Solutions to the List of tasks

**Git commands:**

1. How do you set up a script to run every time a repository receives new commits through push?

Solution 1: We can use post-receive githooks

Solution 2: We could also use Jenkins or GitLab CI to monitor the repository and trigger scripts/actions once a commit is landed

2. How do you find a list of files that have changed in a particular commit?

Solution: git diff-tree -r --name-only --no-commit-id <COMMIT\_SHA>

Monitoring using scripts: (Shared Video Recording)

1. Monitor a log file, detect a pattern detection, send an email on detection

Setup a vagrant instance

Installed and configured sendmail

tail -f /tmp/testlog | while read line

do case "$line" in

\*"ERROR"\*) echo "$line" | mail -s "Error in Server Down" azherullahkhan@gmail.com;

;;

esac

done

2. Monitor process particular process on an instance, send an email in case of state change like process got stopped, taking more CPU that threshold

cat /tmp/check\_app.sh

#!/bin/sh

APP=sendmail

ACTIVE\_APP=$(ps -uax | grep -v grep | grep ${APP} | wc -l)

if [ $ACTIVE\_APP -eq 0 ]

then

echo "${APP} is NOT RUNNING" | mail -s "Error: ${APP} is Down" azherullahkhan@gmail.com

fi

crontab -e

\*/5 \* \* \* \* /tmp/check\_app.sh

Checking CPU usage of a particular process

ps -eo %cpu,cmd | grep -v grep | grep nginx

**Docker: (Shared Video Recording)**

1. Create a sample docker container with a Node.js Express app and demonstrate the installation.

Guidelines :

● You should be able to find what system packages are needed by looking through the app

● You should not need to change the app code in any way

● The app should be running as a non-privileged user

● The app should be automatically restarted if crashes or is killed

● The app should maximize all of the available CPUs

● Timezone should be in IST

● Follow best practices when writing a dockerfile

$ cat Dockerfile

FROM node:16

# Create app directory

WORKDIR /usr/src/app

# A wildcard is used to ensure both package.json AND package-lock.json are copied

# where available (npm@5+)

COPY package\*.json ./

RUN npm install

# Bundle app source

COPY . .

USER node

EXPOSE 8080

CMD [ "node", "server.js" ]

$ cat server.js

'use strict';

const express = require('express');

// Constants

const PORT = 8080;

const HOST = '0.0.0.0';

// App

const app = express();

app.get('/', (req, res) => {

res.send('Welcome from Node JS running Docker on MacBookPro');

});

app.listen(PORT, HOST);

console.log(`Running on http://${HOST}:${PORT}`);

$ cat package.json

{

"name": "docker\_web\_app",

"version": "1.0.0",

"description": "Node.js on Docker",

"author": "Azher Khan",

"main": "server.js",

"scripts": {

"start": "node server.js"

},

"dependencies": {

"express": "^4.16.1"

}

}

$ docker build . -t ak/node-web-app

$ docker run -u node -m "300M" --memory-swap "1G" -p 49161:8080 -d ak/node-web-app

**Security: (Shared Video Recording)**

1. Show how to block ports

iptables -L

iptables -A INPUT -p tcp -s 0/0 -d 0/0 --dport 80 -j DROP

2. Show how to setup port forwarding

iptables -L -n -v

iptables -t nat -A OUTPUT -o lo -p tcp --dport 80 -j REDIRECT --to-port 8088

iptables -t nat -I PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080

iptables -I INPUT -p tcp --dport 8080 -j ACCEPT

iptables -L -n -v

**Network: (Shared Video Recording)**

1. Show list of processes using the network.

iptraf-ng

2. Show the list of IPs a process is connected to

netstat -n

3. Show how to list open files and kill processes tied to a user

lsof -u nginx

**Code:**

Write a short program that prints each number from 1 to 100 on a new line.

For each multiple of 3, print "AVA" instead of the number.

For each multiple of 5, print "AMO" instead of the number.

For numbers which are multiples of

!#/bin/bash

i=1

while [ $i -le 100 ]

do

if [ `expr $i % 3` -eq 0 ]; then

#echo $i

echo "----"

echo "AVA"

elif [ `expr $i % 5` -eq 0 ]; then

#echo $i

echo "----"

echo "AMO"

else

echo "----"

echo "$i"

fi

i=$(($i+1))

done