**1: Write a python script that can communicate with the rest API found at http://jsonplaceholder.typicode.com/posts**

The script should be able to do the following using command line argument:

- read a post by id and display the title and body on separate lines.

- create a post where the title and body are command line arguments

- update the title and body of a post by post id

#####################################################################################################

**2: Using either Chef, Puppet or Saltstack, compose the necessary syntax in a single file that will do the following:**

note: assume you are running on Debian/Ubuntu

- Install Apache2

- Rename the "default" vhost to "my\_website"

- Create a directory called /opt/scripts

- Puts an empty file in the scripts directory called "holding.sh" which is executable by all users

**Answer:**

Have used shell script to accomplish the above task as I don’t have hands on experience in above CM tools.

Am writing this script for Red Hat Enterprise Linux Server release 6.6 and assume “yum” (software installation repo) is configured already

**Installation.sh**

#!/bin/bash

echo "Installing Apache Server"

yum -y install httpd

echo "Rename the "default" vhost to "my\_website""

sed -i "s|VirtualHost \_default\_:80|VirtualHost my\_website:80|g" /etc/httpd/conf/httpd.conf

echo "Creating a directory called /opt/scripts"

mkdir -p /opt/scripts

echo "Puts an empty file in the scripts directory called "holding.sh" which is executable by all users"

touch /opt/scripts/holding.sh

chmod +x /opt/scripts/holding.sh

#####################################################################################################

**3: Put the python script and the configuration management files in a Github public repository for review.**

Assumptions:

1. Below is the Github public repository

<https://github.com/balachandarms/auto_cm.git>

1. Cloned the above repo to local system

git clone <https://github.com/balachandarms/auto_cm.git>

Add and commit the files to above repo

1. git add installation.sh
2. git commit installation.sh –m “[balachandarms][Infra] Commit installation file”

#####################################################################################################

**4: Also in the repository place a text file called responses.txt containing answers to the following questions:**

\* When might you use sed, awk and cut together?

\* What is your favourite text editor and why?

\* What steps would you take to configure a server to use Apache and Tomcat?

\* Users begin complaining about "slowness" on your web application. What steps would you take to investigate this?

**Answers: responses.txt**

1. Based the scenario will use sed, awk and cut commands. We can use these commands together using "|" symbol

cut : this command mostly will use when we know the exact delimiter

sed : this command mostly will use to do search and replace the patterns in files without opening and same can be done using awk also

awk : this is a powerful command having syntax like C. which will be used to processing the files which has data arranged in rows and columns format.

2. vi is my favourite text editor and below are the reasons why am using it

1. more simpler to use
2. we can find this in all the Unix systems by default
3. it has two modes (command and insert)
4. it’s a simple program that does not require more system memory and CPU usage.

3. What steps would you take to configure a server to use Apache and Tomcat?

Assuming we want to configure java based application server using Apache and tomcat in linux

1. Install Apache and update the server domain name in apache conf files

2. Install jdk/jre to be used by tomcat server

3. Install Tomcat Server and configure the ports in which it has to run

4. Deploy java based application in Tomcat webapps directory

5. Add proxy & reverse proxy pass in Apache conf file for java application deployed in tomcat

ProxyPass / http://10.20.100.101:8080/java\_application/

ProxyPassReverse / http://10.20.100.101:8080/java\_application/

4. \* Users begin complaining about "slowness" on your web application. What steps would you take to investigate this?

First we need to isolate whether the problem is at web application or on the network

If the slowness is reported by all the users then it could be issue with application side.

If the slowness is reported by only some set of users and for other if it’s working fine then it could users network issue.

We have to check below things at web application end to find out the slowness

We will check the web application logs for any Errors

We will check is there any issue with Database connectivity if web application is using Database services

We will check is there any issue with middleware applications if web application is using any middleware applications

We will check is there any issue with Linux server where the web application is hosted (Memory, Disk Space, I/O and CPU stats)

We can use developer tools in the browsers (like chrome/firefox etc.) and check which call is taking more time (can also check at file level).

We can use few monitoring tools like **AppDynamics** where we have an integrated view of all the systems that are in the application and can spot the slowness.

Add and commit the responses.txt to above repo

1. git add responses.txt
2. git commit responses.txt –m ‘[balachandarms][Infra] Commit responses.txt file’

#####################################################################################################