Assignment-7 :Hemanth Reddy

TensorFlow is an open-source software library for machine learning and artificial intelligence that focuses on training and inference of deep neural networks. You can install the latest version of TensorFlow on your local machine by selecting any installed python3 or conda3 module and installing it in your home directory Turn in the path of the command/python script and the screenshot to display the version and location of the TensorFlow you have installed.

A:

To install tensorflow: pip3 install tensorflow

Below is the screenshot :

```
(torch_venv) hemanth@sh01:~$ pip show tensorflow
Name: tensorflow
Version: 2.11.0
Summary: TensorFlow is an open source machine learning framework for everyone.
Home-page: https://www.tensorflow.org/
Author: Google Inc.
Author-email: packages@tensorflow.org
License: Apache 2.0
Location: /home/hemanth/pip/torch_venv/lib/python3.9/site-packages
Requires: flatbuffers, six, tensorflow-io-gcs-filesystem, typing-extensions, setuptools, numpy, absl-py, tensorflow-estimator, google-pa
sta, protobuf, keras, libclang, grpcio, termcolor, h5py, wrapt, opt-einsum, gast, astunparse, packaging, tensorboard
Required-by:
(torch_venv) hemanth@sh01:~$
```

Re-deploy individual WS:

```
hemanth@sh01:~$ aws ec2 create-image --region us-east-1
--instance-id i-0f8ebb12c5c8d9c70 --name "csye-hemanth-
c04"--no-reboot
{
    "ImageId": "ami-0e14a3e4433783c68"
}
```

```
i-069615d271af8c158     t3.small        running 172.31.87.179   us-east-1b      csye-rajat-c04  ec2-34-238-252-250.compute-1.amazonaws.com
i-0c6276a8edcfff6c5     t3.small        running 172.31.80.100   us-east-1b      csye-hemanth-c04        ec2-3-88-10-75.compute-1.amazonaws.com
hemanth@sh01:~$
```

Remove the slurmctld.service and slurmdbd.service services using the following commands:

sudo systemctl stop slurmctld.service
sudo systemctl disable slurmctld.service
sudo systemctl stop slurmdbd.service
sudo systemctl disable slurmdbd.service

Set cron job to only start EC2 at your work period

Crontab -e

0 9-17 * * 1-5 aws ec2 start-instances --instance-ids i-0c6276a8edcfff6c5
0 18 * * 1-5 aws ec2 stop-instances --instance-ids i-0c6276a8edcfff6c5

```
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command

0 9-17 * * 1-5 aws ec2 start-instances --instance-ids  i-0c6276a8edcfff6c5
0 18 * * 1-5 aws ec2 stop-instances --instance-ids i-0c6276a8edcfff6c5
```

0 9 * * 1-5 ssh -i "~/CSYE7374Spring23.pem" ubuntu@ ec2-3-88-10-75.compute-1.amazonaws.com
0 18 * * 1-5 echo "exit" | ssh -i "~/CSYE7374Spring23.pem" ubuntu@ ec2-3-88-10-75.compute-1.amazonaws.com

```
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command

0 9 * * 1-5 ssh -i "~/CSYE7374Spring23.pem" ubuntu@ ec2-3-88-10-75.compute-1.amazonaws.com
0 18 * * 1-5 echo "exit" | ssh -i "~/CSYE7374Spring23.pem" ubuntu@ ec2-3-88-10-75.compute-1.amazonaws.com
```

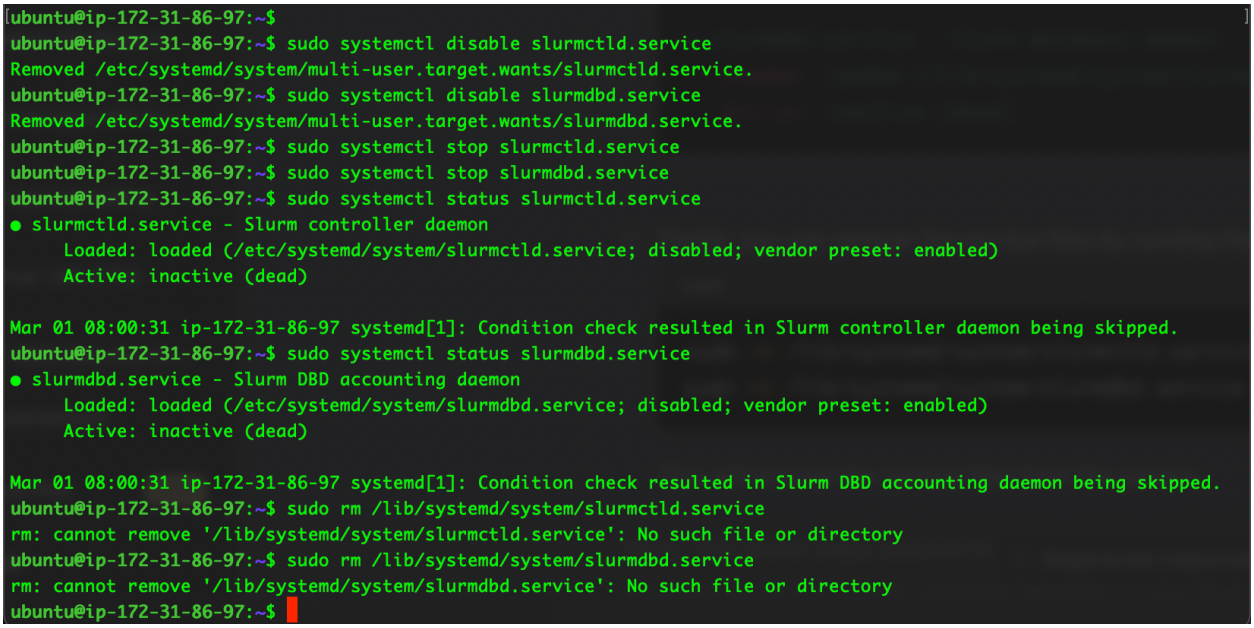
The patching script is a custom script that contains the necessary commands to patch the OS.

Step-1: nano patching_script.sh

```
#!/bin/bash

# Update the package lists
sudo apt-get update

# Upgrade the packages
sudo apt-get upgrade -y

# Clean up the package cache
sudo apt-get clean
sudo apt-get autoclean
sudo apt-get autoremove -y
```

Final step: 0 0 * * 0 ~/ **patching_script.sh (Added this after crontab -e)**

Re-configure team server (Team) o Remove OS services: slurmctld.service, slurmdbd.service o Remove MYSQL package (apt installed) o Set up cron job to patch the OS every week:

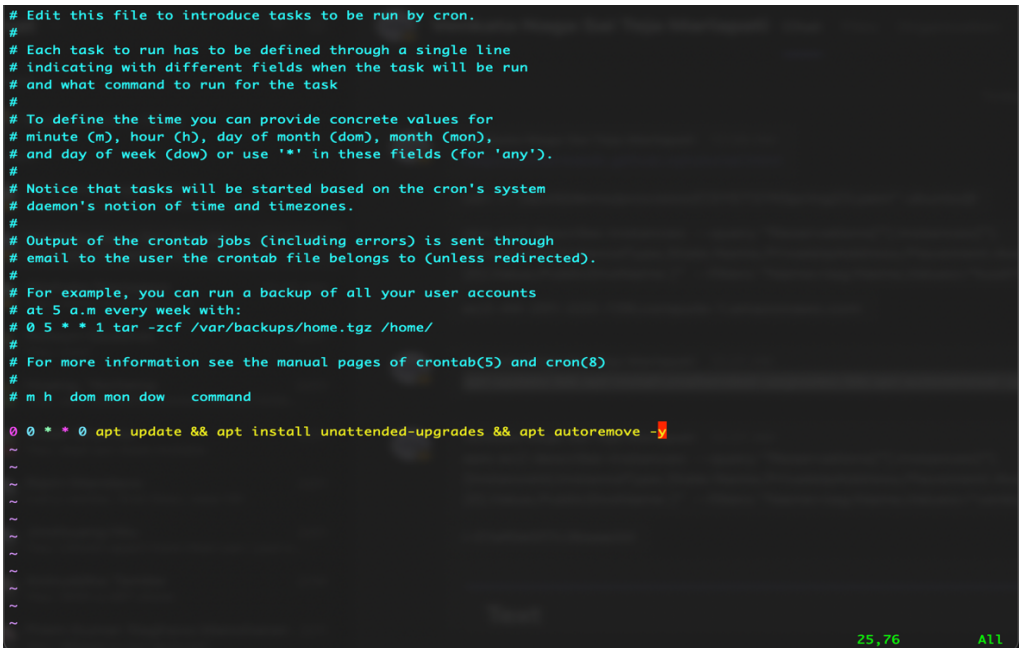


Remove the **MYSQL** package using the following command:

This command will remove the **mysql-server** package that was previously installed using **apt**.

```
ubuntu@ip-172-31-86-97:~$ mysql --version
-bash: /usr/bin/mysql: No such file or directory
ubuntu@ip-172-31-86-97:~$
```

Set up a cron job to patch the os everyweek


Edit this file to introduce tasks to be run by cron.
#
Each task to run has to be defined through a single line
indicating with different fields when the task will be run
and what command to run for the task
#
To define the time you can provide concrete values for
minute (m), hour (h), day of month (dom), month (mon),
and day of week (dow) or use '*' in these fields (for 'any').
#
Notice that tasks will be started based on the cron's system
daemon's notion of time and timezones.
#
Output of the crontab jobs (including errors) is sent through
email to the user the crontab file belongs to (unless redirected).
#
For example, you can run a backup of all your user accounts
at 5 a.m every week with:
0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
For more information see the manual pages of crontab(5) and cron(8)
#
m h dom mon dow   command
0 0 * * 0 apt update && apt install unattended-upgrades && apt autoremove -y
25,76 All