VE482 — Introduction to Operating Systems

Scripting on Unix systems

Manuel — UM-JI (Fall 2019)

The goal of this document is to provide some basics on Bash scripting. The reader is highly encouraged to refer to the man pages, especially for Bash itself. Another important topic, not covered here, but related to Bash scripting is regular expressions; it is recommended to read about it.

1 Basic Unix commands

Simple commands to run basic tasks:

- Navigation: `ls, cd, dir`
- File and directory manipulation: `mv, rm, touch, cp, mkdir, rmdir`
- File reading: `cat, tac, head, tail`
- Text manipulation: `tr, cut, fmt, echo, less, more`
- Search: `find, grep`
- Permissions: `chmod, chown, chgrp, su`
- Processes: `kill, killall, pkill, pgrep, ps`
- File system: `du, df, mount, free`
- Network: `ifconfig, ssh, route, ping, iptables`
- Generic commands: `date, time, lsusb, alias, screen, seq`
- Development: `rsync, diff, patch, gcc, g++, make, git`
- More scripting tools: `awk, sed, perl`

Important topic for reading: regular expressions

2 Bash scripting

Bash allows much more than just writing commands on a terminal. It is in fact a Turing complete programming language best used for writing scripts.

2.1 Basics

```
#!/bin/bash
# the first line of a bash script is always #!/bin/bash
# anything following # is a comment
a=asd # assign asd to variable a
echo $a #display the content of variable a
$1 #first argument to the script
$2 # second argument to the script, more arguments $3, $4...
$@ # all the arguments
$! # last process number
$? # exit code from the previous command
# refer to bash man page for more advanced operations on variables e.g.
a="1234.jpg"; echo ${a%.jpg} ${a:2,3}
```

2.2 Conditional statements

```
[ expression = value ] # test an expression, man test for more details

# if statement
if [ $a =  "qwe" ] ; then
  list of statements
fi

# case keyword
case $i in
  a) list of statements
  ;;
  b) list of statements
  ;;
  *) list of statements #default actions
esac
```

2.3 Loops

```
# for loops (list is a space separated list of elements (e.g. filenames))
for i in list ; do
  list of statements
done

# for loops (iterate a predefined number pof times)
for((i=0; i<10; i++)) ; do
  list of statements
done

# while loops
while some expression ; do
  list of statements
done
```

2.4 Arrays

```
# simple array
a[3]=4; echo ${a[3]}
i=2; a[$i]=1

# associative array
declare -A b=([key1]=value1 [key 2]=value2);
echo ${b[key1]}; echo ${b[key2]}; echo ${b[key 2]};
c=key1; echo ${b[$c]}
```

2.5 Functions

```
function name () {
  core of the fuction
}
```