1Z0-1084-22 Q&As

Oracle Cloud Infrastructure 2022 Developer Professional

Pass Oracle 1Z0-1084-22 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.pass4itsure.com/1z0-1084-22.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Oracle Official Exam Center

- **Instant Download** After Purchase
- **100% Money Back** Guarantee
- **365 Days** Free Update
- **800,000+** Satisfied Customers

QUESTION 1

A developer using Oracle Cloud Infrastructure (OCI) API Gateway must authenticate the API requests to their web application. The authentication process must be implemented using a custom scheme which accepts string parameters from the API caller. Which method can the developer use In this scenario?

A. Create an authorizer function using request header authorization.

B. Create an authorizer function using token-based authorization.

C. Create a cross account functions authorizer.

D. Create an authorizer function using OCI Identity and Access Management based authentication

Correct Answer: B

Having deployed the authorizer function, you enable authentication and authorization for an API deployment by including two different kinds of request policy in the API deployment specification:

An authentication request policy for the entire API deployment that specifies:The OCID of the authorizer function that you deployed to Oracle Functions that will perform authentication and authorization.The request attributes to pass to the

authorizer function.Whether unauthenticated callers can access routes in the API deployment.

An authorization request policy for each route that specifies the operations a caller is allowed to perform, based on the caller\\'s access scopes as returned by the authorizer function. Using the Console to Add Authentication and Authorization

Request Policies To add authentication and authorization request policies to an API deployment specification using the Console:

Create or update an API deployment using the Console, select the From Scratch option, and enter details on the Basic Information page. For more information, see Deploying an API on an API Gateway by Creating an API Deployment and

Updating API Gateways and API Deployments. In the API Request Policies section of the Basic Information page, click the Add button beside Authentication and specify:

Application in : The name of the application in Oracle Functions that contains the authorizer function. You can select an application from a different compartment. Function Name: The name of the authorizer function in

Oracle Functions. Authentication Token: Whether the access token is contained in a request header or a query parameter.

Authentication Token Value: Depending on whether the access token is contained in a request header or a query parameter, specify:

Header Name: If the access token is contained in a request header, enter the name of the header. Parameter Name: If the access token is contained in a query parameter, enter the name of the query parameter.

https://docs.cloud.oracle.com/en-us/iaas/Content/APIGateway/Tasks/apigatewayaddingauthzauthn.htm

QUESTION 2

You are a consumer of Oracle Cloud Infrastructure (OCI) Streaming service. Which API should you use to read and process the stream?

A. ListMessages

B. GetMessages

C. GetObject

D. ReadMessages

Correct Answer: B

https://docs.cloud.oracle.com/en-us/iaas/Content/Streaming/Concepts/streamingoverview.htm Building consumers to read and process messages from a stream using the GetMessages API.

QUESTION 3

Which Oracle Cloud Infrastructure (OCI) load balancer shape is used by default in OCI container Engineer for Kubernetes?

A. 400 Mbps

B. 8000 Mbps

C. There is no default. The shape has to be specified.

D. 100 Mbps

Correct Answer: D

Specifying Alternative Load Balancer Shapes The shape of an Oracle Cloud Infrastructure load balancer specifies its maximum total bandwidth (that is, ingress plus egress). By default, load balancers are created with a shape of 100Mbps. Other shapes are available, including 400Mbps and 8000Mbps. https://docs.cloud.oracle.com/en-us/iaas/Content/ContEng/Tasks/contengcreatingloadbalancer.htm

QUESTION 4

Your Oracle Cloud Infrastructure Container Engine for Kubernetes (OKE) administrator has created an OKE cluster with one node pool in a public subnet. You have been asked to provide a log file from one of the nodes for troubleshooting

purpose.

Which step should you take to obtain the log file?

A. ssh into the node using public key.

B. ssh into the nodes using private key.

C. It is impossible since OKE is a managed Kubernetes service.

D. Use the username open and password to login.

Correct Answer: B

Kubernetes cluster is a group of nodes. The nodes are the machines running applications. Each node can be a physical machine or a virtual machine. The node\\'s capacity (its number of CPUs and amount of memory) is defined when the

node is created. A cluster comprises:

- one or more master nodes (for high availability, typically there will be a number of master nodes)

- one or more worker nodes (sometimes known as minions) Connecting to Worker Nodes Using SSH

If you provided a public SSH key when creating the node pool in a cluster, the public key is installed on all worker nodes in the cluster. On UNIX and UNIX-like platforms (including Solaris and Linux), you can then connect through SSH to the

worker nodes using the ssh utility (an SSH client) to perform administrative tasks.

Note the following instructions assume the UNIX machine you use to connect to the worker node:

Has the ssh utility installed.

Has access to the SSH private key file paired with the SSH public key that was specified when the cluster was created.

How to connect to worker nodes using SSH depends on whether you specified public or private subnets for the worker nodes when defining the node pools in the cluster. Connecting to Worker Nodes in Public Subnets Using SSH Before you

can connect to a worker node in a public subnet using SSH, you must define an ingress rule in the subnet\\'s security list to allow SSH access. The ingress rule must allow access to port 22 on worker nodes from source 0.0.0.0/0 and any

source port To connect to a worker node in a public subnet through SSH from a UNIX machine using the ssh utility:

1- Find out the IP address of the worker node to which you want to connect. You can do this in a number of ways: Using kubectl. If you haven\\'t already done so, follow the steps to set up the cluster\\'s kubeconfig configuration file and (if necessary) set the KUBECONFIG environment variable to point to the file. Note that you must set up your own kubeconfig file. You cannot access a cluster using a kubeconfig file that a different user set up. See Setting Up Cluster Access. Then in a terminal window, enter kubectl get nodes to see the public IP addresses of worker nodes in node pools in the cluster. Using the Console. In the Console, display the Cluster List page and then select the cluster to which the worker node belongs. On the Node Pools tab, click the name of the node pool to which the worker node belongs. On the Nodes tab, you see the public IP address of every worker node in the node pool. Using the REST API. Use the ListNodePools operation to see the public IP addresses of worker nodes in a node pool. 2- In the terminal window, enter ssh opc@ to connect to the worker node, where is the IP address of the worker node that you made a note of earlier. For example, you might enter ssh opc@192.0.2.254. Note that if the SSH private key is not stored in the file or in the path that the ssh utility expects (for example, the ssh utility might expect the private key to be stored in ~/.ssh/id_rsa), you must explicitly specify the private key filename and location in one of two ways: Use the -i option to specify the filename and location of the private key. For example, ssh -i ~/.ssh/my_keys/my_host_key_filename opc@192.0.2.254 Add the private key filename and location to an SSH configuration file, either the client configuration file (~/.ssh/config) if it exists, or the system-wide client configuration file (/etc/ssh/ssh_config). For example, you might add the following: Host 192.0.2.254 IdentityFile ~/.ssh/my_keys/my_host_key_filename For more about the ssh utility\\'s configuration file, enter man ssh_config Note also that permissions on the private key file must allow you read/write/execute access, but prevent other users from accessing the file. For example, to set appropriate permissions, you might enter chmod 600 ~/.ssh/my_keys/my_host_key_filename. If permissions are not set correctly and the private key file is accessible to other users, the ssh utility will simply ignore the private key file.

QUESTION 5

Your organization uses a federated identity provider to login to your Oracle Cloud Infrastructure (OCI) environment. As a developer, you are writing a script to automate some operation and want to use OCI CLI to do that. Your security team

doesn\\'t allow storing private keys on local machines.

How can you authenticate with OCI CLI?

A. Run oci setup keys and provide your credentials

B. Run oci session refresh --profile

C. Run oci session authenticate and provide your credentials

D. Run oci setup oci-cli-rc --file path/to/target/file

Correct Answer: C

Token-based authentication for the CLI allows customers to authenticate their session interactively, then use the CLI for a single session without an API signing key. This enables customers using an identity provider that is not SCIM-

supported to use a federated user account with the CLI and SDKs.

Starting a Token-based CLI Session

To use token-based authentication for the CLI on a computer with a web browser:

In the CLI, run the following command. This will launch a web browser.

oci session authenticate

In the browser, enter your user credentials. This authentication information is saved to the .config file.