Bash Script Source Code:

#!/bin/bash

# Set your Linode API access token

ACCESS\_TOKEN=your\_linode\_api\_token

# Set deployment parameters

region="ap-south" # Replace with your desired region

image="linode/ubuntu22.04" # Replace with your desired Ubuntu image

label="ProjectTestCode" # Replace with your desired label

root\_pass="YourRootPassword,./" # Replace with your desired root password

type="g6-dedicated-2" # Replace with your desired instance type

desired\_cpu\_version="EPYC 7713" # Replace with your desired CPU version

# Provision 50 instances with desired CPU model

counter=1

while [[ $counter -lt 51 ]];

do

echo "Provisioning instance $counter..."

# Create the Linode instance

create\_instance\_response=$(curl -H "Content-Type: application/json" \

-H "Authorization: Bearer $ACCESS\_TOKEN" \

-X POST -d '{

"label": "'"$label$counter"'",

"image": "'"$image"'",

"type": "'"$type"'",

"region": "'"$region"'",

"root\_pass": "'"$root\_pass"'",

"booted": true

}' \

"https://api.linode.com/v4/linode/instances")

# Extract the Linode instance ID from the response

instance\_id=$(echo "$create\_instance\_response" | jq -r '.id')

# Wait for the Linode instance to be running

while true; do

instance\_status\_response=$(curl -s -H "Authorization: Bearer $ACCESS\_TOKEN" "https://api.linode.com/v4/linode/instances/$instance\_id")

instance\_status=$(echo "$instance\_status\_response" | jq -r '.status')

if [[ "$instance\_status" == "running" ]]; then

break

fi

sleep 150

done

# Get the Linode instance IP address

instance\_ip=$(echo "$instance\_status\_response" | jq -r '.ipv4[0]')

# SSH into the Linode instance and check the CPU version

cpu\_version=$(sshpass -p "$root\_pass" ssh -o StrictHostKeyChecking=no root@"$instance\_ip" "lscpu | grep 'Model name:' | awk -F': ' '{print \$2}'")

# Compare the CPU version

if [[ "$cpu\_version" == \*"$desired\_cpu\_version"\* ]]; then

echo "Ubuntu server on Linode $instance\_id has the desired CPU version: $desired\_cpu\_version"

$counter+=$(counter+1)

else

echo "Ubuntu server on Linode $instance\_id does not have the desired CPU version. Current CPU version: $cpu\_version"

# Delete the Linode instance

delete\_instance\_response=$(curl -X DELETE \

-H "Authorization: Bearer $ACCESS\_TOKEN" \

"https://api.linode.com/v4/linode/instances/$instance\_id")

echo "Linode $instance\_id deleted."

fi

done

Python Source Code

import requests

import time

import os

# Set your Linode API access token

ACCESS\_TOKEN = "your\_linode\_api\_token"

# Set deployment parameters

region = "us-iad" # Replace with your desired region

image = "linode/ubuntu22.04" # Replace with your desired Ubuntu image

label = "TestCodeInstance" # Replace with your desired label

root\_pass = "YourRootPassword,./" # Replace with your desired root password

type = "g6-dedicated-2" # Replace with your desired instance type

desired\_cpu\_version = "EPYC 7713" # Replace with your desired CPU version

# Provision 50 instances with desired CPU model

counter = 1 #initial counter

while counter <= 50: #condition for 50 instances to be provisioned

print(f"Provisioning instance {counter}...")

# Create the Linode instance

create\_instance\_payload = {

"label": f"{label}{counter}",

"image": image,

"type": type,

"region": region,

"root\_pass": root\_pass,

"booted": True

}

create\_instance\_headers = {

"Content-Type": "application/json",

"Authorization": f"Bearer {ACCESS\_TOKEN}"

}

create\_instance\_response = requests.post(

"https://api.linode.com/v4/linode/instances",

json=create\_instance\_payload,

headers=create\_instance\_headers

)

# Extract the Linode instance ID from the response

instance\_id = create\_instance\_response.json().get("id")

# Wait for the Linode instance to be running

while True:

instance\_status\_response = requests.get(

f"https://api.linode.com/v4/linode/instances/{instance\_id}",

headers=create\_instance\_headers

)

instance\_status = instance\_status\_response.json().get("status")

if instance\_status == "running":

break

#let the linode instance done all the initial setup

time.sleep(150)

# Get the Linode instance IP address

instance\_ip = instance\_status\_response.json().get("ipv4")[0]

# SSH into the Linode instance and check the CPU version

ssh\_command = f"sshpass -p '{root\_pass}' ssh -o StrictHostKeyChecking=no root@{instance\_ip} lscpu"

cpu\_version = os.popen(ssh\_command).read()

# Compare the CPU version

if desired\_cpu\_version in cpu\_version:

print(f"Ubuntu server on Linode {instance\_id} has the desired CPU version: {desired\_cpu\_version}")

counter += 1

else:

print(f"Ubuntu server on Linode {instance\_id} does not have the desired CPU version. Current CPU version: {cpu\_version}")

print(f"Deleting Linode {instance\_id}...")

delete\_instance\_response = requests.delete(

f"https://api.linode.com/v4/linode/instances/{instance\_id}",

headers=create\_instance\_headers

)