**Chapter 6 GUIDED EXERCISE**

**MANAGING LOCAL GROUP ACCOUNTS**

In this exercise, you will create groups, use them as supplementary groups for some users without changing those users' primary groups, and configure one of the groups with sudo access to run commands as root.

**OUTCOMES**

You should be able to:

• Create groups and use them as supplementary groups.

• Configure sudo access for a group.

**BEFORE YOU BEGIN**

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

On workstation, run **lab users-group-manage start** to start the exercise. This script creates the necessary user accounts to set up the environment correctly.

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

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

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

*...output omitted...*

[student@servera ~]$

**2.** On servera, switch to root using **sudo**, inheriting the full environment of the root user.

[student@servera ~]$ **sudo su -**

[sudo] password for student: **student**

[root@servera ~]#

**3.** Create the operators supplementary group with the GID of 30000.

[root@servera ~]# **groupadd -g 30000 operators**

**4.** Create admin as an additional supplementary group.

[root@servera ~]# **groupadd admin**

**5.** Verify that both the operators and admin supplementary groups exist.

[root@servera ~]# **tail /etc/group**

*...output omitted...*

operators:x:30000:

admin:x:30001:

**6.** Ensure that the users operator1, operator2 and operator3 belong to the group

operators.

6.1. Add operator1, operator2, and operator3 to operators.

[root@servera ~]# **usermod -aG operators operator1**

[root@servera ~]# **usermod -aG operators operator2**

[root@servera ~]# **usermod -aG operators operator3**

6.2. Confirm that the users are successfully added to the group.

[root@servera ~]# **id operator1**

uid=1001(operator1) gid=1001(operator1)

groups=1001(operator1),30000(operators)

[root@servera ~]# **id operator2**

uid=1002(operator2) gid=1002(operator2)

groups=1002(operator2),30000(operators)

[root@servera ~]# **id operator3**

uid=1003(operator3) gid=1003(operator3)

groups=1003(operator3),30000(operators)

**7.** Ensure that the users sysadmin1, sysadmin2 and sysadmin3 belong to the group

admin. Enable administrative rights for all the group members of admin. Verify that any

member of admin can run administrative commands.

7.1. Add sysadmin1, sysadmin2, and sysadmin3 to admin.

[root@servera ~]# **usermod -aG admin sysadmin1**

[root@servera ~]# **usermod -aG admin sysadmin2**

[root@servera ~]# **usermod -aG admin sysadmin3**

7.2. Confirm that the users are successfully added to the group.

[root@servera ~]# **id sysadmin1**

uid=1004(sysadmin1) gid=1004(sysadmin1)

groups=1004(sysadmin1),30001(admin)

[root@servera ~]# **id sysadmin2**

uid=1005(sysadmin2) gid=1005(sysadmin2)

groups=1005(sysadmin2),30001(admin)

[root@servera ~]# **id sysadmin3**

uid=1006(sysadmin3) gid=1006(sysadmin3)

groups=1006(sysadmin3),30001(admin)

7.3. Examine **/etc/group** to verify the supplemental group memberships.

[root@servera ~]# **tail /etc/group**

*...output omitted...*

operators:x:30000:operator1,operator2,operator3

admin:x:30001:sysadmin1,sysadmin2,sysadmin3

7.4. Create the **/etc/sudoers.d/admin** file such that the members of admin have full administrative privileges.

[root@servera ~]# **echo "%admin ALL=(ALL) ALL" >> /etc/sudoers.d/admin**

7.5. Switch to sysadmin1 (a member of admin) and verify that you can run a **sudo** command as sysadmin1.

[root@servera ~]# **su - sysadmin1**

[sysadmin1@servera ~]$ **sudo cat /etc/sudoers.d/admin**

[sudo] password for sysadmin1: **redhat**

%admin ALL=(ALL) ALL

7.6. Exit the sysadmin1 user's shell to return to the root user's shell.

[sysadmin1@servera ~]$ **exit**

logout

[root@servera ~]#

7.7. Exit the root user's shell to return to the student user's shell.

[root@servera ~]# **exit**

logout

[student@servera ~]$

7.8. Log off from servera.

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

logout

Connection to servera closed.

[student@workstation ~]$

**8. Evaluation**

On workstation, run the **lab users-group-manage grade** script to complete this exercise.

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

**9. Finish**

On workstation, run **lab users-group-manage finish** to complete this exercise. This

script deletes the user accounts created at the start of the exercise.

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

This concludes the guided exercise.