Operating System fundamentals

Local users and groups

Karel de Grote Hogeschool

Inhoud

1. users and groups
2. super user access
3. managing local users
4. managing local groups
5. managing passwords

Karel de Grote Hogeschool

Course text

- Chapter 10 Manage local users and groups
  - (RedHat chapter 6)
  - Describe User and Group Concepts
  - Gain Superuser Access
  - Manage Local User Accounts
  - Manage Local Group Accounts
  - Manage User Passwords

1

Users and groups

Users and groups

- every user has a unique number (user-id)
- users can be member of one or more groups
- there is at least one group of which the user is a member (the primary group)
- groups also have a unique number (group-id)

Users

- different types
  - root
    - userid = 0
    - homedir = /root
  - "normal" users
    - userid >= 1000 (depends on the system)
    - shell
    - homedir = /home/username
  - service accounts
    - userid < 1000
    - no shell in general
    - examples: mail, lp, syslog, backup, …

/etc/passwd

- username:password:userid:groupid:comment:homedir:shell

```
root:x:0:0:root:/root:/bin/bash
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
vbox:x:999:999:vbox,,,:/home/vbox:/bin/bash
donald:x:1005:1005:D.Trump,Puma Park,35,,Perth:/home/donald:/bin/bash
```

- password is now always x and is encrypted in /etc/shadow
  - ! or * means there is no password
  - SHA512 hashing (scramble data, one-way encryption)
  - https://nordpass.com/most-common-passwords-list

The “id” command

You can find out your user-id and the groups to which you belong with the “**id**” command:

```
$ id
UID=1000(student) GID=1000(student)
groups=1000(student),4(adm),20(dialout),24(cdrom),27(sudo),30(dip)
,46(plugdev),120(lpadmin),132(lxd),133(sambashare),994(ollama)
```

Groups

- users are members of groups
  - every user has a “primary group” (in /etc/passwd)
  - for normal users a separate group is created with the same name
- users are also added to other groups
  - wheel: if the user can execute sudo (see later)
  - cdrom: if the user can use the CD or DVD drive
  - lpadmin: if the user can administer the printer
  - ...

/etc/group

- groupname:x:groupId:userlist

```
adm:x:4:syslog,donald
sambashare:x:124:donald
vbox:x:999:donald
donald:x:1005:
```

- the primary group is indicated in /etc/passwd
- there used to be passwords for groups (now always x)

The “groups” command

You can find out the groups you belong to with the “**groups**” command:

```
$ groups
student adm dialout cdrom sudo dip plugdev lpadmin lxd sambashare
ollama
```

Exercise

- look at the information in /etc/passwd, /etc/group en /etc/shadow
- do this on the KdG server and on your own virtual machine
- to which groups does your login belong?

2

Super user access

Karel de Grote Hogeschool

The “su” command

You can change your identity with the “**su**” command:

- **su** kris -> change your identity to user “kris”
- su -> change your identity to user “root”
- **su –l** -> start a new shell with the environment of “root”
- su -l kris -> start a new shell with the environment of “kris”

You’ll need to enter the password of the new user

With the (3 identical) options **-**, **-l** or **--login** a new shell is started and thus also a new environment

Exercise (on virtual machine)

```
whoami
export var=Hello
echo ${var}
```

```
su
whoami
echo ${var}
Exit
```

With this command, the original enviroment is kept

```
su -
whoami
echo ${var}
exit
```

With this command - -l or --login, you get a new login environment

The “sudo” command

- It is not wise to allow root to log into the system
- Solution: disable root login
  - How is this accomplished?
  - The su command does not work anymore
- Give permission to certain users to temporarily execute commands as root
- This is done with the “sudo” command
- Everything is logged in /var/log/secure
  - execute “**tail -f** /var/log/secure” in a terminal and try to do sudo in another terminal
- It is also possible to start a shell using “**sudo -i**”

Karel de Grote Hogeschool

Sudo access

How to give sudo access to a user?

- /etc/sudoers
  - global configuration
  - in redhat: all users of group “wheel” can use sudo
  - %wheel ALL=(ALL)   ALL
- /etc/sudoers.d
  - directory in which you can add configuration files
  - every separate file can give access to certain users or groups

Configure sudo with sudoers file

Configuration file: `/etc/sudoers`

```
## Allows people in group wheel to run all commands
%wheel          ALL=(ALL)  ALL
```

- ➔ % wheel is a group name, not a user

- ➔ ALL=(ALL:ALL)   ALL

  ALL = run the command on any host with this file

  ALL = run the command as any other user

  ALL = run the command as any other group on this system

  ALL = any command

Examples

- %wheel ALL=(ALL) ALL
  - everyone in group “wheel” can use sudo
  - from any machine
  - can assume any identity
  - can execute any command
- %games ALL=(operator) /bin/id
  - everyone in group “games” can use sudo
  - from any machine
  - can only assume the identity of “operator”
  - can only execute the command /bin/id
- ansible ALL=(ALL)NOPASSWD: ALL
  - the user “ansible” can use sudo
  - from any machine
  - can assume any identity
  - doen’t need to enter a password
  - can execute any command

Manage local users

Karel de Grote Hogeschool

Adding users

- You can add a new user with the “**useradd**” command
- default values can be found in /etc/login.defs
- examples:
  - **useradd** trinity
  - useradd -d /home/neo -c "The One" -m -s /bin/bash neo
- No option to specify a password (why not?)
- a new group is created with the same name (primary group)

- you can set the password with “**passwd**”

Exercise

- create a user “neo” with default options
- create user “trinity” with useradd and set her home directory to /Users/trinity
  (option -d)
- show the last two lines of /etc/passwd, /etc/shadow en /etc/group
- set the password for trinity
- show the last two lines of /etc/shadow

Modify a user

Use “**usermod**” to modify a user

Options (use --help for a complete overview):

- `-aG` add user to secondary groups
- `-g` modify the users' primary group
- `-L` lock user account
- `-U` unlock user account
- `-c` add data to the comment field
- …

Remove a user

- User “**userdel**” to delete a user
- The home directory will stay intact
- Use the **-r** option to also delete the home directory
- The owner of all files previously owned by the user will be changed by the user-id

Manage local groups

4

Karel de Grote Hogeschool

Adding groups

- You can add a group with “**groupadd**”
  - You can specify the group-id with **-g**
  - You can create a “system group” with **-r**

- Remark: adding a user to a group is done with:
  usermod -aG group username

Modifying groups

You can modify a group with “**groupmod**”

options:

- **-n** change the name of the group
- **-g** change the group-id

Removing groups

You can delete a group with “**groupdel**”

Primary and secondary groups

Every user has a primary group

- you can find it in /etc/passwd
- is used when a new file is created

A user can also belong to other (secondary) groups

- you can find them in /etc/group
- enables access to certain files, directories, and programs

On can temporarily change the primary group with the "**newgrp**" command

Exercise

- Create a new group “matrix”
- Change the primary group of neo and trinity to “matrix”
- Verify this with “id”
- Log in as neo and create a file, verify the primary group with ls -l
- Add neo to the “wheel” group
- Verify with “id”
- Log in again as neo and try to use sudo
- *Advanced: give trinity access to sudo, but make sure she can only use it to assume neo’s identity. Test it.*

Manage passwords

5

The shadow file

- All passwords are in /etc/shadow
- These are all encrypted (with SHA256)
- /etc/shadow also contains other information about passwords
  - when does it expire?
  - when does the user get a notification to change it?
  - …

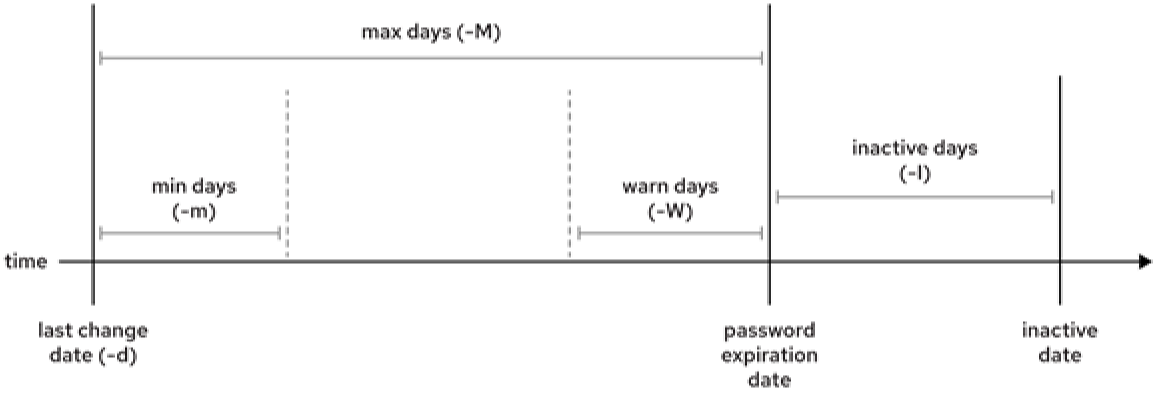
The shadow file

Fields in /etc/shadow:

- the user name
- the encrypted password
- The days from when the password was last changed,since 1970-01-01
- The minimum days since the last password change before the user can change it again.
- The maximum days without a password change before the password expires. An empty field means that the password never expires.
- The number of days ahead to warn the user that their password will expire.
- The number of days without activity, starting with the day the password expired, before the account is automatically locked.
- The day when the account expires in days since the epoch. An empty field means that the account never expires.
- The last field is typically empty and reserved for future use.

Change password properties

Use the “**chage**” command to change the properties of a password



```
[root@host ~]# chage -m 0 -M 90 -W 7 -I 14 sysadmin05
[root@host ~]# chage -d 0 USER # force password change
                                # at next login
```

Default settings

Default settings for account creation are set (by the administrator) in

```
/etc/login.defs

PASS_WARN_AGE 7
UID_MIN     1000


#FAIL_DELAY      3
```

# means commented out.

3 is the default value which the admin can change

Service accounts

There are specific accounts that are only meant to run a service

It should not be possible to login with those accounts, but there should be a password (to use the service)

-> set shell to “/sbin/nologin”

Exercises

6

Karel de Grote Hogeschool

Exercises

- KdG server
  - Ex1001-ex1099
- VM Virtualbox
  - 10.1 till 10.12
- RedHat
  - ch06s02
  - ch06s04
  - ch06s06
  - ch06s08
  - ch06s10
  - ch06s11

KdG Karel de Grote
Hogeschool