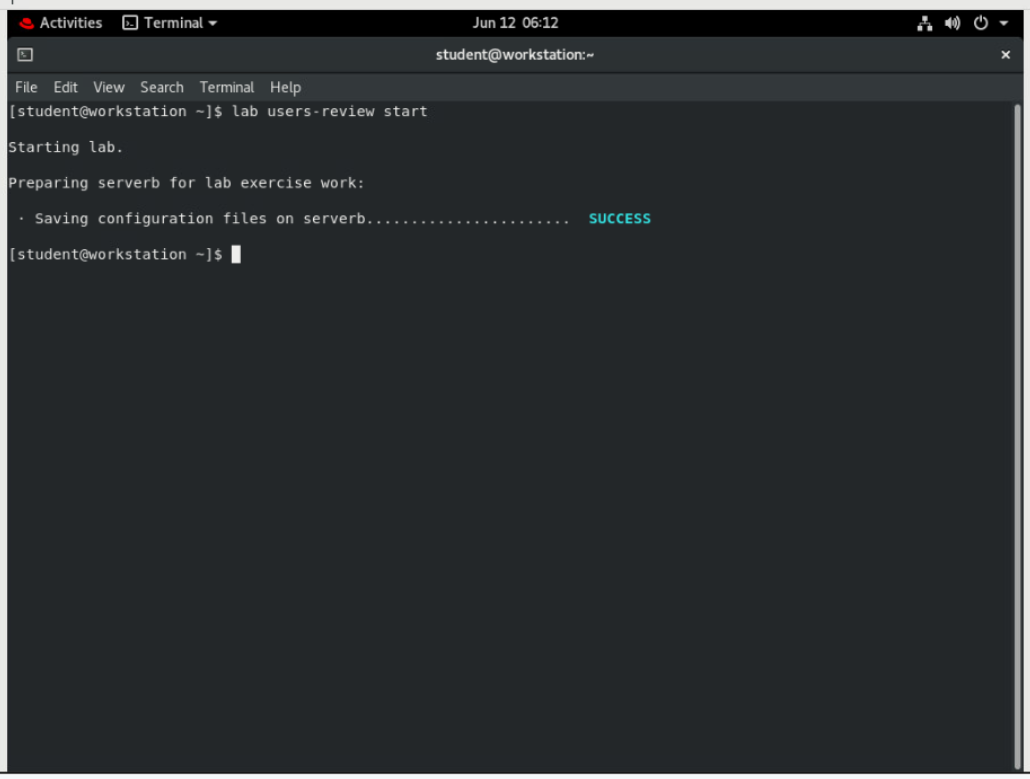
<https://franklin.instructure.com/courses/12488/modules/items/683350>

[kiosk@foundation0 ~]$ rht-vmctl start all

[kiosk@foundation0 ~]$ rht-vmview view workstation

On workstation, run the **lab edit-review start** command.

**[student@workstation ~]$ lab users-review start**

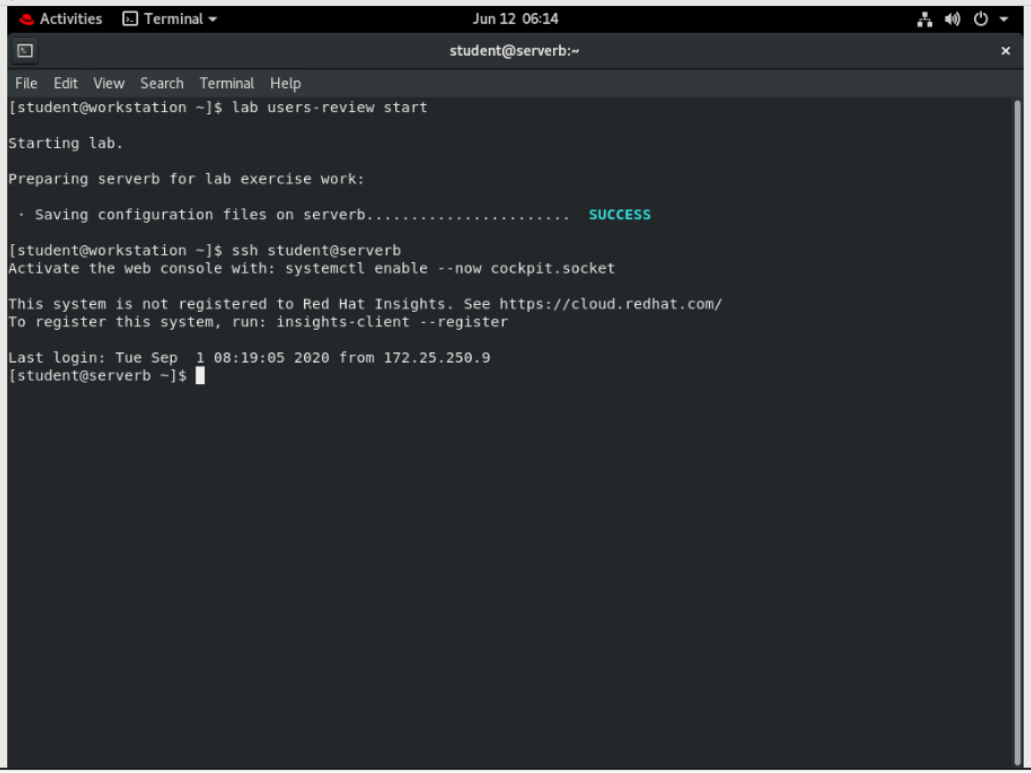
****

1. From workstation, open an SSH session to serverb as student.

**[student@workstation ~]$ ssh student@serverb**

*...output omitted...*

**[student@serverb ~]$**



1. On serverb, ensure that newly created users have passwords that must be changed every 30 days.
   1. Set PASS\_MAX\_DAYS to 30 in /etc/login.defs. Use administrative rights while opening the file with the text editor. You can use the **sudo vim /etc/login.defs** command to perform this step. Use student as the password when **sudo** prompts you to enter the student user's password.

*...output omitted...*

# Password aging controls:

#

# PASS\_MAX\_DAYS Maximum number of days a password may be

# used.

# PASS\_MIN\_DAYS Minimum number of days allowed between

# password changes.

# PASS\_MIN\_LEN Minimum acceptable password length.

# PASS\_WARN\_AGE Number of days warning given before a

# password expires.

#

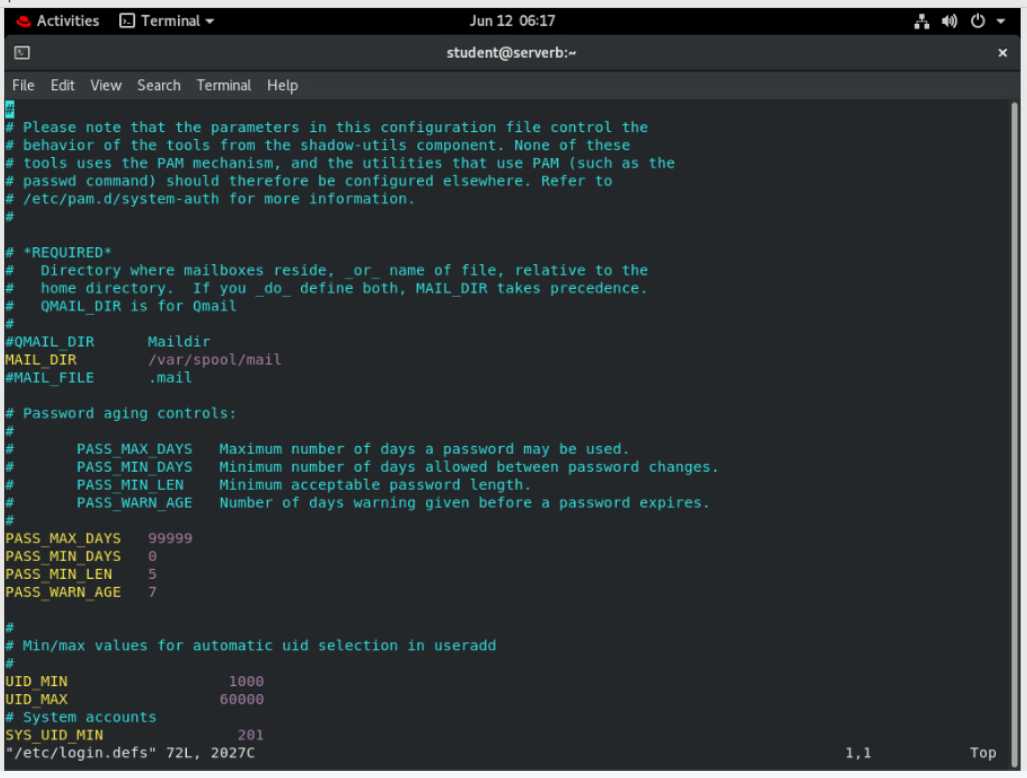
**PASS\_MAX\_DAYS 30**

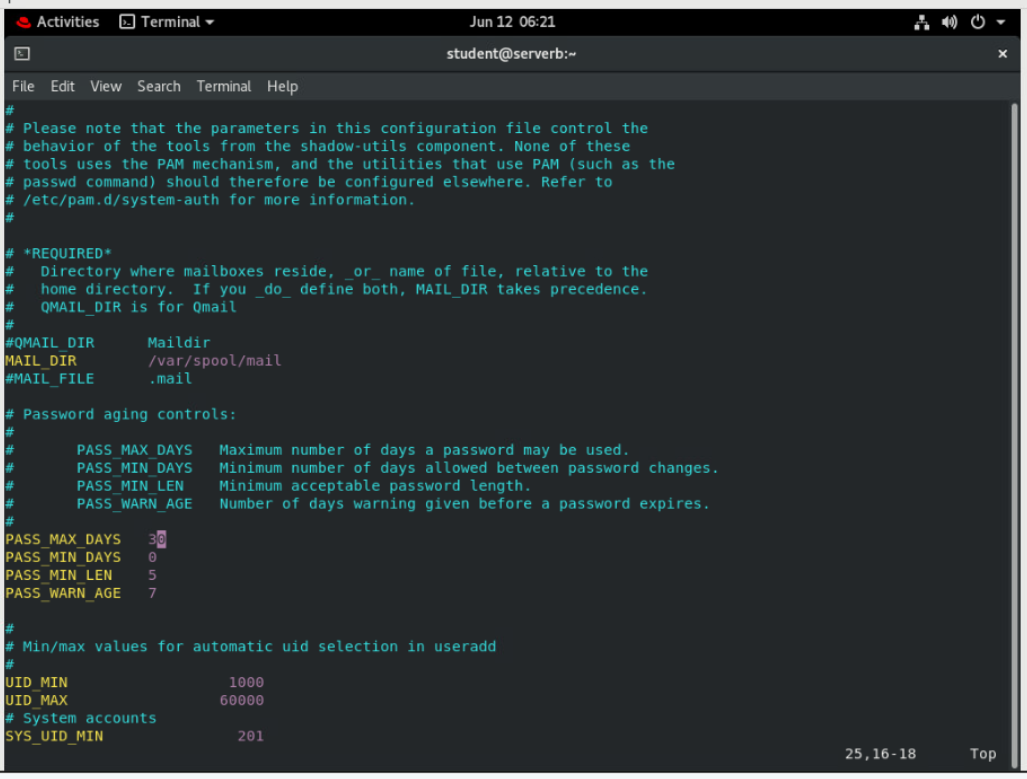
PASS\_MIN\_DAYS 0

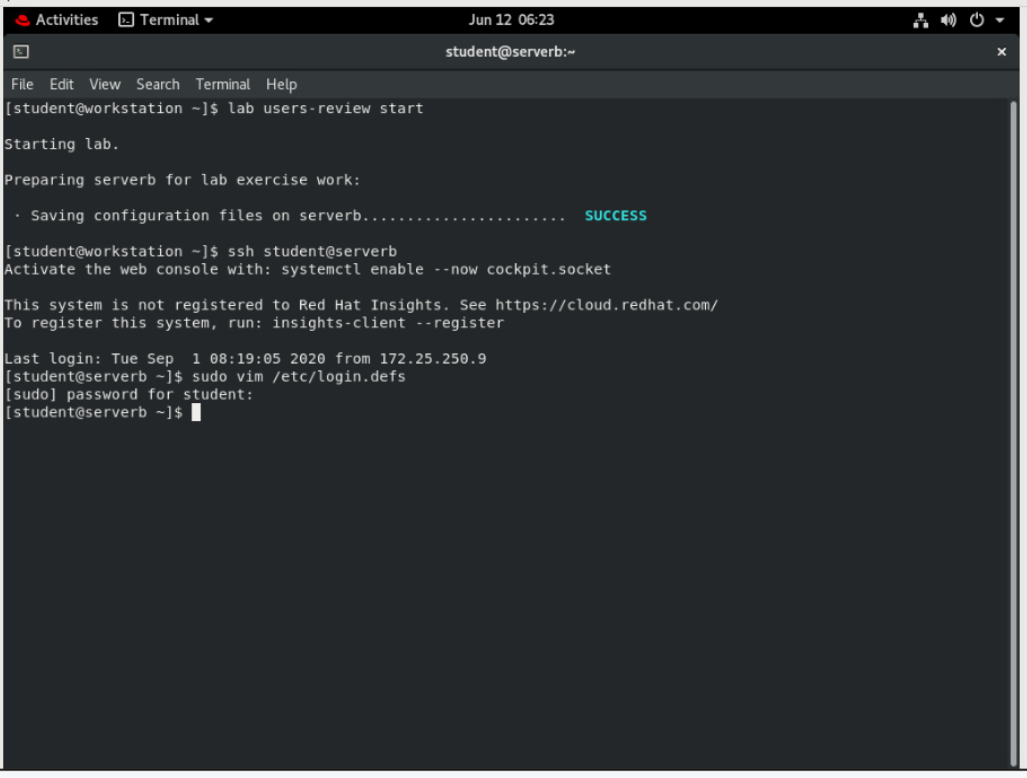
PASS\_MIN\_LEN 5

PASS\_WARN\_AGE 7

*...output omitted...*

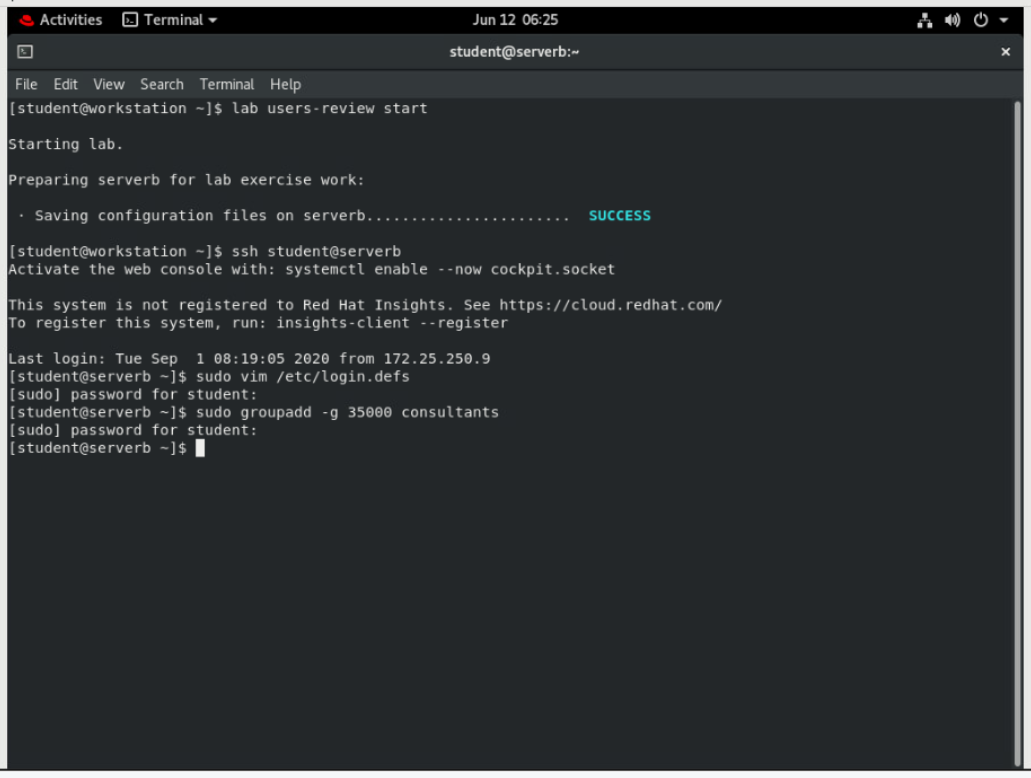






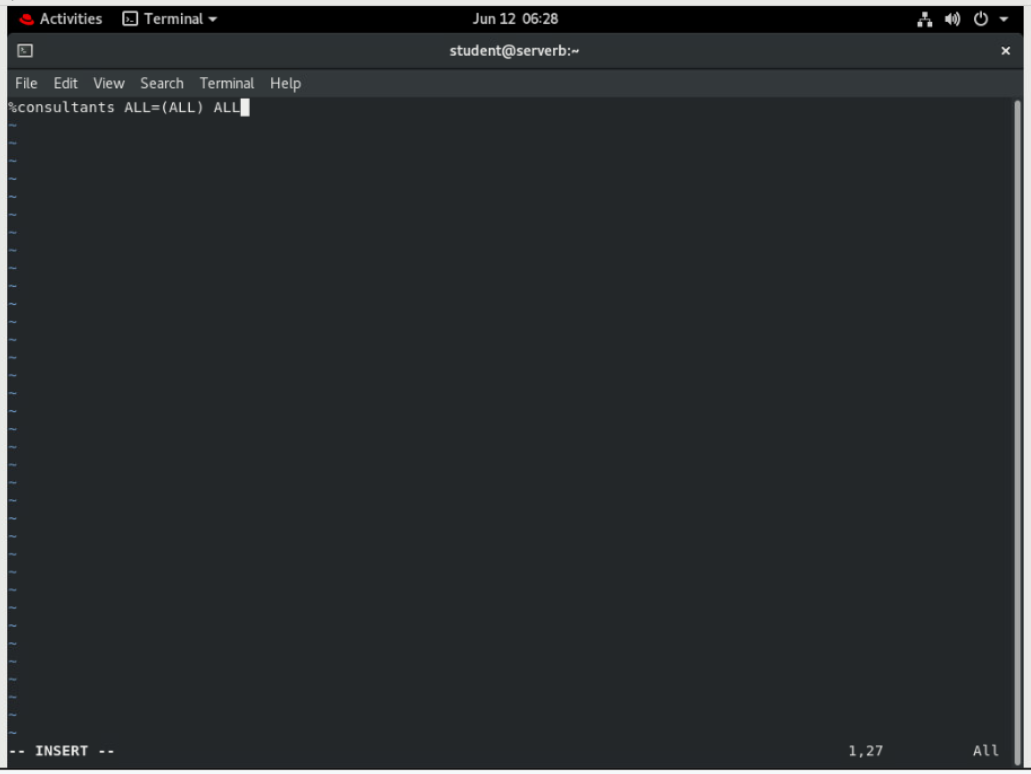
1. Create the new group called consultants with a GID of 35000.

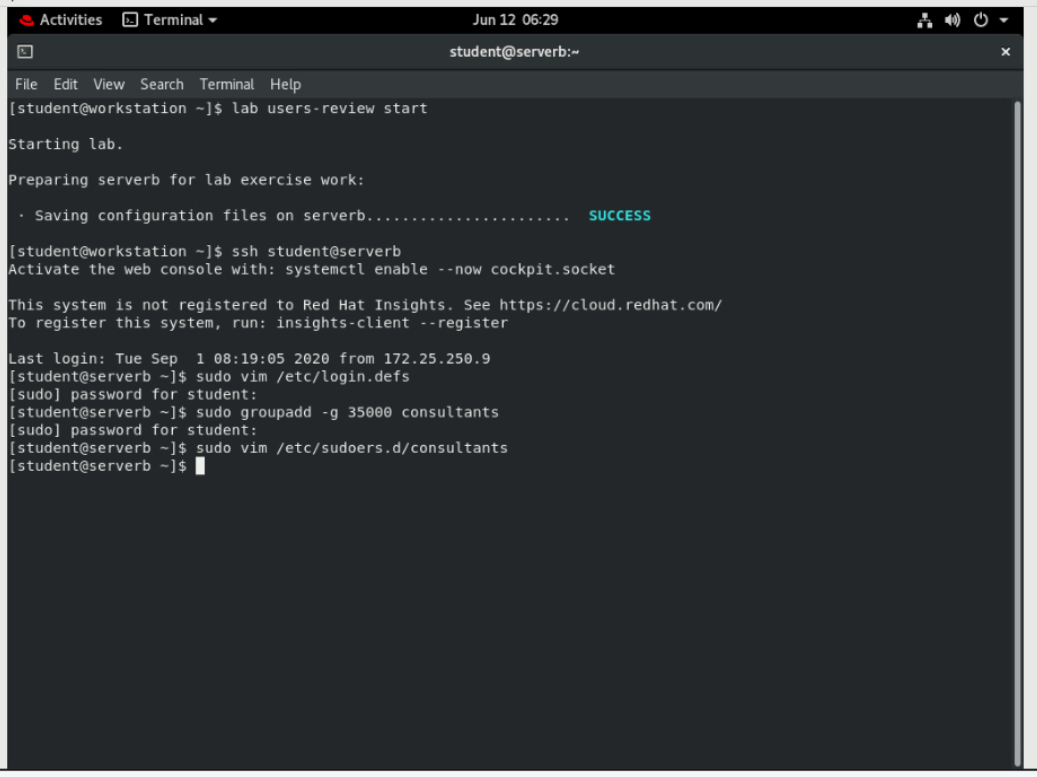
**[student@serverb ~]$ sudo groupadd -g 35000 consultants**



1. Configure administrative rights for all members of consultants to be able to execute any command as any user.
   1. Create the new file /etc/sudoers.d/consultants and add the following content to it. You can use the **sudo vim /etc/sudoers.d/consultants** command to perform this step.

%consultants ALL=(ALL) ALL



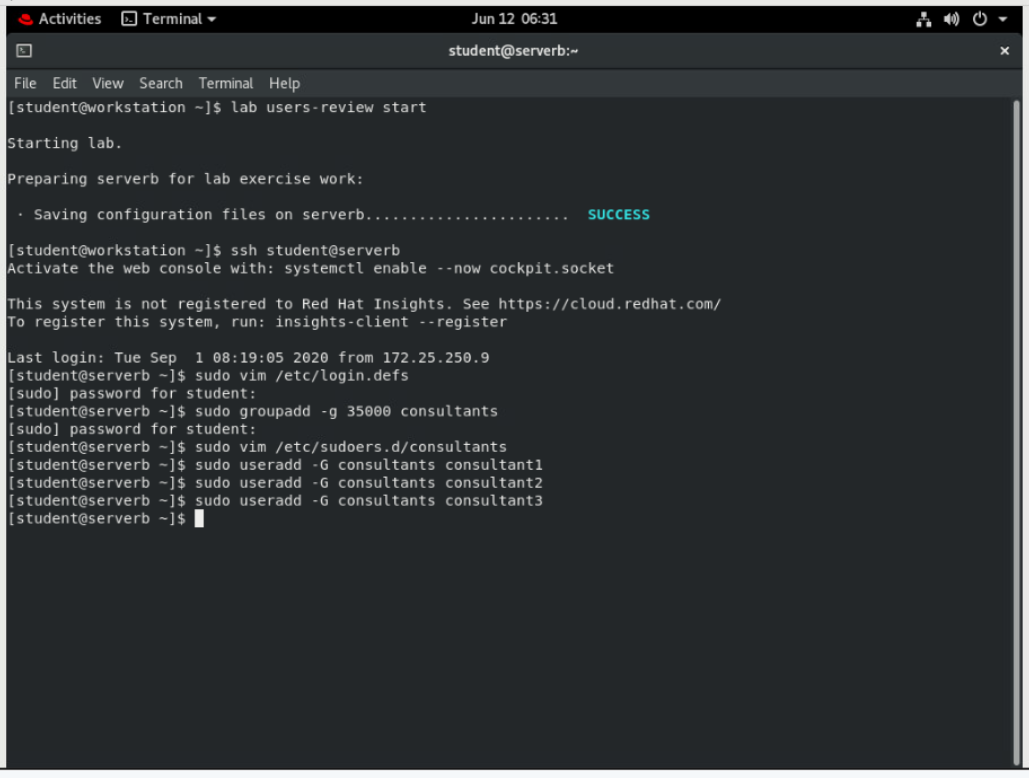


1. Create the consultant1, consultant2, and consultant3 users with consultants as their supplementary group.

**[student@serverb ~]$ sudo useradd -G consultants consultant1**

**[student@serverb ~]$ sudo useradd -G consultants consultant2**

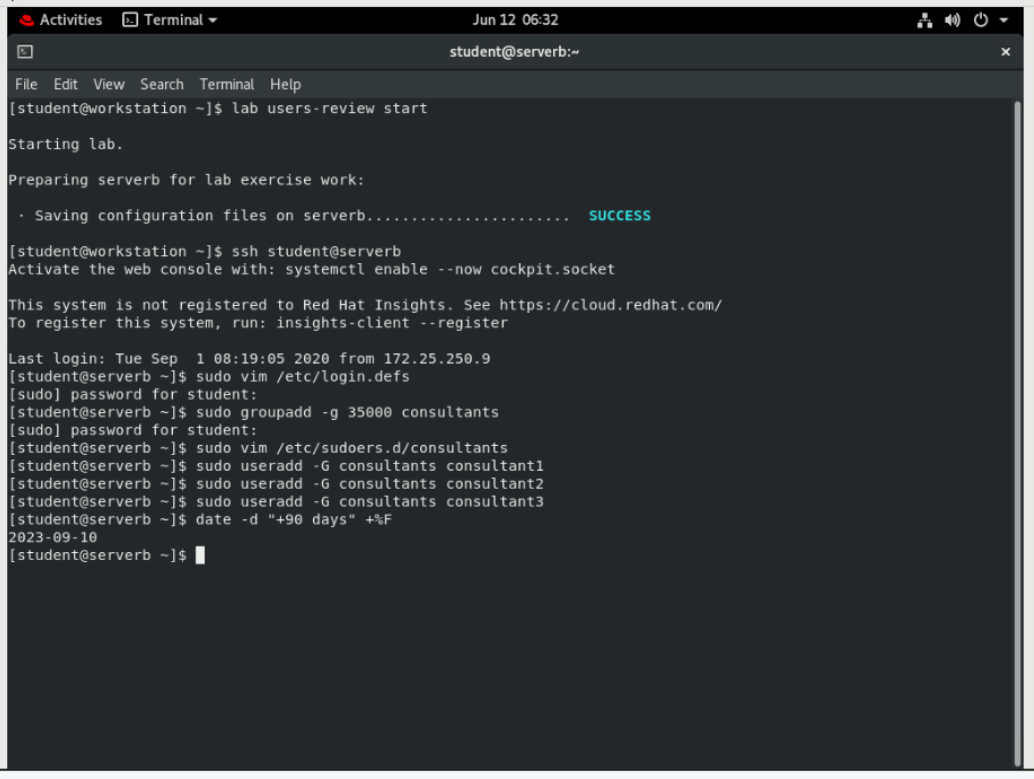
**[student@serverb ~]$ sudo useradd -G consultants consultant3**



1. Set the consultant1, consultant2, and consultant3 accounts to expire in 90 days from the current day.
   1. Determine the date 90 days in the future. You may get a different value as compared to the following output based on the current date and time of your system.

**[student@serverb ~]$ date -d "+90 days" +%F**

2019-04-28

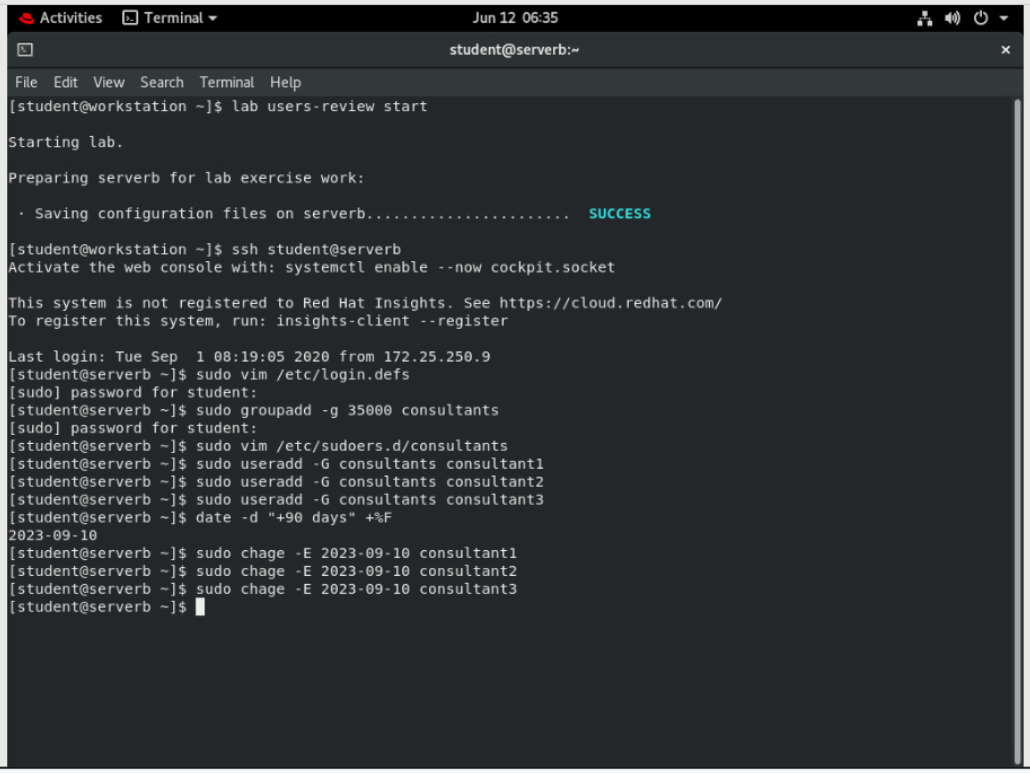


* 1. Set the account expiry date of the consultant1, consultant2, and consultant3 accounts to the same value as determined in the preceding step.

**[student@serverb ~]$ sudo chage -E *2019-04-28* consultant1**

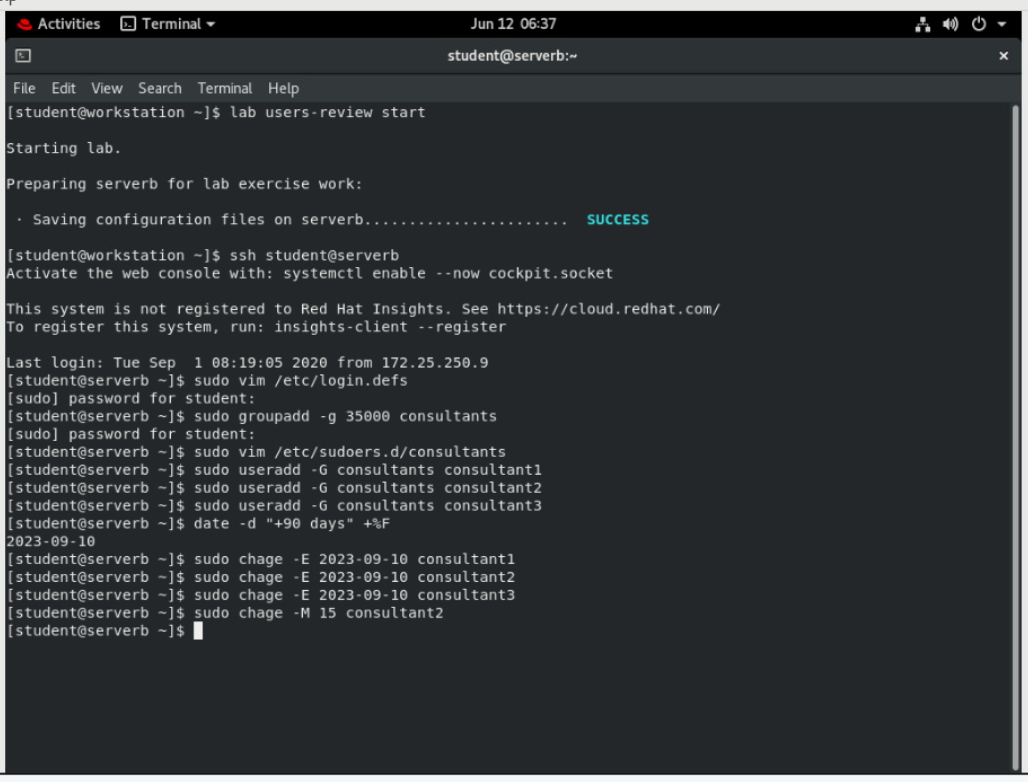
**[student@serverb ~]$ sudo chage -E *2019-04-28* consultant2**

**[student@serverb ~]$ sudo chage -E *2019-04-28* consultant3**



1. Change the password policy for the consultant2 account to require a new password every 15 days.

**[student@serverb ~]$ sudo chage -M 15 consultant2**

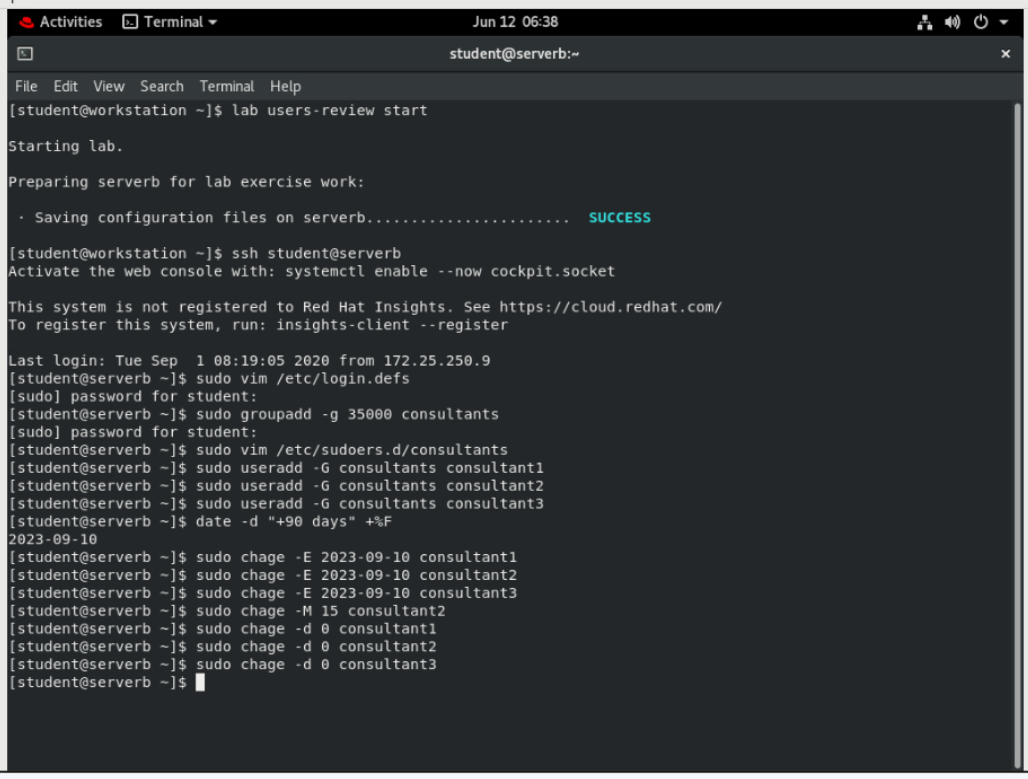


1. Additionally, force the consultant1, consultant2, and consultant3 users to change their passwords on the first login.
   1. Set the last day of the password change to 0 so that the users are forced to change the password whenever they log in to the system for the first time.

**[student@serverb ~]$ sudo chage -d 0 consultant1**

**[student@serverb ~]$ sudo chage -d 0 consultant2**

**[student@serverb ~]$ sudo chage -d 0 consultant3**

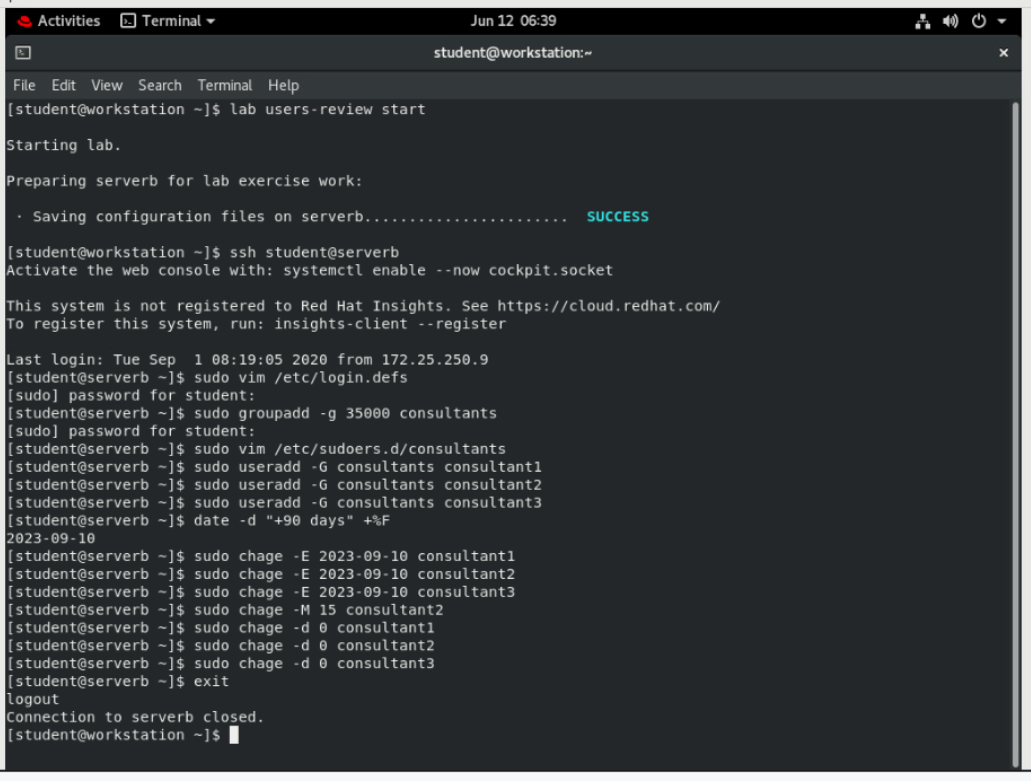


* 1. Log off from serverb.

**[student@serverb ~]$ exit**

logout

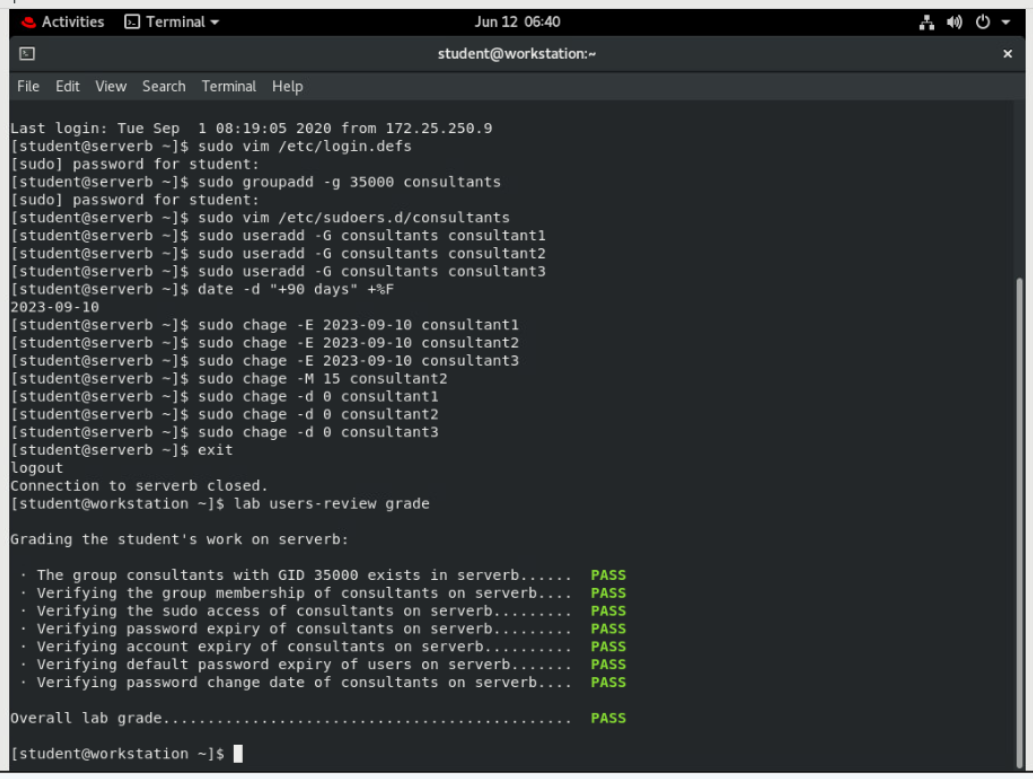
Connection to serverb closed.

****

**Evaluation**

On workstation, run the **lab users-review grade** command to confirm success of this exercise.

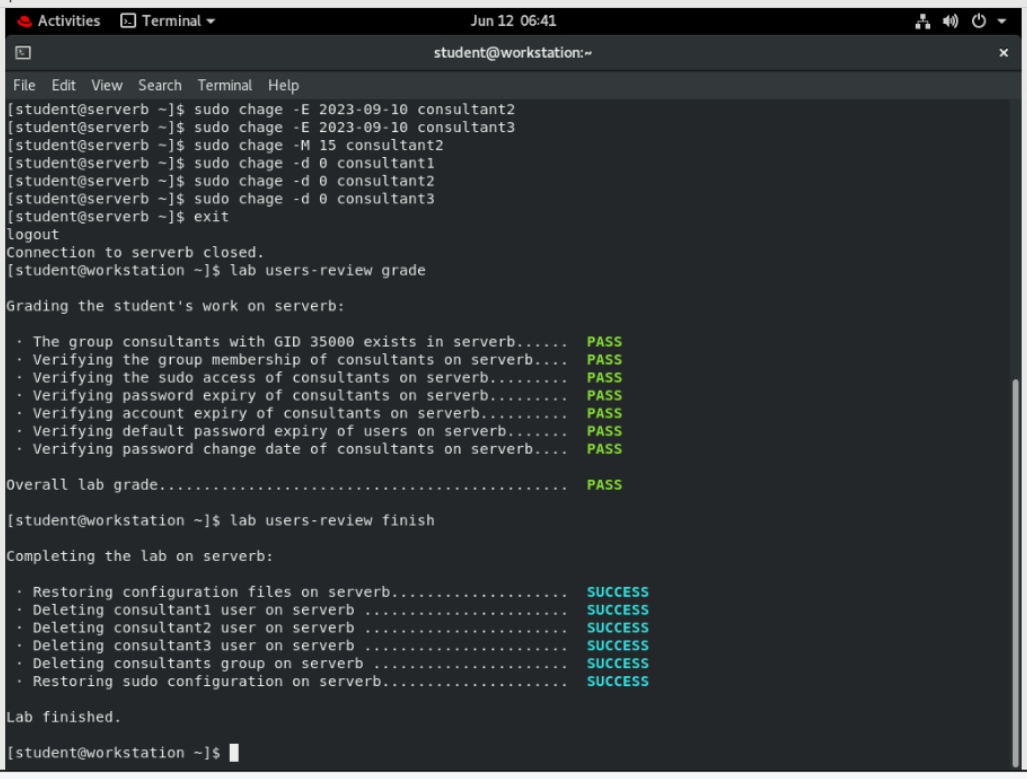
**[student@workstation ~]$ lab users-review grade**

****

**Finish**

On workstation, run **lab users-review finish** to complete this lab. This script deletes the user accounts and files created throughout the lab to ensure that the environment is clean.

**[student@workstation ~]$ lab users-review finish**



This concludes the lab.