

Operating System fundamentals

Environment variables



Inhoud

1. Variables
2. Environment variables
3. Startup scripts

Course text

- chapter 5: Create, View and Edit Text Files



Variables

Creating variables

- we already saw how to create variables
- no need to declare them (type is always String)
- assignment:
`variableName=value`
- be careful not to put spaces around the =
- examples:
`count=40`
`full_name="John Doe"`
`file1=/tmp/abc`
`_id=RH123`
- variables are always created in the current shell

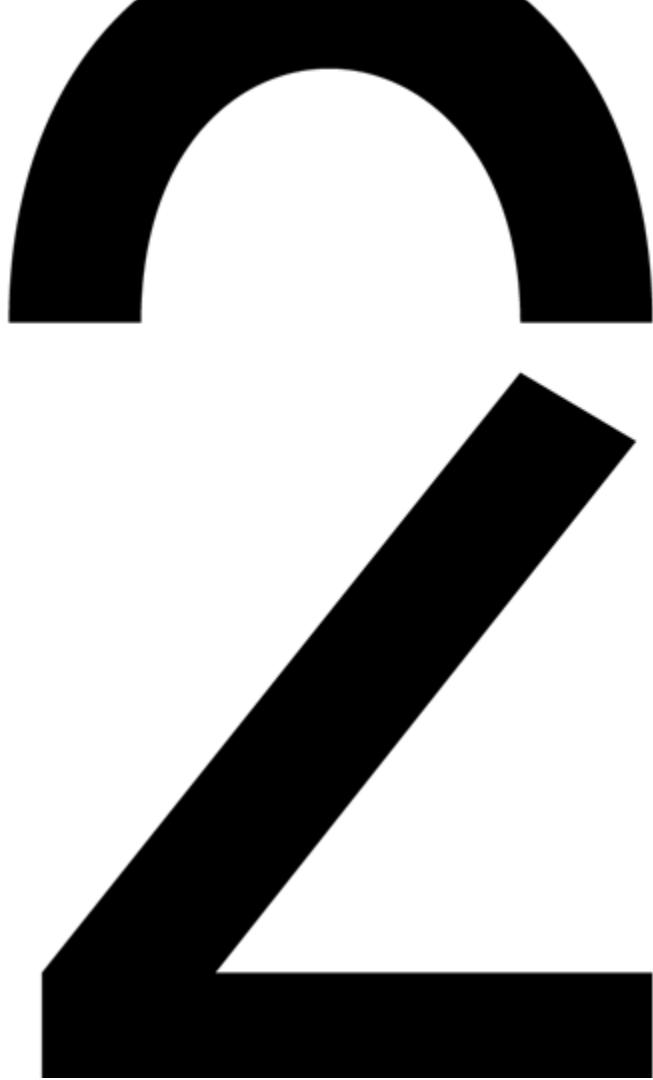
Using variables

- you use the value of a variable by putting a \$ in front of the name
- but it is better to put the name between \${}
count=30
echo \$count
echo \${count}
- echo \$countx\$count → does not work
- echo \${count}x\${count} → does work

Variables

- you can get a list of all variables with the “set” command
- this also shows the “environment variables” (their names are in uppercase)
- you can search for a specific variable with:
set | grep variableName
- you can delete a variable with the “**unset**” command

Environment variables



Environment variables

- environment variables are variables that are defined system wide
- they are shared with all programs that are spawned from the current shell
- by convention they get names in uppercase
- use the command “**env**” to see all the environment variables

Common environment variables

- the following environment variables are almost always available:
 - **HOME**: contains the path of your personal home directory
 - **USER**: contains your user name
 - **LANG**: contains your preferred language
 - **EDITOR**: contains your preferred editor(vi or nano)
 - **SHELL**: contains the path to the program executing your current shell
 - **PATH**: contains directories separated by : in which there are executables
- you can convert a variable into a environment variable with the “export” command

```
export PATH=${PATH}:/home/kris/bin
```

Common environment variables

- Some environment variables determine the behaviour of your shell and other commands

```
LANG=fr_FR.UTF-8
```

```
date
```

```
ls dfsdffsdg
```

```
LANG=en_US.UTF-8
```

```
date
```

```
ls dfsdffsdg
```

```
LANG=nl_BE.UTF-8
```

```
date
```

```
ls dfsdffsdg
```

Common environment variables

- Some environment variables determine the behaviour of your shell and other commands
- changing the format of the history:
`history`
HISTTIMEFORMAT= "%F %T: "
`history`
- changing the prompt:
`echo $PS1`
PS1= "[\u@\h \A \W]\\$"

Common environment variables

- **PATH** contains all paths in which executables are searched for
- the paths are separated with a :
- this also exists in Windows and MacOS
- you can extend the PATH with:
`export PATH=${PATH}:/home/kris/bin`
- to find out in which path a certain executable can be found, use the “which” command
which ls

Startup scripts

3

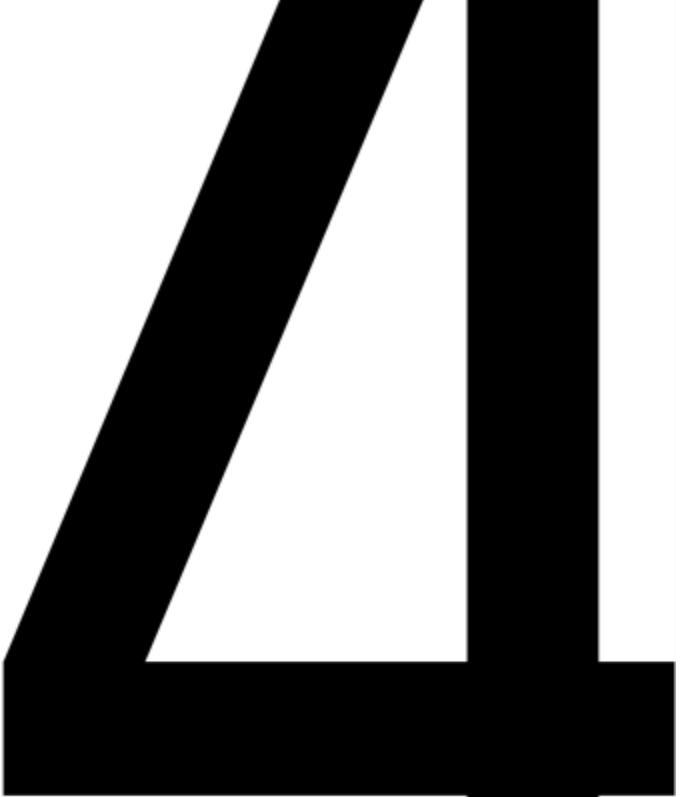
Startup scripts

- there are different scripts that are executed at startup of a shell:
 - /etc/profile
 - /etc/bashrc
 - ~/.bash_profile
 - ~/.bashrc
- e.g. if you want to change your favorite editor to nano,
you add a line to ~/.bashrc
`export EDITOR=nano`

Startup scripts

- one can also define “aliases” in `~/.bashrc`
`alias ll="ls -las"`
- example:
`alias hello='echo "hello world, hello $USER" '`
`hello`
- use “unalias” to remove an alias

Exercises



Exercises

- KdG
- RedHat
 - ch05s06
 - ch05s07