

Meerza Ahmed | DSCI 551

\*Due 1/19/24 @ 11:59 PM

**Q1: [4 pts]** Explain what each of the following **commands** does, and what each of its **arguments** means.

- `chmod 400 "dsci2024.pem"`
  - The `chmod` command edits read, write, execute permissions for a file or directory. The three digit number in this case '400' represents the permissions for the users. The first digit is for the owner, the next one is for the group and the last digit is for anyone else. The value is dictated by the sum of the permission values; (4 = read), (2 = write), (1 = execute). So '400' means read permission to the owner, and no permissions to the group or anyone else. Finally "dsci2024.pem" is just the file whose permissions are being modified.
- `ssh -i "dsci2024.pem" https://ubuntu@ec2-54-183-13-46.us-west-1.compute.amazonaws.com`
  - SSH or the Secure Shell protocol is a network protocol that allows a secure connection between two systems. In the context of a command line the 'ssh' command invokes a connection from the user terminal to the address listed in the command. In this case the address is to amazonaws.com. The '-i' invokes a identity file from where we get access data, in this case the access file is a local file called 'dsci2024.pem' which we download off of AWS. Basically we are getting access to our ubuntu server on AWS by invoking the ssh command from our terminal and using an identity file called dsci2024.pem to invoke access to that ubuntu server
- `wget dlcnd.apache.org/spark/spark-3.5.0/spark-3.5.0-bin-hadoop3.tgz`
  - `wget` is a command that retrieves and downloads content from the address specified in the command. In this case the address is for a .tgz or zip file for Spark; off of apache's website.
- `tar xvf spark-3.5.0-bin-hadoop3.tgz`
  - The `tar` command can be used to compress a bunch of files, or be used to extract, maintain or modify files that were created with `tar`. The `tar` command's arguments further direct the actions of the `tar` command; In this instance 'xvf' stands for x-extract, v-display verbose output, f- specifies the file name of the archive. Note that the file in this instance is called spark-3.5.0-bin-hadoop3.tgz.

**Q2: [4 pts]** Please add the following lines to the end of `~/bashrc` file (i.e., `.bashrc` file under the home directory). Show a screenshot of the file with modified content [last few lines of the file]. Explain what each of the following commands does.

- `export JAVA_HOME=/usr/lib/jvm/default-java`
  - The `export` command is used to create an environment variable which is exported to all child process, hence why we include it in a folder such as `bash`. In this case the `export` command is used to invoke the default java library so it is available to use by all child processes in our AWS ubuntu server. The first part is a variable called `JAVA_HOME` and it is set to the default java library so that we can use java on our machine, we put this variable in our `bash` file so that it persists and we don't have to call it every time we boot our ubuntu server.
- `export PATH=$PATH:~/spark-3.5.0-bin-hadoop3/bin`
  - The `export` command is used in the same context as question 2a, the difference is that this time it is used to set a `PATH` variable, which is used to tell the `bash` shell where to find different executable files or scripts. In this case the path leads to the bin file located in home -> `spark-3.5.0-bin-hadoop3 -> bin`. The '`PATH=$PATH:`' simply defining the path in our `bash` file.

```

GNU nano 4.8 /home/ubuntu/.bashrc
# colored GCC warnings and errors
export GCC_COLORS="error=01:31;warning=01:35;note=01:36;caret=01:32;locus=01:01"
# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands. Use like so:
# sleep 10; alert
alias alert='notify-send --urgency=low -i "[ $? = 0 ]" && echo terminal || echo error)' "${history/tail -n1;sed -e '\$/\s*\([0-9]\+\s*//;/s/[:&])\s*alert$/\s*'"

# Alias definitions
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
. ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc)
if ! shopt -o posix; then
if [ -f /usr/share/bash-completion/bash_completion ]; then
. /usr/share/bash-completion/bash_completion
elif [ -f /etc/bash_completion ]; then
. /etc/bash_completion
fi
fi

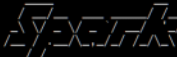
export JAVA_HOME=/usr/lib/jvm/default-java
export PATH=$PATH:~/spark-3.5.0-bin-hadoop3/bin

```

**Q3: [2 pts]** Install Spark by following the instructions in the handout (note the handout has been updated and make sure you use the latest version on d2l) and submit a screenshot showing the successful starting up of `pyspark`. [screenshot should include the prompt from "`pyspark`"]

```

root@ip-172-31-46-240:~#
spark-3.5.0-bin-hadoop3/bin/spark-sql.cmd
spark-3.5.0-bin-hadoop3/bin/sparkR
spark-3.5.0-bin-hadoop3/bin/spark-submit.cmd
spark-3.5.0-bin-hadoop3/bin/find-spark-home.cmd
spark-3.5.0-bin-hadoop3/bin/run-example.cmd
spark-3.5.0-bin-hadoop3/bin/spark-connect-shell
spark-3.5.0-bin-hadoop3/bin/spark-sql2.cmd
spark-3.5.0-bin-hadoop3/bin/spark-shell.cmd
spark-3.5.0-bin-hadoop3/bin/spark-class2.cmd
spark-3.5.0-bin-hadoop3/bin/spark-class
spark-3.5.0-bin-hadoop3/bin/spark-class.cmd
spark-3.5.0-bin-hadoop3/bin/spark-submit2.cmd
spark-3.5.0-bin-hadoop3/bin/spark-shell
spark-3.5.0-bin-hadoop3/bin/find-spark-home
ubuntu@ip-172-31-46-240:~$ pyspark
Python 3.8.10 (default, May 26 2023, 14:05:08)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
A[[[A[24/01/15 03:08:17 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
welcome to

 version 3.5.0

Using Python version 3.8.10 (default, May 26 2023 14:05:08)
Spark context Web UI available at http://ip-172-31-46-240.us-east-2.compute.internal:4040
Spark context available as 'sc' (master = local[*], app id = local-1705288099194).
SparkSession available as 'spark'.
>>>

```