Page 1 and 2: question without commands.
Page 3 – 7 : questions with answers.

Aliases are set manually or in a login script. Aliases that were set manually will be gone the next time you login.

1. List your aliases

2. Remove the alias *ll*.

3. Create an alias called *?* That tells you who you are.

4. Where is the *ls* command located on your system?

5. Copy the *ls* binary to your login directory and call it *lijst*.

6. What happens if you execute the command by typing *lijst*?

7. Execute the command by specifying an absolute path.

8. Add your login directory to the *PATH* variable and use the *which* command to locate the binary *lijst*.

9. List the contents of the file /etc/hosts

10. List the contents of the file /etc/hosts again but now use output redirection to save the output in the file machines in your login directory.

11. Rename the file *machines* in *hostfile*.

12. What are the permissions of the the *lijst?*

13. Remove the execute bit. What happens if you execute the lijst command?

14. Create a file called *names* in your login directory with the following content.

jan dekker
harry koster
wim de Bie
jan janssen
wimmie van wemmenhoven

- List all lines that contain the string *jan*
- List all lines that begin with *ja*.
- List the first names of all entries.
- List all entries that end with *er*.

15. Create a variable called *number* and store the value *100* in it.
    Start a new shell.
    What is the value of the variable *number*.
    Exit the shell.

16. Make sure that the variable *number* is also available in a new shell that you start from the current shell.

17. Make sure that when you login, the variable *number* is always set, and

that the variable is passed on to the next shell.

18. Create a directory *newdir* in your login directory.

19. Move all files from your login directory to the directory *newdir*.

20. Create a tarfile called newdir.tar from the *newdir* directory.
The file should be created in your login dir.

21. Remove the *newdir* directory, including all files in it.

23. Restore the newdir directory from the tarfile.

24. Create a file called *file1* that contains *line1.*

25. Create a hardlink called *file1-h1*

26. Change the permission of the file *file1-hl* to *rwx*------.

What are the permissions the file *file1*?
(Are they the same as file1-lh?)

27. Create a symbolic link called *file1-sl* from *file1*.

28. List the inodes of *file1*, *file1-hl* and *file-sl*.

29. Remove *file1-sl*.

30. List the first two lines of the file *newdir/names*.

31. Use *sed* to change the string *harry* in *henry* in the file *newdir/names*.

32. Go to your *.ssh* directory and remove all files.

33. Create a new public/private key pair.

34. Use ssh to login to your own vm via *localhost*.

35. List the content of the .ssh directory. What new file
do you see and what is in it?

35a. Copy the publickey file to authorized_keys

36. Change the permission of the *authorized_keys* to *rw*-------.

37. Exit your ssh session to *localhost*. And start a new session.
Did you have to enter your password, to login?

38. Use the *su* command to change your effective user-id to *0*.
Create a user called *tempuser*. And give the user a password.

39. What is the difference between '*su*' and '*su –*'?

When you use the dash, the login scripts of the user
are also executed, so the environment of the user is
als established, next to the change of the effective
user-id.

39. Login as *linuser* and try to reboot the system using *sudo*.

40. Allow *linuser* to reboot the system.

Aliases are set manually or in a login script. Aliases that were set manually will be gone the next time you login.

1. List your aliases

```
[linuser@ipa ~]$ alias
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l.='ls -d .* --color=auto'
alias ll='ls -l --color=auto'
alias ls='ls --color=auto'
alias vi='vim'
```

2. Remove the alias *ll*.

```
[linuser@ipa ~]$ unalias ll
```

3. Create an alias called *?* That tells you who you are.

```
[linuser@ipa ~]$ alias ?="who am i"
[linuser@ipa ~]$ ?
root     pts/2        2017-12-05 15:44 (pi159)
```

4. Where is the *ls* command located on your system?

```
[linuser@ipa ~]$ which ls
alias ls='ls --color=auto'
      /bin/ls
```

5. Copy the *ls* binary to your login directory and call it *lijst*.

```
[linuser@vm0 ~]$ cp /bin/ls ~/lijst
```

6. What happens if you execute the command by typing *lijst*?

```
[linuser@vm0 ~]$ lijst
bash: lijst: command not found...
```

7. Execute the command by specifying an absolute path.

```
[linuser@vm0 ~]$ /home/linuser/lijst
lijst
```

8. Add your login directory to the *PATH* variable and use the *which* command to locate the binary *lijst*.

```
[linuser@vm0 ~]$ PATH=$PATH:~
[linuser@vm0 ~]$ which lijst
~/lijst
[linuser@vm0 ~]$ lijst
lijst
```

Note: Instead of running PATH=$PATH:~ you could also run PATH=$PATH:$HOME or PATH=$PATH:/home/linuser

9. List the contents of the file /etc/hosts

```
[linuser@vm0 ~]$ cat /etc/hosts
127.0.0.1   localhost localhost.localdomain
::1         localhost localhost.localdomain
```

10. List the contents of the file /etc/hosts again
but now use output redirection to save the output
in the file machines in your login directory.

```
[linuser@vm0 ~]$ cat /etc/hosts > ~/machines
```

11. Rename the file *machines* in *hostfile*.

```
[linuser@vm0 ~]$ mv ~/machines ~/hostfile
```

12. What are the permissions of the the *lijst?*

```
[linuser@vm0 ~]$ ls -l lijst
-rwxr-xr-x. 1 linuser linuser 117656 Dec  5 10:16 lijst
```

13. Remove the execute bit. What happens if you execute the
lijst command?

```
[linuser@vm0 ~]$ ls -l
total 120
-rw-rw-r--. 1 linuser linuser    158 Dec  5 10:32 hostfile
-rw-r--r--. 1 linuser linuser 117656 Dec  5 10:16 lijst

[linuser@vm0 ~]$ lijst
-bash: /home/linuser/lijst: Permission denied
```

14. Create a file called *names* in your login directory with the
following content.

```
jan dekker
harry koster
wim de Bie
jan janssen
wimmie van wemmenhoven
```

Now, get the following info from the file *names*.

- List all lines that contain the string *jan*
- *[linuser@vm0 ~]$* ***grep jan names***
- List all lines that begin with *ja*.
- [linuser@vm0 ~]$ **grep "^ja" names**
- List the first names of all entries.
- [linuser@vm0 ~]$ **cut -f1 -d" " names**
- List all entries that end with *er*.
- [linuser@vm0 ~]$ **grep "er$" names**

15. Create a variable called *number* and store the value *100* in it.
    Start a new shell.
    What is the value of the variable *number*.
    Exit the shell.

```
[linuser@vm0 ~]$ number=100
[linuser@vm0 ~]$ bash
[linuser@vm0 ~]$ echo $number
[linuser@vm0 ~]$ exit
```

16. Make sure that the variable *number* is also available in a new shell
that you start from the current shell.

```
[linuser@vm0 ~]$ export number
[linuser@vm0 ~]$ bash
[linuser@vm0 ~]$ echo $number
100
```

17. Make sure that when you login, the variable *number* is always set, and

that the variable is passed on to the next shell.

```
[linuser@vm0 ~]$ echo "export number=100" >> ~/.bash_profile
```

18. Create a directory *newdir* in your login directory.

```
[linuser@vm0 ~]$ mkdir newdir
```

19. Move all files from your login directory to the directory *newdir*.

```
[linuser@vm0 ~]$ mv * newdir
```

20. Create a tarfile called newdir.tar from the *newdir* directory.
The file should be created in your login dir.

```
[linuser@vm0 ~]$ tar cvf newdir.tar newdir
newdir/
newdir/hostfile
newdir/lijst
newdir/names
```

21. Remove the *newdir* directory, including all files in it.

```
[linuser@vm0 ~]$ rm -rf newdir
```

23. Restore the newdir directory from the tarfile.

```
[linuser@vm0 ~]$ tar xvf newdir.tar
```

24. Create a file called *file1* that contains *line1.*

```
[linuser@vm0 ~]$ echo "line1" > file1
```

25. Create a hardlink called *file1-h1*

```
[linuser@vm0 ~]$ ln file1 file1-hl
```

26. Change the permission of the file *file1-hl* to *rwx*------.

```
[linuser@vm0 ~]$ chmod 700 file1-hl
```

What are the permissions the file *file1*?
(Are they the same as file1-lh?)

27. Create a symbolic link called *file1-sl* from *file1*.

```
[linuser@vm0 ~]$ ln -s file1 file1-sl
```

28. List the inodes of *file1*, *file1-hl* and *file-sl*.

```
[linuser@vm0 ~]$ ls -li file*
5065806 -rwx------. 2 linuser linuser 6 Dec  5 11:18 file1
5065806 -rwx------. 2 linuser linuser 6 Dec  5 11:18 file1-hl
5065809 lrwxrwxrwx. 1 linuser linuser 5 Dec  5 11:25 file1-sl -> file1
```

29. Remove *file1-sl*.

```
[linuser@vm0 ~]$ rm file1-sl
```

What is the content of the file *file1*?
What is the content of the file *file1-hl*?

30. List the first two lines of the file *newdir/names*.

```
[linuser@vm0 ~]$ head -2 newdir/names
jan dekker
harry koster
```

31. Use *sed* to change the string *harry* in *henry* in the file *newdir/names*.

```
[linuser@vm0 ~]$ sed -i 's/harry/henry/' newdir/names
```

32. Go to your *.ssh* directory and remove all files.

```
[linuser@vm0 ~]$ cd .ssh
[linuser@vm0 .ssh]$ rm -rf *
```

33. Create a new public/private key pair.

```
[linuser@vm0 .ssh]$ ssh-keygen -t rsa
```

34. Use ssh to login to your own vm via *localhost*.

```
[linuser@vm0 .ssh]$ ssh localhost
```

35. List the content of the .ssh directory. What new file
do you see and what is in it?

35a. Copy the publickey file to authorized_keys

```
[linuser@vm0 ~]$ cp id_rsa.pub authorized_keys

[linuser@vm0 ~]$ ls .ssh
id_rsa  id_rsa.pub  known_hosts

[linuser@vm0 ~]$ cat .ssh/known_hosts
localhost ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBFgO2A8Z5TC1Eajnn9oJehLgODxt
dJnkhFfTfrrvMGBOtfApV7BoQ56aeziJCZHxGj9vSwuqDDIOTJTMam8FGuQ=
```

36. Change the permission of the *authorized_keys* to *rw-------*.

```
[linuser@vm0 .ssh]$ chmod 600 .ssh/authorized_keys
```

37. Exit your ssh session to *localhost*. And start a new session.
Did you have to enter your password, to login?

```
[linuser@vm0 .ssh]$ exit
logout
Connection to localhost closed.
[linuser@vm0 .ssh]$ ssh localhost
Last login: Tue Dec  5 12:02:58 2017 from ::1
```

38. Use the *su* command to change your effective user-id to *0*.
Create a user called *tempuser*. And give the user a password.

```
[root@vm0 linuser]# useradd -m tempuser
[root@vm0 linuser]# passwd tempuser
Changing password for user tempuser.
```

39. What is the difference between '*su*' and '*su –*'?

When you use the dash, the login scripts of the user
are also executed, so the environment of the user is
als established, next to the change of the effective

`user-id.`

39. Login as *linuser* and try to reboot the system using *sudo*.

```
[linuser@vm0 ~]$ sudo reboot

We trust you have received the usual lecture from the  local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for linuser:
linuser is not in the sudoers file.  This incident will be reported.
```

40. Allow linuser to reboot the system.
Make sure you login as root.

> If you want to allow a user to execute a particular command you could do the following:
>
> – add the user to the file /etc/sudoers with the following contents.
>
> 1. ALL (means able to execute from all terminals)
> 2. the command that the user is allowed to execute.
>
> So, for example: if /sbin/reboot is the command, then this entry should be added to the configuration file /etc/sudoers.
>
> *linuser ALL= /sbin/reboot*

Run the following command to add linuser to */etc/sudoers*.

```
  echo “linuser ALL= /sbin/reboot” >> /etc/sudoers
```

Note: Make sure you use >> and not >.

Login as linuser and reboot the system.