TECHNOLOGY ANALYST 2020

LINUX INTRO

- GET AROUND LINUX
- USE BASIC FILE COMMANDS
- IDENTIFY AND MANAGE PROCESSES
- WORK WITH LOG FILES

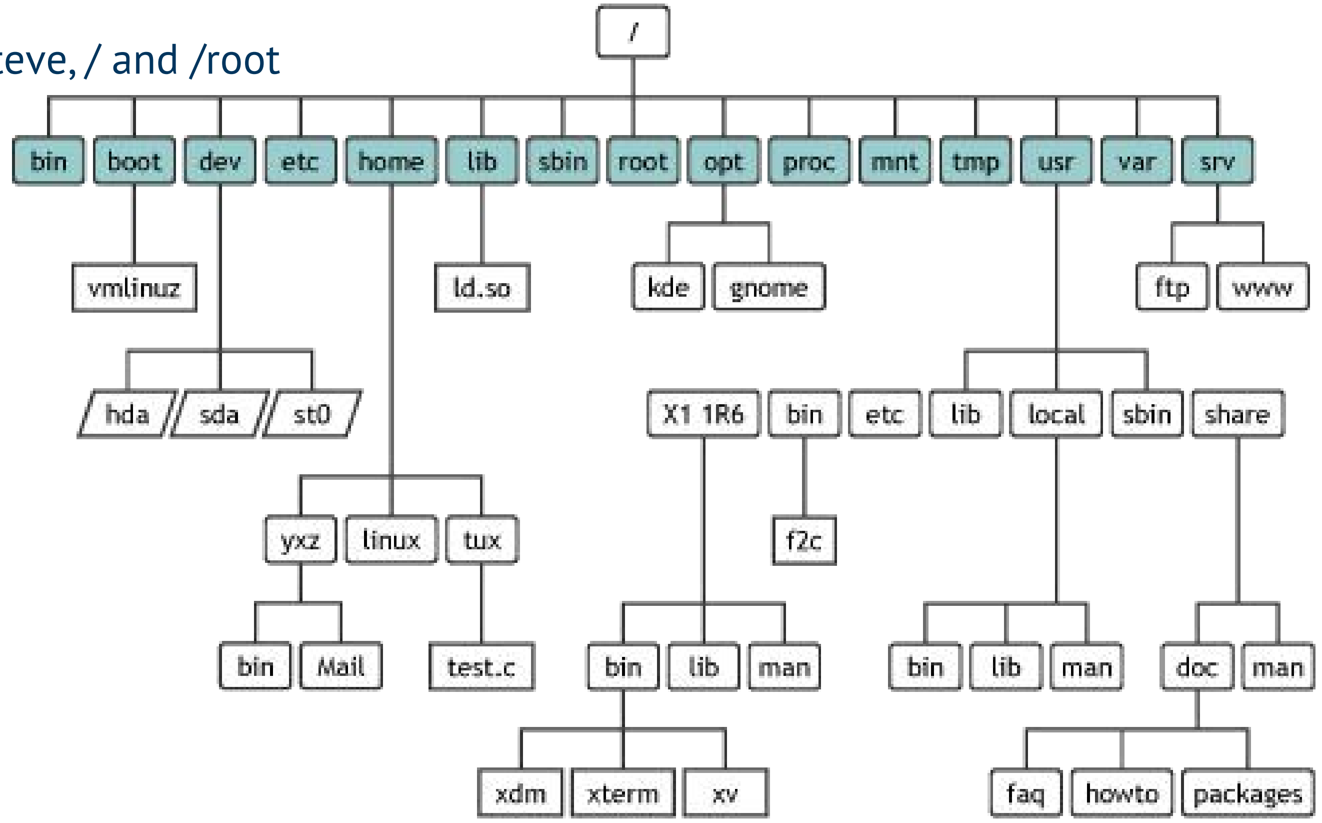
BASICS

LOGGING ON & WHO ARE YOU

- Using an SSH Client we will log on to Linux
- Who are you logged in as?

LOGGING ON & WHO ARE YOU

- The Linux file system in short a quick tour
  - /tmp, /home, /usr, /var what are they?
  - Names for directories like /home, /home/steve, / and /root



LOGGING ON & WHO ARE YOU

- Using CD to get about and some useful short cuts
  - Absolute and relative paths
  - tilde (~) and minus (-)

CREATING DIRECTORIES

- Creating directories with mkdir
- Creating child directories when the parent doesn’t exist

TASK

- Search the web to find out how to create child directories using mkdir if the parent does not exist
- Once you have found the command create the directory structure steve/nick/frank

VIEW FILE ATTRIBUTES

- Viewing the attributes of files and directories with ls
  - Core options
    - -l   -a   -ltr   -S   -R

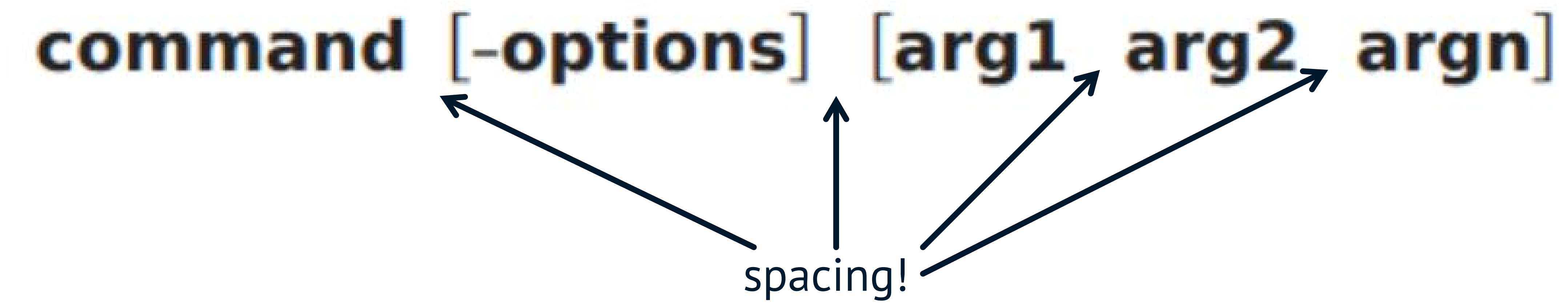
- What do all those columns mean?

TASK

- List attributes of all files and directories in /etc/cron.daily
- List the file/directories in /etc/cron.daily in ascending order of size
- Show all hidden files in your home directory
- How do I change the output of the -l option to show a more readable file size? e.g. MB, GB, etc. List the /etc/cron.daily directory using the option you find.

COMMAND SYNTAX

- All commands are lowercase
- Spacing is important
  - Between command and options
  - Between options and arguments
  - Between all arguments



```
command [-options] [arg1 arg2 argn]
```

GETTING HELP

- Use the Internet
  - GOOGLE
    - man ls
    - man cp
- stackoverflow.com
  - Some useful answers, but not always the simplest, or correct so you should try them out
  - When trying out destructive commands make sure you create a directory with some test files in
- The built in manual called **man**
- When searching the web also include the operating system type of RHEL or Debian

REFERENCES

- https://www.tecmint.com/cd-command-in-linux/
- https://www.tecmint.com/15-basic-ls-command-examples-in-linux/
- https://shapeshed.com/unix-ls/

BASICS EXERCISE (10 MINUTES)

- List the files and directories in /etc
- List the files and directories in your home directory
- List all the files and directories in your home directory
- Recursively list the /usr/include directory
- Change to the following locations using absolute path
  - /usr/bin
  - /usr/local/bin
  - /var/log
- Change to the following locations using relative paths, starting from /var/log and then relative from the next, and so on
  - /etc/rc.d
  - /usr/share
  - /usr/lib

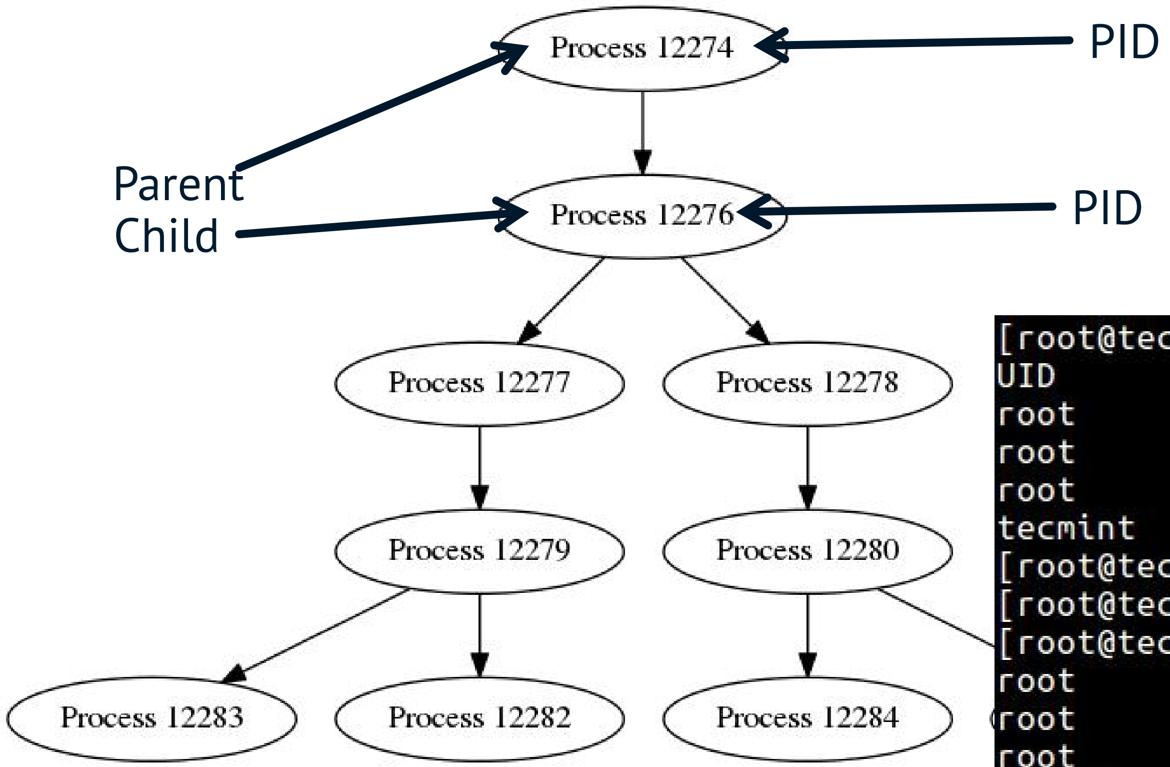
PROCESSES

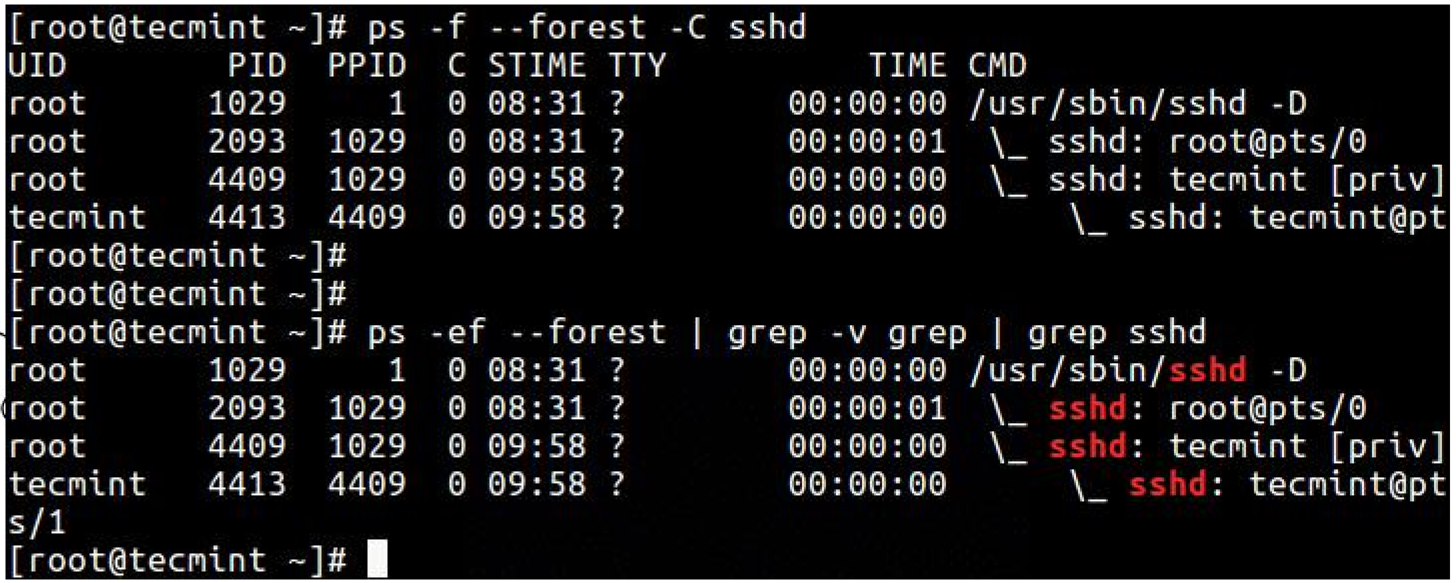
WHAT IS A PROCESS

- All process start as a file containing executable code called a binary
- When we type in the command and press enter it is allocated memory, resource and CPU time

- The PS command to show process status
  - your own processes
  - all processes with extra attributes and what the columns mean
  - getting other attributes using -o

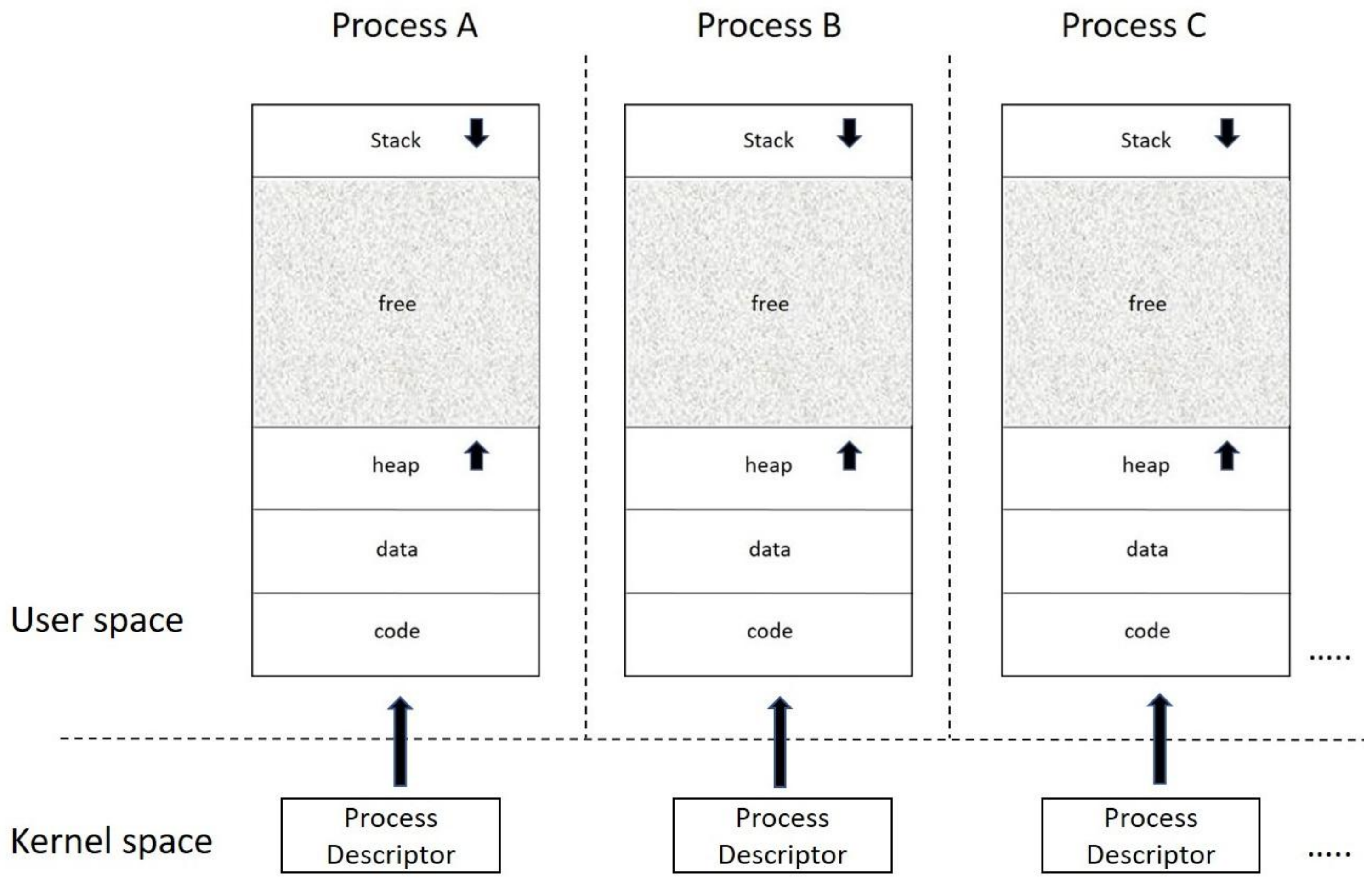
PROCESS TREE





```
[root@tecmint ~]# ps -f --forest -C sshd
UID          PID  PPID  C STIME TTY          TIME CMD
root        1029     1  0 08:31 ?        00:00:00 /usr/sbin/sshd -D
root        2093  1029  0 08:31 ?        00:00:01  \_ sshd: root@pts/0
root        4409  1029  0 09:58 ?        00:00:00  \_ sshd: tecmint [priv]
tecmint     4413  4409  0 09:58 ?        00:00:00      \_ sshd: tecmint@pt
[root@tecmint ~]#
[root@tecmint ~]#
[root@tecmint ~]# ps -ef --forest | grep -v grep | grep sshd
root        1029     1  0 08:31 ?        00:00:00 /usr/sbin/sshd -D
root        2093  1029  0 08:31 ?        00:00:01  \_ sshd: root@pts/0
root        4409  1029  0 09:58 ?        00:00:00  \_ sshd: tecmint [priv]
tecmint     4413  4409  0 09:58 ?        00:00:00      \_ sshd: tecmint@pt
s/1
[root@tecmint ~]#
```

WHAT'S IN A PROCESS



MANAGING PROCESSES

- We can signal processes with the **kill** command or some special CONTROL key presses
  - CTRL+C = terminate currently running process on my terminal
  - CTRL+Z = stop/sleep the currently running process on my terminal so I can start it later
  - kill *PID* = terminate the process with the number *PID* - NOTE only your processes
  - kill -9 *PID* = terminate the process regardless - can be dangerous
- A signal informs the process to do something whilst it is running
- Default it to terminate and free up all resources

OTHER COMMANDS

- Some other commands to look at in the future for managing processes

- strace
  - View the C library calls on a live running program to see what it is doing

- lsof
  - View all the files and network ports a program has open

REFERENCES

- https://www.cyberciti.biz/faq/how-to-check-running-process-in-linux-using-command-line/
- https://opensource.com/article/18/9/linux-commands-process-management
- https://www.tecmint.com/ps-command-examples-for-linux-process-monitoring/

PROCESS EXERCISE (10 MINUTES)

- Identify the daemon processes on your system (look for the processes whos TTY = ?)
- Identify the processes you are running
- Let us start up a few more shell processes. Type in the following commands;
    - vi test
    - :!sh   # whilst in VI
    - bash
  - Can you now list all the current processes that you have running?
- Can you now terminate the vi process?

KNOWING YOUR ENVIRONMENT

WHAT ARE VARIABLES?

- Imaging the scenario
  - We have 3 environments DEV, QA and PROD
  - Our application uses a database to store important information, e.g. users account details
  - Should DEV applications be able to query the production database?
  - Should the PROD environment applications be able to connect with the DEV database?
- How could I create my application so that I only need to package it the once?
  - BUT - allow it to run in any of the 3 environments, but allow you to change the Database connection details?

CREATING VARIABLES

- Let’s say our application has a variable for the Database server hostname called DBHOST
- Our database hostnames for DEV and PROD are;
  - DEV is db.dev.neueda.com
  - PROD is db.prod.neueda.com
- Our variable the must be set before running the command;
  - For DEV
    - DBHOST=db.dev.neueda.com
  - For PROD
    - DBHOST=db.prod.neueda.com

MAKING THE VARIABLE AVAILABLE

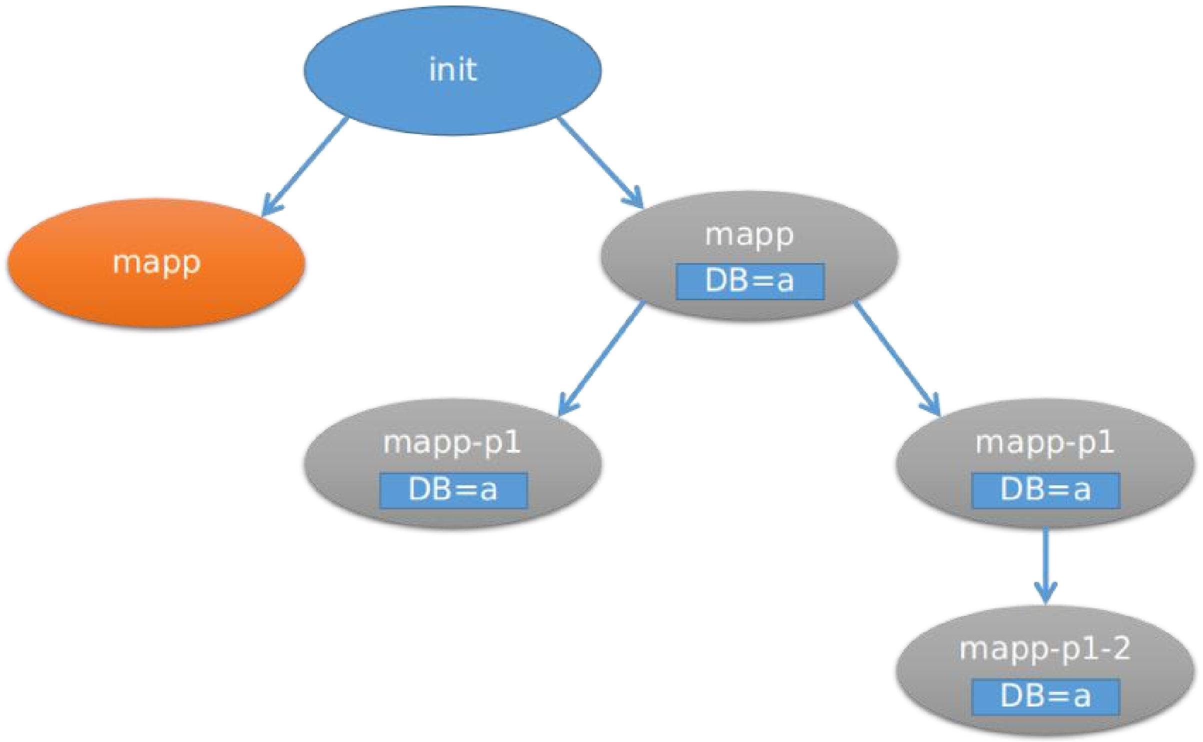
- Creating a variable doesn’t make it available to your program
- If you want to make yourself available to the world you go on holiday or hope that work has an opening
- If a manufacturing company want’s to make its products available to the world they ….. their goods

EXPORTING

- export DBHOST
- For BASH and KSH shells
  - export DBHOST=db.dev.neueda.com

INHERITANCE

- In the loosest form



VIEWING VARIABLES

- To view all variables for your terminal shell session
  - set
- To view all exported variables for your terminal shell session
  - env
- To view a specific variable
  - echo $HOME
  - echo $TTY
- To remove a variable from your session
  - unset DBHOST

Neueda

REFERENCES

- https://www.tutorialspoint.com/unix/unix-using-variables.htm
- https://www.tutorialspoint.com/unix/unix-special-variables.htm
- https://linuxize.com/post/how-to-set-and-list-environment-variables-in-linux/

ENVIRONMENT EXERCISE (5 - 10 MINUTES)

- Type in **man ls**
- Check the current state of MANPATH variable
  - Does it have a value?
  - Does it exist in the environment?
- Create the variable **MANPATH** and set the value to **hello**
- Can you still view the ls help page using **man ls**?
- Now make **MANPATH** an environment variable
- Can you view the **ls** help page now?
- Now make the pages work again using the **unset** command
- Check that you can view the **ls** help page

WORKING WITH FILES

BEFORE WE START

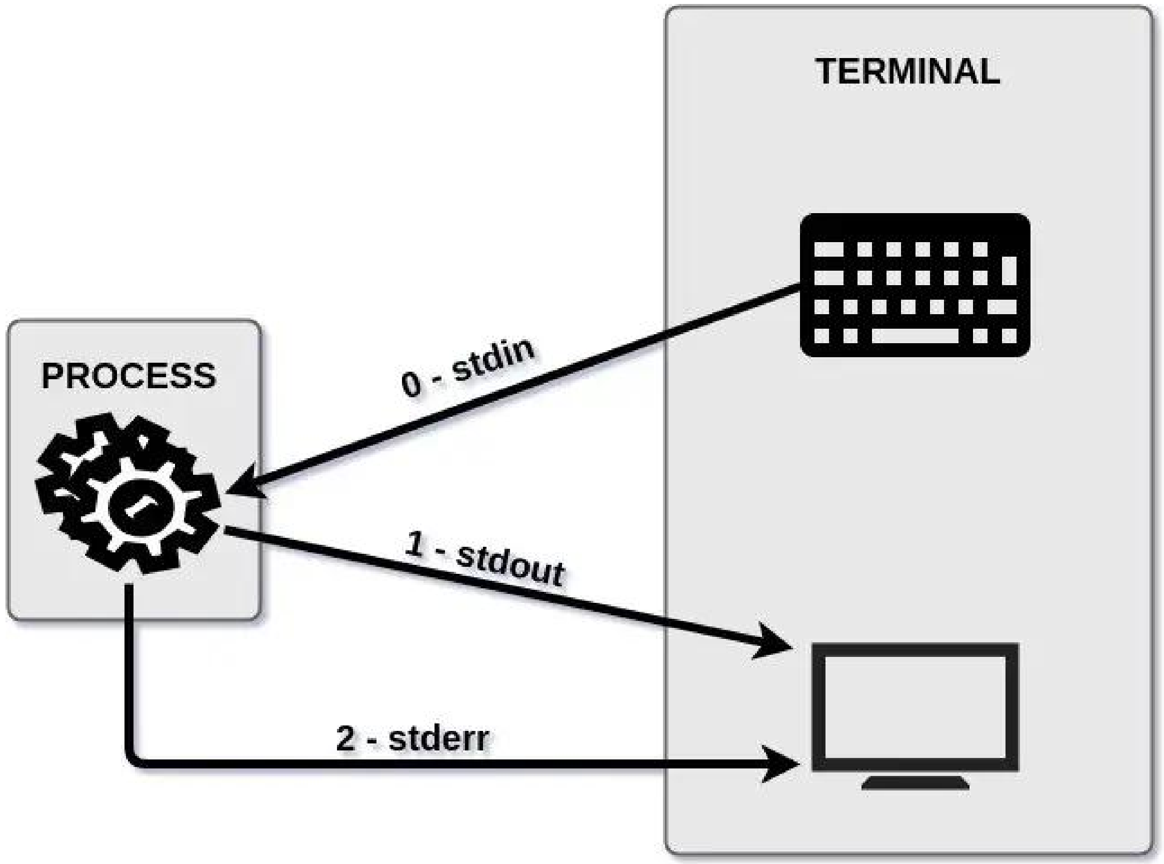
- Let’s download some files from the web to our account
- An example system profile
  - https://www.dropbox.com/s/zz7ah9v1o40fhav/profile?dl=0
- An example system log normally /var/log/messages
  - https://www.dropbox.com/s/4bzvkjhvw34nzkt/ExampleMessages.log?dl=0

- How to
  - curl -so etc_profile https://www.dropbox.com/s/zz7ah9v1o40fhav/profile?dl=0
  - curl -so ExampleMessages.log https://www.dropbox.com/s/4bzvkjhvw34nzkt/ExampleMessages.log?dl=0

- On a shared server you can copy all the files from **/home/shared**

REDIRECTING OUTPUT

- All processes get 3 file descriptors (streams, file handles) by default



EXAMPLES - CAPTURING OUTPUT

- Capturing output using >
- NOTE: This will create the file if it does not exist
- NOTE: This will empty any files that exists
- **IMPORTANT: Don’t redirect to the file you are using!**

EXAMPLES - APPENDING OUTPUT

- Capturing output with >>
- NOTE: This will create the file if it does not exist
- NOTE: This will add to the end of the file (append) if it does exist

CAPTURING ERRORS

- Output captured using the 2> or 2>>
- NOTE: Same rules apply for single > or double >>
- Merging stdout and stderr into the same file use stdout redirection followed by 2>&1
  - e.g.   ls /tmp /nothing >output.txt 2>&1

THROWING OUT THE RUBBISH

- A special device file exists that will not fill up the disk

- Use this device when you are not interested in
  - Storing the output for stdout or stderr
  - Don’t want to see the output of stderr on the screen to make your view easier

FILE MANIPULATION

- 4 commands
  - Copy = cp
  - Rename/Move = mv
  - Delete = rm
  - Delete Directory = rmdir or rm -r
- Syntax of cp and mv
  - command *src dest*
  - command *src src src …. directory*

CAN'T READ OR WRITE TO MY FILE, GRRRRR!

- Check the owner of the file
  - ls -l
  - Look at the 3rd column - is it you?
  - Look at the 4th column, only if the 3rd isn't you - are you a member of that group
  - Remember the id command
- Check the permissions in the 1st column of ls -l
  - If you own the file (3rd column) then check letters in the 1st column 2 - 4
  - If you are a group member (4th column) then check the letters in the 1st column 5 - 7
  - Otherwise only the last 3 characters are what you can do
- If you own the file you can change access
  - chmod u+r file
  - chomd u-r file

REFERENCES

- https://www.digitalocean.com/community/tutorials/an-introduction-to-linux-i-o-redirection
- https://www.tutorialspoint.com/unix/unix-io-redirections.htm
- https://devconnected.com/input-output-redirection-on-linux-explained/
- https://www.putorius.net/linux-io-file-descriptors-and-redirection.html
- http://linuxcommand.org/lc3_lts0050.php
- https://ryanstutorials.net/linuxtutorial/filemanipulation.php
- https://www.baeldung.com/linux/file-manipulation
- https://developer.ibm.com/technologies/linux/tutorials/l-lpic1-103-3/

BREAK FOR 15 MINUTES

BASICS OF VIEWING FILES

WHAT TYPE OF FILE IS IT?

- Linux does not use file extensions
  - Although some programs might

- Files are identified through 1st 2 bytes of the content

- Use **file** to identify

VIEWING FILE CONTENT

- Commands to use

- cat
- less
  - space = pg dn, b = pg up, /searchptn, n = next search ptn, N = previous search ptn, q = quit
- head
- tail
- tr

TASK

- Using **cat** how can I show line numbers?
- How do I show the first 5 lines of /etc/passwd?
- How do I show the last 3 lines of /etc/passwd?
- How do I skip the first 3 lines and show the rest of the file, or how do I start output from line 4 to the end of the file /etc/passwd?

LOG FILES

- What are log files?
- Where are they normally stored?
  - /var/log
  - With the application in it’s deployment location
- How do I know how they are formatted?

REFERENCES

- https://2buntu.com/articles/1491/viewing-text-files-on-linux-cat-head-tail-more-and-less/
- https://learnbyexample.gitbooks.io/command-line-text-processing/content/tail_less_cat_head.html

FILES EXERCISE (15 MINUTES)

- Create a copy of the /etc/profile in your home directory and call it ourprofile using cat and redirection to stdout
- Using the following command
  - cat >$HOME/mytextfile
  - Keep typing text in with new lines, at least 5 lines
  - On an empty line (after pressing enter) press CTRL+D
- Use head to show the first 2 lines of your file mytextfile and you should recognise what you typed in
- Use tail to show the last 2 lines of mytextfile
- Turn your mytextfile into all uppercase letters and store in a file called bigtext
- Now make the bigtext file so that you cannot read it
- Try to view the file bigtext with the command cat.
- Fix the file so that you can view it again

VIEWING DATA FROM PROGRAMS

STUDENT TASK

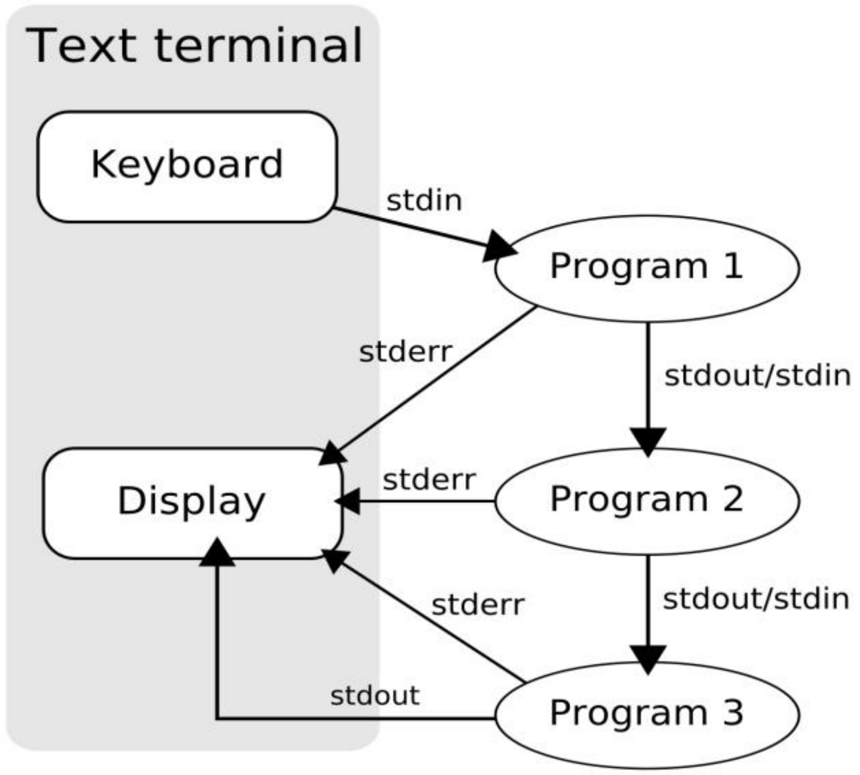
- How would you get the output of a full process listing and then convert that output to all uppercase?
  - Think of what we have covered so far

WHAT IS A PIPELINE

- An efficient way to pass output of one command as input to another via shared memory
- Let’s look at how we can take the last example and make it a pipeline

WHAT GOES THROUGH THE PIPELINE

- Only stdout goes through the pipeline



EXAMPLES - REVISITING SOME OLD FRIENDS

- less
- head
- tail

TASKS

- Which command allows us to view text a page at a time?
- A command to show me file attributes in /etc for all files and directories
- A command that will only show me the first 10 lines of the system processes
- How can I remove the header before passing it through to head?
- Get me 3 lines following the 10th line of the /etc/passwd file, that’s lines 11, 12 and 13

SORTING DATA

- What is sorting?
- How does sort identify columns
- Sort examples
  - Reverse sort the file names
    - ls | sort -r
  - Show PIDS in reverse order
    - ps -ef | sort -k2nr

TASK

- What column in ls -l is the file size?
- If n = numeric sort, what character do you think will reverse?
- Now sort the file names in reverse order
- Show the process which has spent the longest on the processor, page it so to see the output

COUNTING

- The **wc** command is useful command for counting
  - lines
  - words
  - characters

UNIQUE

- The uniq command is used to
  - Identify unique items
  - Remove duplicates
  - Show only duplicates
  - Count the number of occurrences of each unique pattern

- Can only be used on sorted data
  - All unique items must be consecutive

REFERENCES

- https://ryanstutorials.net/linuxtutorial/piping.php
- https://www.geeksforgeeks.org/piping-in-unix-or-linux/
- https://www.geeksforgeeks.org/sort-command-linuxunix-examples/
- https://www.computerhope.com/unix/usort.htm
- https://shapeshed.com/unix-sort/

FILTERING DATA

EXTRACTING COLUMNS WITH AWK

- 2 commands **cut** and **awk**

- CUT is very basic, but good if you want to get characters
  - Column order cannot be changed

- AWK is a full scripting language, but great for extracting columns
  - We'll show you how to
    - Extract columns with print and printf
    - Change order of output
    - Change the input field separator
    - Get the last column
    - Search for something in a particular column

GRABBING LINES WITH GREP

- A pattern matching command that displays lines matching your specified pattern
- In simplest form can grab character matching
  - grep local /etc/profile
  - grep root /etc/passwd

- NOTE: local could also match locale or delocalize or localitise

- Word boundaries are denoted by using **\b**

TASK

- How could i find any errors in a file called /var/log/messages

CASE SENSITIVE AND SPACES

- The pattern is case sensitive
- To ignore case use the **-i** option
- To invert the match use **-v** option
- If you're looking for multiple words, or you have spaces in your search you must quote the pattern

REGULAR EXPRESSIONS

- Regular expressions allow you to match variations in words
  - e.g. Helo, Hello, Hellllooooo
  - Mapping out a log file to get to a specific part you need

- A regular expression is a combination of a set of special characters

- Used in many language today, but started in Unix
  - Java
  - Python

BASIC CHARACTER SET

- Meta-Characters - characters that describe characters

.	= any single character
[aeiou]	= a single character that will match either a or e or i or o or u
[a-z]	= a single character that will match a lowercase letter
[a-z0-9]	= a single character that will match either a lowercase letter or a digit
[^0-9]	= not matching a single digit

- Anchors - stop matching either side of my pattern

^	= Match the pattern from the beginning of the line
$	= Match the pattern at the end of the line

- Quantifiers - how many times does that character occur?

*	= zero or more occurances of the character before the *, e.g. a*
{x,y}	= a specific number of occurances, one or more = {1,}, exactly 10 = {10}, between 5 and 8 = {5,8}

TASKS

- Find all users in the /etc/passwd file where the user ID is a single digit only
- Find all users who do not have a home directory in /home

GREP IN A PIPELINE

- Grep takes standard input if no file is specified
- Let’s use **curl** a command line program to access web pages

- Examples
  - Let’s look for the Lucky button on the Google search page
  - Let’s find all the <a href= values in the bbc.co.uk website

- How do we create an expression for matching Email addresses
- How do we create an expression for matching credit card number entry

TASK

- Create a regular expression for a UK and NI telephone number

MANIPULATING DATA

- To alter data we use **sed**
- Unlike a text editor which is interactive, sed is a stream editor
- All actions must be supplied up front
- Sed only works with the input provided as a file or stdin
- By default the modified text is sent to stdout
- Linux versions of sed allow for the original file to be updated in place
- Uses regular expressions

EXAMPLES

- Delete lines 1 through 10 in the output of /etc/passwd;
- Only print lines 5 through 10
- Find the letters root in the file
- Show users from adm to nobody
- Replace the first occurance on every line matching the letters **root** with **Superman**
- Replace all occurances of **root** with **Superman**

TASK

- Using ps -ef print out only the command and arguments
- Write a sed command that will ensure that a 16 digit credit card number will only ever be 16 digits with no other characters. The user may supply the credit card with spaces, - or . characters between each 4 characters.

GREP AND PIPELINES

- Remember our counting and uniqueness commands?
- Here we will construct a command to tell us how many plain files and how many directories there are in /etc

REFERENCES

- AWK
  - https://www.geeksforgeeks.org/awk-command-unixlinux-examples/
  - https://www.howtogeek.com/562941/how-to-use-the-awk-command-on-linux/
  - https://www.tutorialspoint.com/awk/index.htm
- GREP
  - https://www.geeksforgeeks.org/grep-command-in-unixlinux/
  - https://phoenixnap.com/kb/grep-command-linux-unix-examples
  - https://www.howtogeek.com/496056/how-to-use-the-grep-command-on-linux/
  - https://www.tutorialspoint.com/unix_commands/grep.htm

REFERENCES

- SED
  - https://www.geeksforgeeks.org/sed-command-in-linux-unix-with-examples/
  - https://www.digitalocean.com/community/tutorials/the-basics-of-using-the-sed-stream-editor-to-manipulate-text-in-linux
  - https://www.tutorialspoint.com/sed/index.htm

THANKYOU