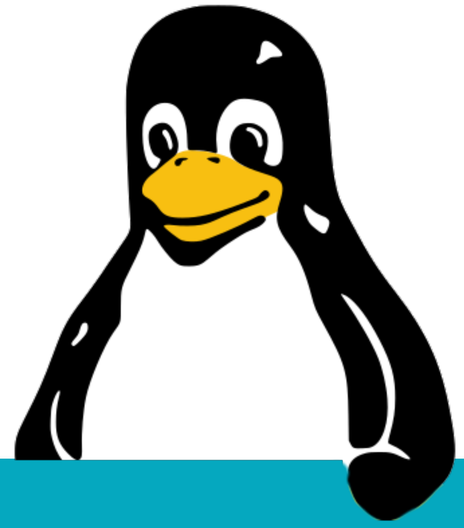
Linux, day 4



Objectives covered

Objective	Summary	Boek
3.1	Given a scenario, create simple shell scripts to automate common tasks.	3 4 25

LAB: The shell environment

Assignment

- Adjust your account's "*~/.bashrc*".
  - Add "*/opt/test*" to $PATH.
- Make the "*/opt/test*" directory.
  - Copy the "*ls*" binary to "*/opt/test/testls*".
- Do a new SSH login (or start a new Bash).
  - Can you now run "*testls*"?

Solution

```
As root:
# echo 'PATH=$PATH:/opt/test' >> /etc/bashrc

As you:
$ sudo mkdir -p /opt/test
$ sudo cp -p /usr/bin/ls /opt/test/testls

$ ssh localhost "which testls"
```

LAB: Shell scripting: a start

LAB: Create a simple script

- Create a script which:
  - Runs ;)
  - Reads a name from the first, passed parameter.
  - Asks for a greeting interactively (with *read*).
  - Outputs a greeting to the name.

Example

```
$ ./greeting.sh Tess
How would you like to be greeted?
Hello
Hello Tess
```

Solution

```
#!/bin/bash
NAME=$1

read -p "What greeting would you like? " GREET

echo "${GREET} ${NAME}"
```

Solution

```
$ chmod +x greeting.sh

$ ./greeting.sh Tess
What greeting would you like? Hello
Hello Tess
```

LAB: Shell scripting: I/O & Files

Assignment 1

- Find all files with "passwd" in their name.
- Find all files, literally and exactly called "passwd".
- Find all world-writable files.

Assignment 2

- SSH from one VM to the other,
- Use a HEREDOC to run:
  - *touch /tmp/foobar*
  - *ls /tmp*
  - *echo $(whoami) > /tmp/foobar*
  - *cat /tmp/foobar*

Solution

```
$ locate passwd
$ find / -type f -name "*passwd*"

$ locate -b '\passwd'
$ find / -type f -name "passwd"
```

Solution

```
$ find / -perm -o+w -type f
```

Solution

```
$ ssh localhost << EOF
touch /tmp/foobar
ls /tmp
EOF
```

LAB: Shell scripting: Flow

Assignment

- Write a shell script which:
  - Checks if it’s run as root; if not, "*exit 1*".
  - Reads a command from passed parameters.
    - This command is either "create" or "remove".
  - Asks for a number (for now, keep it <20).
  - Either creates or removes "user1", "user2", "user3" etc.
  - Uses HEREDOC to make "welcome.txt" in their homedir.

Assignment

- If you run it, this looks like:

```
$ sudo /tmp/dummy-users.sh create
How many?
5

$ cat /home/user5/welcome.txt    #works
```

Solution

- I have a sample solution on Github.
- It's here, in the Lesson 004 directory.

Closing

Other resources

- GNU.org Bash manual

- Exercism: gamified learning programming.
  - 80+ assignments for Bash.

- TLDP.org has some fun assignments.

Homework

- Reading:
  - Chapter 4: Processing and analyzing text
  - Chapter 25: Shell scripting

Homework

- Create a shell script to "*ping sweep*" a network.
  - The script should ask for a base address,
  - Assume a netmask of /24.
- e.g.: *pingsweep 192.168.10.0*

Reference materials

Resources

- Creating shell scripts in Enterprise Linux (PluralSight)
- A roundup of 15 popular Linux shells.
- Linux shell metacharacters
- Bash globbing tutorial
- Bash read command
- Bash getopts example

Resources

- Really in-depth on pipes, forks and more
- Testing file characteristics in Bash
- Why you don’t read lines with "for"
- An excellent, in-depth study guide for shell scripting
- Practice more, take the challenge!
- Or try Tutorials Ground

Resources

- Exercism: gamified learning programming.
- TLDP.org has some fun assignments.
- Arithmetic in Bash
- GNU.org Bash manual