BRICS
2022 CHINA

Test Project (Offline)

BRICS-FS-11_Cloud Computing

2022 BRICS Skills Competition

Session A: The Deployment and O&M of OpenStack Platform

Business scene:

An enterprise plans to use openstack to build an enterprise cloud platform for deploying various enterprise applications for external and internal services. The cloud platform can realize IT resource pooling, flexible allocation, centralized management, performance optimization, and unified security certification. The system structure is as follows:

The construction of the enterprise cloud platform uses two cloud servers provided by the competition platform, and the configuration is as follows:

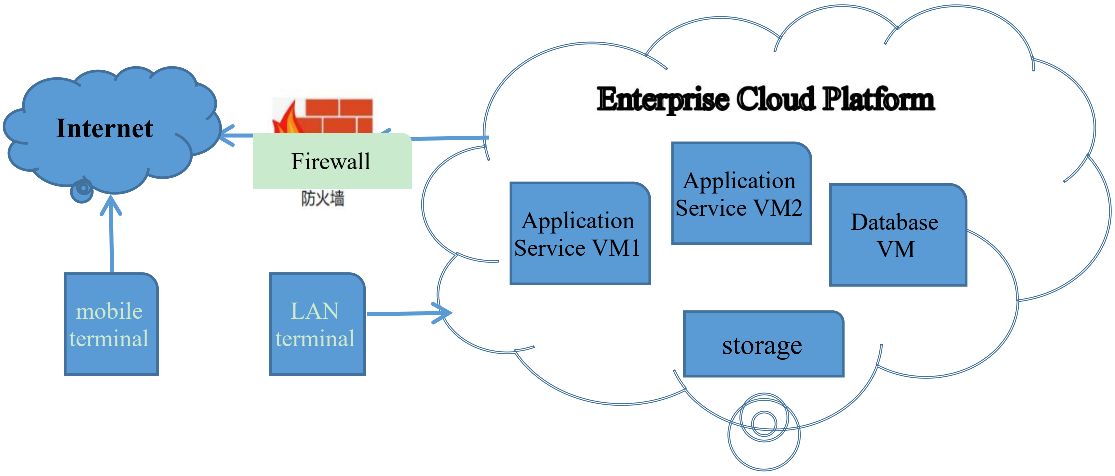


Table 1 IP address planning

Device Name	Host Name	Interface	IP Address	Remark
Cloud host 1	Controller	eth0	172.17.x.10/24	Vlan 1x0
		eth1	172.18.x.10/24	Vlan 2x0
Cloud host 2	Compute	eth0	172.17.x.20/24	Vlan 1x0
		eth1	172.18.x.20/24	Vlan 2x0

Illustrate:

1. Contestants should check by themselves whether the hardware and network of the workstation PC are normal;

2. The competition is conducted in a cluster mode, and the competition platform provides each team with an independent tenant and user and two cloud hosts with the same resource quotas. Contestants log in to the competition cloud platform with their username and password, and build the OpenStack enterprise private cloud by using the two allocated cloud hosts;

3. Contestants should check whether the two cloud hosts and network configurations provided by the competition platform are normal;

4. The x in the table is the exam number;

5. The account resources, competition resource packages and accessories and their storage locations required for the exam will be given in the test seat information table and equipment confirmation sheet;

6. In order to ensure the security of the server, please modify the server password by yourself during the competition. Please confirm your IP address, user name and password when submitting information in the examination system.

Task 1 Initialization of environment on private cloud platform (5 points)

1．Initialize the operating system 【2 points】

(1) Use the provided username and password to log in to the competition cloud platform;

(2) According to the IP address planning in Table 1， set the host name of the control node as Controller, set the host name of the computing node as Compute, and modify the hosts file to map the IP address to the host name;

(3) Set the IP address of each server node to ensure normal network communication;

(4) Close Firewall and set it to not start to boot, set SELinux to Permissive mode and to be permanently closed;

(5) Check the host name of the control node and computing node, and use commands to check the SELinux status;

(6) Extract the permanent shutdown of SELinux Key Information by using the head command, tail command and cut command;

(7) Submit the above commands and return results to the answer box.

2．Mount the installation CD image 【1 point】

(1) Upload the provided CentOS-7-x86_64-DVD-1804.iso and chinaskills_cloud_iaas.iso disc images to the /root directory of the Controller node;

(2) Create the /centos and the /iaas directory in the /opt directory by using commands;

(3) Mount the image file CentOS-7-x86_64-DVD-1804.iso to the /centos directory, and mount the image file chinaskills_cloud_iaas.iso to the /iaas directory;

(4) Submit the above commands and return results to the answer box.

3．Set up a file sharing server 【0.5 points】

Please install the vsftp server on the Controller node and set it to start automatically at boot. Submit the above command and the returned result to the answer box.

4．Set the yum source 【0.5 points】

(1) Set the ftp repository to /opt/;

(2) Set the local yum source named local.repo for the controller node;

(3) Configure the ftp source named ftp.repo for the compute, and the ftp server address is the IP of controller node;

(4) Please submit the contents of the yum source files of the two nodes to the answer box.

5．Deploy time synchronization servers 【1 point】

(1) Deploy the chrony server on the Controller node, allow other nodes to synchronize time, start the service and set it to start at boot;

(2) Specify the controller node as the upstream NTP server on the compute node, restart the service and set it to start at boot;

(3) Please use the chronyc command to synchronize the system time of the control node.

Task 2 Task of OpenStack build (10 points)

1. Modify variable file 【0.5 points】

(1) Install the iaas-xiandian software package on the control node and computing node respectively, and modify the basic variables in the configuration script file (the configuration script file is

/etc/xiandian/openrc.sh);

(2) make the variable file effective by using commands after the modification is completed, and then execute echo $INTERFACE_IP;

(3) Please submit commands and return results to the answer box.

2. Build database components 【1 point】

(1) Fill the script with the provided scripting framework iaas-install-mysql.sh, install services such as mariadb, mencached, rabbitmq, etc. on the controller node and complete the related configuration;

(2) Modify the maximum number of connections for mencached in the configuration file to 2048 after completion;

(3) Please submit the revised configuration file to the answer box.

3. Build the authentication service component 【1 point】

(1) Fill the script with the provided scripting framework iaas-install-keystone.sh, install the keystone service on the controller node and complete the related configuration;

(2) Use the openstack command to request a token after completion;

(3) Please submit the above commands and return results to the answer box.

4. Build image service components 【1 point】

(1) Fill the script with the provided scripting framework iaas-install-glance.sh, install glance service on the controller node and complete the relevant configuration;

(2) Upload cirros-0.3.4-x86_64-disk.img to the /root directory of the control node after completion;

(3) Upload the image to the openstack platform using the openstack command and name the image cirros;

(4) Please submit the operation commands and return results uploaded by the image to the answer box.

5. Build computing service components 【1.5 points】

(1) Fill the script with the provided scripting framework iaas-install-nova-controller.sh and iaas-install-nova-compute.sh, install the nova service on the controller and compute nodes and complete the configuration;

(2) Add the computing resources of the control node to the cluster after completion;

(3) List the nodes that can provide computing resources by using the openstack command;

(4) Submit commands that list computing resources and return results to the answer box.

6. Build network components and initialize the network 【4 points】

(1) Fill the script with the provided scripting framework iaas-install-neutron-controller.sh and iaas-install-neutron-compute.sh, install and configure the neutron service on the controller and compute nodes;

(2) Create the external network(ext-net) of the cloud host, the subnet is ext-subnet, the available network segment of the floating IP of the cloud host is 172.18.x.100~ 172.18.x.200, and the gateway is 172.18.x.1;

(3) Create the cloud host internal network(int-net1), the subnet is int-subnet1, the cloud host subnet IP available network segment is 10.0.0.100 ~ 10.0.0.200, the gateway is 10.0.0.1;

(4) Create the cloud host internal network(int-net2), the subnet is int-subnet2, the available network segment of the cloud host subnet IP is 10.0.1.100 ~ 10.0.1.200, and the gateway is 10.0.1.1;

(5) Add a router named ext-router, add a gateway on the ext-net network, add an internal port to the int-net1 network, and complete the connection between the int-net1 and the ext-net;

(6) Use the openstack command to complete the above tasks, and submit commands and return results to the answer box after completion.

7. Build a graphical interface 【1 point】

Please fill the script with the provided scripting framework iaas-install-dashboard.sh, install the dashboard service on the controller node and complete the relevant configuration. Please obtain the dashboard homepage information through the curl command, and submit the obtained homepage information to the answer box.

Task 3 OpenStack operations tasks (15 points)

1. Manage users【1 point】

(1) Create user named testuser in keystone with password as password;

(2) Change the password of testuser to 000000 and view the details of testuser by using commands after creation;

(3) Add the user to the admin project and give ordinary user permissions to test the login after completion;

(4) Please take a screenshot and submit it to the answer box after completing the login system with testuser.

2. Query services 【0.5 points】

List service catalogs and endpoints and view endpoints for glance services with the command. Submit the above commands and return results to the answer box.

3. Manage images 【2 points】

(1) Log in to the controller node, run the glance command, upload the image CentOS_6.5_x86_64_XD.qcow2 and name it testone;

(2) Use the Openstack command to change the image name to examimage;

(3) Give the image a tag named lastone, and use the Openstack command to view the image list after the tag is changed;

(4) Submit the above commands and return results to the answer box.

4. Manage backend configuration files 【0.5 points】

(1) Go to the glance backend storage directory and view any image information by using the qemu command;

(2) View the size of the nova main configuration file by using the du command;

(3) Submit the above commands and return results to the answer box.

5. Manage storage services 【2 points】

(1) Create a cloud disk with the volume type ID after creating a volume type, and query the detailed information of the cloud disk;

(2) Mount the cloud disk to the virtual machine and format the cloud disk as xfs;

(3) Create a file with the name of the station number and the content as the station number, and then uninstall the cloud hard disk;

(4) Make the cloud disk read-only by running the Openstack command, and mount it again to check whether the original file exists;

(5) Create a file in the cloud disk again, and set the file name to station number_02;

(6) Submit the returned results and the problem solving process to the answer box.

6. Manage storage services 【2 points】

(1) create a 5GB cloud disk named disk-2 by using commands;

(2) Mount the cloud hard disk to the /mnt/ directory in the cloud virtual machine, format it as ext4, and submit commands and return information to the answer box by execute df -h;

(3) Uninstall the cloud disk and expand the cloud disk to 10GB by using commands;

(4) Mount the cloud disk to the cloud host by using commands, and submit commands and return information to the answer box;

(5) Expand the file system on the cloud host by using commands, mount it to /mnt/ directory again after the expansion, and submit commands and return information to the answer box after execute the df -hT command.

7. Manage object storage 【0.5 points】

(1) Create a container by using swift related commands, and then upload a file to the container (the file can be created by yourself);

(2) View the container by using commands after uploading;

(3) Submit the above commands and return results to the answer box.

8. Manage security groups 【2 points】

(1) Create a security group with the name group_web and the description as the station number by using commands;

(2) Add a rule to the security group to allow any ip address to access web traffic, and view the details of the security group when done;

(3) Submit the above commands and return results to the answer box.

9. Manage network 【0.5 points】

Please set the int-net1 network as shared by using commands, then view the details of the int-net1 network. Submit commands and return information to the answer box.

10. Manage network 【1 point】

(1) Use the centos7.5 image in the dashboard interface to create a cloud host named test-01;

(2) View the floating IP address pool and create a floating IP by using commands, and then bind the floating IP to the cloud host;

(3) Submit the above commands and return results to the answer box.

11. Manage virtual machines 【3 points】

(1) Use the centos 7.5 image to create a cloud host named test-02 through the opentack command and connect to the int-net1 network;

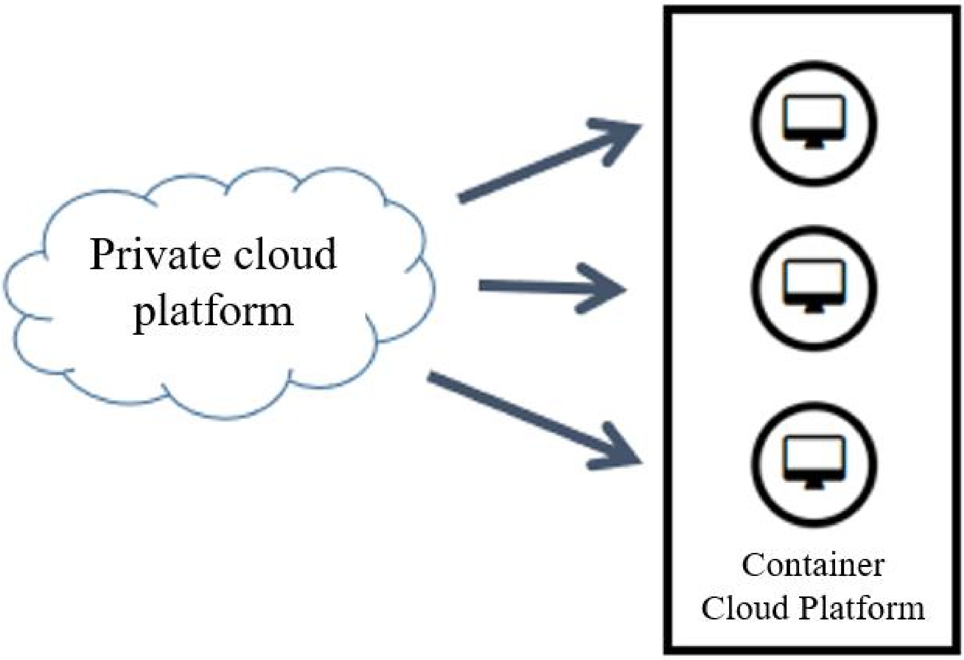
(2) View the detailed information of the cloud host by using commands after the creation is successful, and determine whether the cloud host is a computing node or a control node;

(3) Cold migrate it to the compute node if the cloud host is on the control node;

(4) Cold migrate it to the control node if the cloud host is on a compute node;

(5) Please complete the entire process of this question by using commands. Please paste all commands and results into the answer box.

Session B: Orchestration and O&M of Containers



In order to complete the company's internal daily work management, such as OA management, user management, customer relationship management and other businesses, a company plans to use the k8s platform to build a RuoYi background management system.

Table 1 Planning of IP addresses

Device Name	Host Name	Interface	IP	Remark
Cloud Server 1	Master	eth0	10.0.0.x/24	
Cloud Server 2	Node1	eth0	10.0.0.x/24	
Cloud Server 3	Node2	eth0	10.0.0.x/24	
Cloud Server 4	Harbor	eth0	10.0.0.x/24	

Illustrate:

1. The hosts required for Session B are all pulled from the private cloud platform of Session A;
2. The x in the table is a random IP;
3. According to the information given in the table, check the hardware connection and network configuration to ensure that the network is normal;
4. The resource package and accessories required for the exam are given in the information sheet of the examination seat;
5. During the competition, please do not modify the password of the server system, otherwise, the competition system may not be able to detect the score normally, and the responsibility shall be borne by the participants.

Task 1 Environment initialization of container cloud platform (5 points)

1. Initialize the container cloud platform 【1 point】

(1) Create a cloud server by using the image named CentOS_7.5_x86_64_XD.qcow according to the IP address plan in Table 1;

(2) Ensure that the network is normally connected, set the host name of node according to Table 1;

(3) Close swap, close selinux and firewall permanently, and modify the hosts mapping;

(4) Please submit the contents of the master node hosts file to the answer box.

2. Mount Yum source data persistently 【1 point】

(1) Upload the CentOS-7-x86_64-DVD-1804.iso and chinaskills_cloud_paas.iso to the /root directory of the master node;

(2) Create /centos directory and /paas directory in the /opt directory by using commands;

(3) Mount CentOS-7-x86_64-DVD-1804.iso to the /centos directory, and mount chinaskills_cloud_paas.iso to the /paas directory;

(4) Please set the auto-mount on permanent boot, and submit the content of the files to be automatically mounted on permanent boot to the answer box.

3. Write the Yum source 【1 point】

(1) Set up a local yum source for the master node, the yum source file name is centos.repo;

(2) Install the ftp service and set the root directory of the ftp repository to /opt/;

(3) Configure the ftp source for other nodes, the yum source file name is ftp.repo，the ftp server address is the IP of the master node;

(4) Please submit the contents of the yum source files of other nodes to the answer box.

4．Set time synchronization server 【1 point】

(1) Deploy the chrony server on the master node and allow other nodes to synchronize time;

(2) Start the service and set it to start at boot;

(3) Specify the master node as the upstream NTP server on other nodes, restart the service and set it to start at boot;

(4) Please use the chronyc command on the master node to synchronize the system time of the control node.

5. Set password-free login 【1 point】

(1) Set password-free login for four servers to ensure that the three servers can log in to each other without password;

(2) Please use the scp command to send the hosts file of the master node to the /etc/hosts directory of all nodes;

(3) Submit all the above commands and return results to the answer box.

Task 2 The construction of k8s (10 points)

1. Install the docker application 【1 point】

(1) Install docker-ce on all nodes;

(2) Install the harbor repository on the harbor node, show the normal login to the horbor repository;

(3) Set the login password to the format of "test_station number";

(4) Please submit the screenshot after login to the answer box.

2. Build harbor repository 【2 points】

(1) Set the default docker repository to the horbor address;

(2) Set systemd as docker's startup engine;

(3) Execute the docker verison command to return the result and submit the contents of the daemon.json file after the installation is complete.

3. Install docker-compose 【0.5 points】

(1) Install docker-compose on the master node by using the files under the /opt/paas/docker-compose/v1.25.5-docker-compose-Linux-x 86_6 directory;

(2) Execute the docker-compose version command after the installation is complete, and submit the program return results to the answer box.

4. Upload docker image 【1 point】

(1) Upload all images to the docker repository by using /opt/paas/k8s_image_push.sh on the master node;

(2) Take a screenshot of the library image list in the Harbor repository and submit the screenshot to the answer box.

5. Install the kubeadm tool 【0.5 points】

(1) Install the Kubeadm tool on master and node nodes , sset it to automatically start at boot;

(2) Check whether the Kubeadm tool is installed correctly by using the rpm command with grep after the installation is complete;

(3) Submit the return result of the command to the answer box.

6. Compute nodes obtain necessary images 【1 point】

Pull and install the kubernetes basic image on all nodes by using the docker command. After the pull is complete, use the docker command to view the image list.

7. kubeadm install master 【2 points】

(1) Initialize the master node by using the kubeadm command;

(2) Set the kubernetes internal virtual network segment to 10.244.0.0/16;

(3) Complete the initialization settings of the control node through kube-flannel.yaml;

(4) View the cluster status and all pods by using commands after completion;

(5) Submit the above commands and return results to the answer box.

8. Install the kubernetes network plugin 【0.5 points】

(1) Install the kubernetes network plugin through kube-flannel.yaml;

(2) After the installation is complete, view the node status by using commands;

(3) View the cluster status by using commands after completion;

(4) Submit the cluster status check command and the returned result to the answer box.

9. Install kubernetes GUI 【1 point】

Install the kubernetes dashboard interface, view the home page after completion, and submit the screenshot of the kubernetes dashboard interface to the answer box.

10.Compute Node Scaling 【0.5 points】

(1) View the token by using the kubeadm command on the master node;

(2) Join the node to the kubernetes cluster by using the kubeadm command on all nodes;

(3) After completion, check the status of all nodes on the master node;

(4) Submit the cluster status check command and return result to the answer box.

Task 3: K8s operation and maintenance (15 points)

1. Build the dokcer image by using dockerfile.【1 points】

(1) Create a mysql image based on the mysql:5.7 image;

(2) Initialize the provided sql file into the mysql database, and then use the written dockerfile file to create an image named mysql:latest;

(3) Upload the image to the harbor repository built earlier, write YAML files , validate database contents;

(4) After completion, submit the contents of the dockerfile file, the image list of the harbor repository, and the database contents to the answer box.

2. Persistent storage 【2 points】

(1) Build NFS shared storage, configure nfs-provisioner, and create storageclass;

(2) Dynamically generate pvc through storageclass, the size is 1Gi;

(3) Modify the standard nfs-deployment.yaml file;

(4) Write storageclass.yaml and pvc.yaml files;

(5) Submit the final pvc status screenshot and yaml file to the answer box.

3. Write a deployment file 【3 points】

(1) Upload the provided nginx:latest image to the harbor image repository;

(2) Use this image to write a deployment file, and it is required to mount the created pvc to the /html directory, and the number of copies is 1;

(3) Realize resource limitation: Requires 300Mi of memory, 300M of cpu, 450Mi of memory, 450M of CPU;

(4) Submit the POD status screenshot and yaml file to the answer box.

4. Create a service to provide external access interfaces 【3 points】

(1) Based on the nginx pod service, write a service named nginx-svc;

(2) The service port type of proxy nginx is nodeport. After successful creation, you can access nginx through this service;

(3) Submit the yaml file and access the screenshot to the answer box after completion.

5. Configure metrics-server for resource monitoring 【2 points】

Upload the provided metrics-server image to harbor, modify components.yaml, create metrics-server, and submit the metrics-server status screenshot to the answer box.

6. Configure elastic scaling 【2 points】

Write the deployment-nginx-hpa.yaml file, which requires the minimum number of copies of 1, the maximum number of copies of 3 and automatic expansion when the overall resource utilization exceeds 80%. Submit the yaml file to the answer box.

7. Stress Test 【2 points】

Install the httpd-tools tool, perform stress testing through the external access interface provided by the service. Verify elastic scaling function of the HPA, and submit the HPA status and POD status screenshots to the answer box.

Session C: Automated Deployment and O&M of Enterprise-level Applications

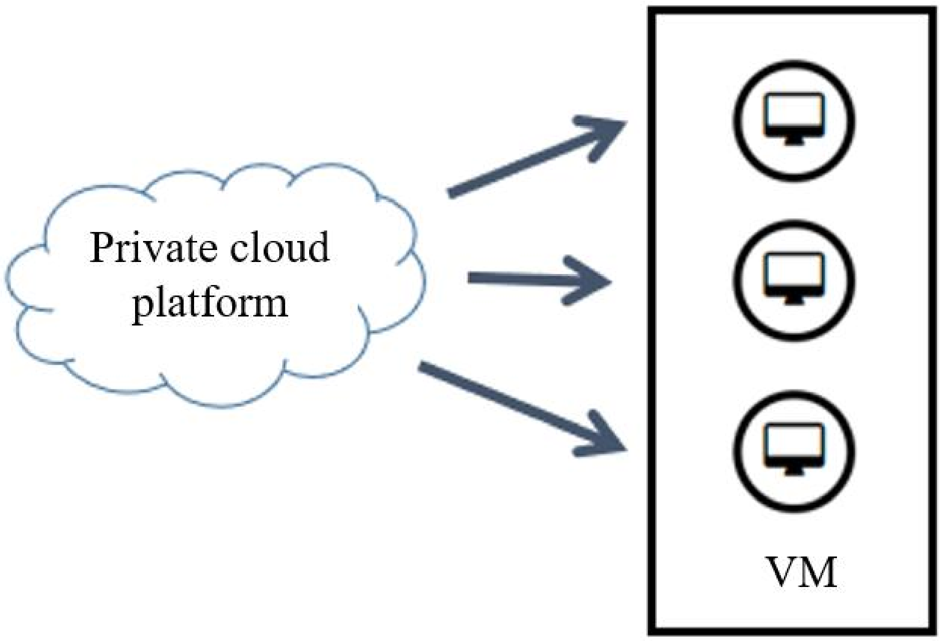


Table 1 Planning of Virtual Machine and Environment

Host name	IP address	roles
Master	Private cloud delivery	Prometheus,grafana,ansible
Node1	Private cloud delivery	agent
Node2	Private cloud delivery	agent

Task 1 Automated deployment of enterprise applications (15 points)

1. Install ansible automated O&M tools【3 points】

(1) Install ansible on the master node using the provided package, and verify that the installation was successful with the ansible --version command after the installation is complete;
(2) Add the test user to all nodes with user password of 000000, and set password-free sudo for the test user;
(3) Configure ssh password-free login so that the master node can log in to the test user of all nodes without password;
(4) Submit the ansible --version command and echo into the answer box.

2. Initialize the ansible automated o&m tool【3 points】

(1) Create the /root/ansible directory as the working directory, and create the ansible.cfg file in this directory;
(2) Place the inventory file in /root/ansible/inventory, and the login user is test without password when logging in;
(3) Set the number of parallel hosts to 2, and allow the test user to escalate to root without password;
(4) Submit the content of the ansible.cfg file into the answer box.

3. Write a host inventory【2 points】

(1) Write a host inventory file, including creating the master user group and adding the master host to the master user group;
(2) Add the node1 and node2 hosts to the created node group, and the host name must not use an IP address;
(3) Execute the ansible-inventory --list, ansible all -m ping and ansible all -a "id" commands and submit the returned results of these three commands into the answer box.

4. Initialize the master node using automated tools【2 points】

(1) Write prometheus.yml to control the master host group, and set the SELinux temporary state setting and startup state to disabled by using the corresponding module;
(2) Install the time synchronization service by using the corresponding module of ansible, use the text editing module to set the scope of the service to 0.0.0.0/0, and set the status to start and automatically start at boot;
(3) Copy the provided prometheus-2.37.0.linux-amd64.tar.gz to the /usr/local/ directory of the target host using the file copy module, and use the shell module to decompress the compressed package;
(4) Submit the yml file and ansible run result to the answer box after completion.

5. Deploy enterprise-grade applications with automated operation and maintenance tools【5 points】

(1) Write the prometheus.yml.j2 template file and add all node information to the file, but the hostname information of the managed node must be entered manually using the variable IP address;
(2) Create the node_exporter.yml file after completion, and write the first play named node that controls the node host group;
(3) Use a shell module to decompress the compressed package, which uses the ansible module to send node_exporter-1.3.1.linux-amd64.tar.gz to /usr/local/ directory of the node host group, and start the service;
(4) Write the second play named master, and the second play controls the master node;
(5) Use the ansible module to transfer the prometheus.yml.j2 file to the master node, then start prometheus by using the script module;
(6) Send the grafana-8.1.2-1.x86_64.rpm package to the /mnt/ directory of the controlled node by using the corresponding module, and then install the package;
(7) Set the grafana service to start and automatically start at boot after the installation is complete;
(8) Log in to prometheus through browser to check whether prometheus successfully monitors all nodes;
(9) Submit the screenshot of the browser feedback, the content of the prometheus.yml.j2 file, the content of the node_exporter.yml file, and the running result to the answer box.

Task 2 O&M of enterprise-level applications (12 points)

1. Monitor mysqld service by using prometheus【3 points】

(1) Send the provided mysqld_exporter-0.14.0.linux-amd64.tar.gz to the agent virtual machine /usr/local/ directory, decompress and install the mariadb service;
(2) Create the mysqld_monitor user and authorize it in the mariadb database, and then create a mariadb

configuration file with the database username and password;
(3) Start the mysqld_exporter component and make sure port 9104 is enabled;
(4) Go back to the prometheus node and add the mysql monitored information in the prometheus.yml file;
(5) Restart prometheus and then refresh the web interface to view the controlled information of mysqld;
(6) Submit the above actions to the answer box.

2. Install alarm component(alertmanager) 【3 points】

(1) Upload the provided alertmanager-0.21.0.linux-amd64.tar.gz to the prometheus node /usr/local/ directory and extract it;
(2) Create a soft link alertmanager-0.23.0.linux-amd64/alertmanager;
(3) Start the alertmanager and view port 9093 after creating the service startup file;
(4) Add the alertmanager information to the prometheus.yml configuration file and restart the prometheus service;
(5) Check whether the status of the alarm manager is normal and whether the status of the agent is abnormal in the web after the agent stops the node_exporter service;
(6) Submit operation commands and screenshots to the answer box.

3. Write alert email file (alertmanager) 【3 points】

Please write the email address information to which the alert will be sent according to the provided address and template of the alertmanager.yml file in the Prometheus virtual machine /usr/local/akertmanager/ directory. Submit the content written in the configuration file to the answer box.

4. Write alarm rules(alertmanager) 【4 points】

There is node_rules.yml file of rules directory in the prometheus path of the prometheus virtual machine. Please write according to the template of providing information:

(1) Alarm rules for memory greater than 50%;
(2) Alarm rule for cpu resource utilization greater than 75%;
(3) Alarm rule for the host disk reads data per second greater than 50mb.

The name of the department can be the job number, and the content of the configuration file is submitted to the answer box.

Task 3 O&M of enterprise-level microservice (13 points)

1. Install istio in K8S cluster 【4 points】

Upload istio-1.10.1-linux-amd64.tar to the master node and extract it to the /root directory. Upload the container image file in the /istio directory to each node and load it to complete the initial installation of istio. Submit the screenshots of the three POD states after successful deployment to the answer board.

2. Deploy the online bookstore bookinfo application 【6 points】

Bookinfo is an online bookstore microservice application, which is divided into four separate microservices:

(1) Productpage. This microservice will generate pages by calling two microservices of details and reviews;

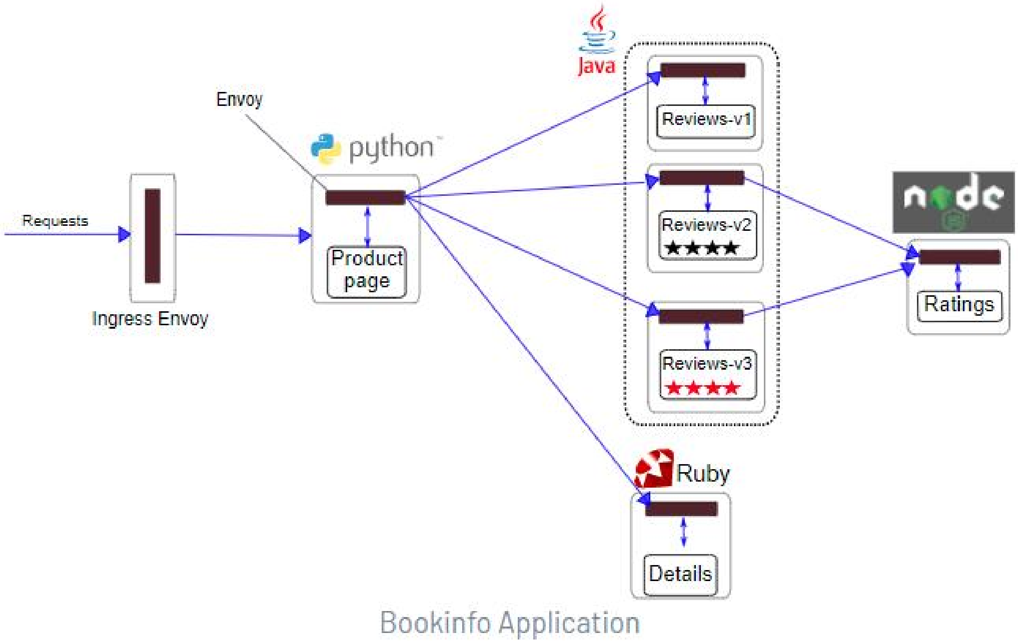
(2) Details. This microservice contains book information;

(3) Reviews. This microservice contains book-related reviews and calls the ratings microservice;

(4) Ratings. This microservice contains rating information consisting of book reviews.

There are 3 versions of the reviews microservice: v1 does not call the ratings service; v2 calls the ratings service and display rating information with 1 to 5 black star icons; The v3 version calls the ratings service and displays rating information with 1 to 5 red star icons.

Implement the deployment of the application service mesh, and provide service routing, monitoring data collection, and policy enforcement for use application by using the Istio control plane.



(1) Write a Dockerfile to build the productpage:v1.0 image. The specific requirements are as follows:

Basic image: centos:7.9.2009;

Install Python3.6.8 tools;

Install productpage service and set up self-starting.

(2) Plan the application microservice deployment architecture, and complete the deployment of the bookinfo microservice application by modifying the yaml file of the microservice application in the container cluster;

(3) Set circuit breaker rules in the container cluster, set circuit breaker rules for the productpage microservice in Bookinfo, and verify the circuit breaker mechanism triggered by the load fortio test client.

3. Deploy the gateway interface to realize external services【3 points】

Deploy the bookinfo gateway through istio-1.10.1/samples/bookinfo/networking/bookinfo-gateway.yaml, and log in to bookinfo through an external browser. Please submit screenshots of browser feedback results to the answer board.