**M Assessment**

**Q1). Using configuration management tool Ansible create multiple users and confirm on your manage host.**

Solution =>

Ansible is a configuration management tool used to configure on OS on an existing infrastructure.

I am going to use Ansible loop for this purpose

To achieve this I am going to launch an Redhat instance(Best works for ansible) and given a hostname ansible-controller

And two remote hosts named it ans-one and ans-two

A screenshot of a computer program

AI-generated content may be incorrect.

This is for instance ans-one

A screenshot of a computer

AI-generated content may be incorrect.

And this is for instance ans-two

A screenshot of a computer

AI-generated content may be incorrect.

Now to make a communication between ansible controller and remote hosts ssh must be configure so generated a ssh key and pasted into authorized\_keys into remote hosts

Since ansible is an agentless so works with ssh and python should be there .

A screenshot of a computer

AI-generated content may be incorrect.

Once we are done with ssh we will make a ansible.cfg (configuration file) for ansible

A screenshot of a computer

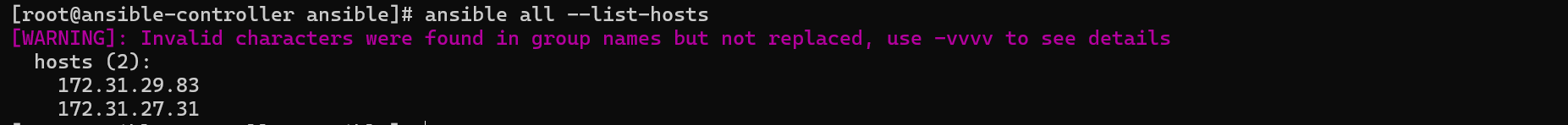
AI-generated content may be incorrect.

Putting my two remote servers into hosts file

A screenshot of a computer

AI-generated content may be incorrect.

As I can see that controller is all set with the remote hosts



Creating a playbook named user.yaml which will create multiple users into remote hosts

A screenshot of a computer

AI-generated content may be incorrect.

To check the syntax used this command since yaml follow indentation

A screenshot of a computer

AI-generated content may be incorrect.

After checking syntax executed this playbook and can see the yellow coloured text that represent that it worked.

A screenshot of a computer

AI-generated content may be incorrect.

Now checking those users into the remote hosts

First checking into ans-one and all the users are there

A computer screen with white text

AI-generated content may be incorrect.

And all the users are also int ans-two

A screen shot of a computer

AI-generated content may be incorrect.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Q2). We have an EC2 instance in N. Virginia Region where a web server is running. create custom image of this server and launch new server in Ohio region using of this image**

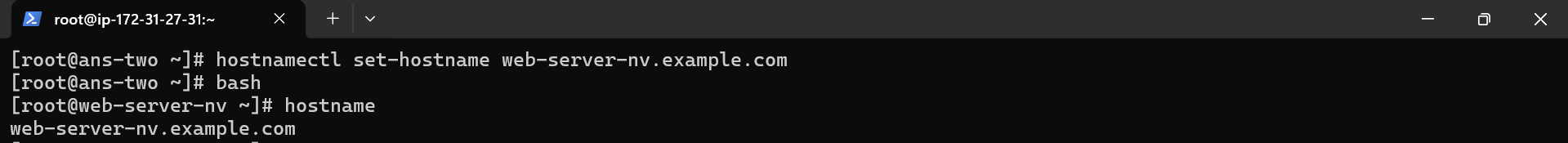
Solution =>

Launching an instance into N.virigia Region

A screenshot of a computer

AI-generated content may be incorrect.

Setting hostname for this instance



Installed httpd using => yum install httpd -y

And after installation created an index.html file which will act as an application

So started and enabled this service and can see that my application is reachable with the ip



Since it is running into n.virginia and I want to use this into ohio so for that purpose I am creating an image from this instance.

So before creating image it is recommended to stop before image creation

A screenshot of a computer

AI-generated content may be incorrect.

Copying this image into ohio region

A screenshot of a computer

AI-generated content may be incorrect.

Now I am able to that image into ohio region

A screenshot of a computer

AI-generated content may be incorrect.

So now launched an instance with that image and named it server-ohio

A screenshot of a computer

AI-generated content may be incorrect.

After giving name to the host we are able to access the same application without download or starting and enabling it .

A screenshot of a computer

AI-generated content may be incorrect.

**Q3). Pull the Ubuntu image from Docker hub and Launch a web application in the container on port no. 8080 and this application should be reachable globally**

Solution =>

Launched an instance and set the hostname to docker.example.com.

This instance will act as a docker host

A screenshot of a computer

AI-generated content may be incorrect.

Installed docker into this machine and started and enabled it.

After that pulled ubuntu image from docker hub

A screenshot of a computer

AI-generated content may be incorrect.

I used this command to see all the available images into this docker host

And using that image launched a container named my-app and also enabled port forwarding so that it will also be accessible globally using 8080 port (I also enabled this port into inbound security group). And assigning it a shell where we will run commands.

A screenshot of a computer

AI-generated content may be incorrect.

After updating ubuntu installed apache2 and creating an index.html file and given some content.



After that started and enabled apache2 and try to access inside docker host with the ip of the container 172.17.0.2

And can see that web page is accessible so now will try to access it globally

A screenshot of a computer

AI-generated content may be incorrect.

Now accessing with the help of public ip of that machine with port no 8080 and can see that page is accessible globally.

A screenshot of a computer

AI-generated content may be incorrect.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Q4). Deploy a web application in the kubernetes pod. And create a replica set. In any case load is going to increase on your replica set. increase the number of replica of the pods**

Solution =>

To deploy a web application (will use nginx ) in Kubernetes first I will create a cluster

One Control plane and one worker node.

A screenshot of a computer

AI-generated content may be incorrect.

Created a script file(k8s.sh) for creating the control plane

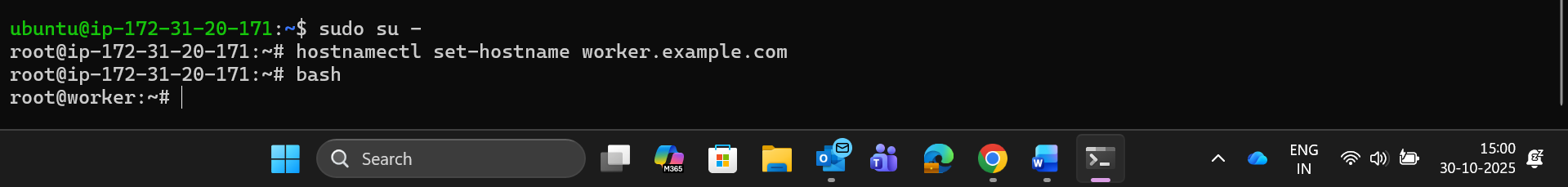
A screenshot of a computer screen

AI-generated content may be incorrect.

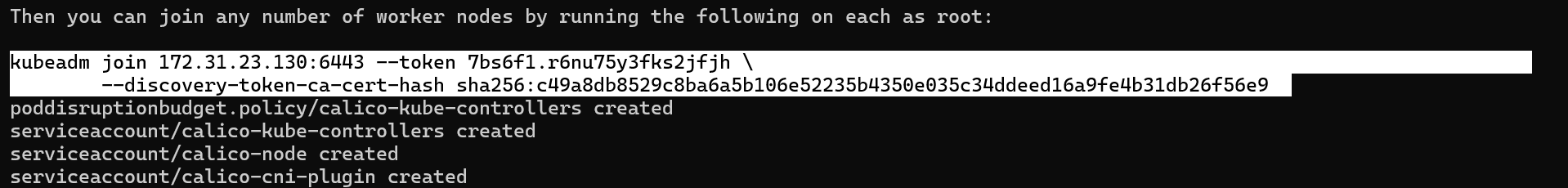
Assigned permission to execute this script



After setting of control plane hosted one more server and named it worker



To connect worker node with the control node we will require this kubeadm token so will save it for further usecase



Also created a script(worker.sh) and assigned permission to execute

A screenshot of a computer

AI-generated content may be incorrect.

Paste that copied token into this script

A screenshot of a computer screen

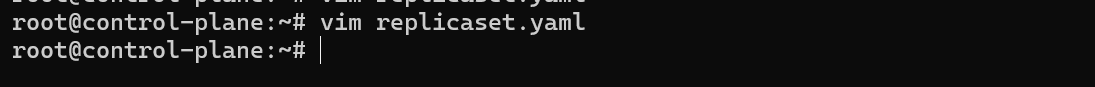
AI-generated content may be incorrect.

Now can see our control and worker machine is ready

A screenshot of a computer

AI-generated content may be incorrect.

So Now creating a replicaset with the help of manifest file which is a yaml (yet another markup language).



Can see apiversion and kind for the replicaset with the help of command

Kubectl api-resources

From these information I have puted the kind and version

And the name of the replicaset is nginx-replicaset and also given it a label so that in future it can be used to group the resources that have the same label like app:web

And replica is one which means it will create only one replica of a pod and total will be one

Selector label is used for the purpose on which pod it will going to apply replication controller.

And finally template is for pod. I am using nginx image for my web application .

It will be accessible with the port 80.

A screenshot of a computer

AI-generated content may be incorrect.

After applying with the command line tool kubectl for accessing k8s cluster

Can see the pod where which will manage the container and container will manage the application

A screenshot of a computer

AI-generated content may be incorrect.

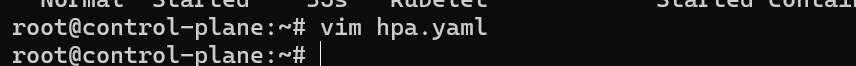
Can see more details about this pod where it is deployed and on which port and which image is being used and controlled by which controller.

A computer screen with white text

AI-generated content may be incorrect.

Initially only one pod was running so will try to scale this with the controller hpa (horizontal pod autoscaling).

So for that purpose created one more manifest file and named it hpa.yaml



Can see the api version and Kind of controller and target on which it will going to apply hpa so which is nginx-replicaset .

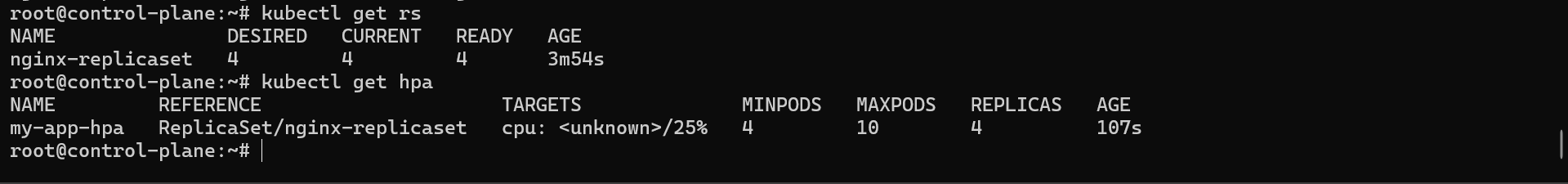
And will scale pod on the behalf of cpu utilization.

So flow is like our hpa will manage the replicaset and replicaset will manage the pods

A screenshot of a computer

AI-generated content may be incorrect.

After applying this hpa we can see our replicaset(rs) and hpa.



Now initially only one pod was there but after some seconds can see it scaled to 4.

A screenshot of a computer

AI-generated content may be incorrect.

To see the more details about this newly created pods used this command.

A screenshot of a computer program

AI-generated content may be incorrect.

Now upon deletion of pod it will again create the new pod.

But in production it is not recommended to directly control pod with replicaset controller because upon deletion of replicaset no pod will be created. It will be gone forever. So will use Deployment.

THANK YOU

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*End\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*