Free Questions for 1Z0-1084-23 by ebraindumps

Shared by Beard on 22-07-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Your team has chosen to use master encryption key (MEK) within an Oracle Cloud Infrastructure (OCI) Vault for encrypting Kubernetes secrets associated with your microservice deployments in OCI Container Engine for Kubernetes (OKE) clusters so that you can easily manage key rotation. Which of the following is NOT valid about rotating keys in the OCI Vault service?

Options:

A- Once rotated, older key versions can be used for encryption until they are deleted.

B- Both software and HSM-protected MEKS can be rotated.

C- When you rotate an MEK, a new key version is automatically generated.

D- Each key version is tracked internally with separate unique OCIDS.

Answer:

A

Explanation:

The correct answer is: 'Once rotated, older key versions can be used for encryption until they are deleted.' The statement that is NOT valid about rotating keys in the OCI Vault service is: 'Once rotated, older key versions can be used for encryption until they are deleted.' In the OCI Vault service, when you rotate a master encryption key (MEK), a new key version is automatically generated. However, once a key is rotated and a new version is created, the older key versions are no longer usable for encryption. The purpose of key rotation is to ensure that the encryption keys are regularly updated and that older keys are no longer used to protect sensitive data. This enhances security by minimizing the impact of potential key compromises. The other statements mentioned are valid: Both software and hardware security module (HSM)-protected MEKs can be rotated. This provides flexibility in choosing the type of MEK and ensures that key rotation can be performed regardless of the encryption method used. Each key version is tracked internally with separate unique OCIDs (Oracle Cloud Identifiers). This allows for easy management and tracking of different key versions within the OCI Vault service. In summary, the statement that is NOT valid is the one suggesting that older key versions can still be used for encryption until they are deleted. Key rotation is designed to ensure the use of the latest key version and to retire older key versions to enhance security.

Question 2

Question Type: MultipleChoice

To effectively test your cloud native applications for "unknown unknowns", you need to employ various testing and deployment strategies. Which strategy involves exposing new functionality or features to only a small set of users?

Options:

A- A/B Testing

B- Component Testing

C- Blue/Green Deployment

D- Canary Deployment

Answer:

D

Explanation:

The strategy that involves exposing new functionality or features to only a small set of users is called Canary Deployment. Canary deployment is a technique used in software development and deployment where a new version of an application or feature is released to a small subset of users or a specific group of servers. This allows for testing and gathering feedback on the new functionality in a controlled and limited environment before making it available to a wider audience. In a canary deployment, a small portion of the traffic is routed to the new version while the majority of the traffic still goes to the stable version. This allows for monitoring and evaluation of the new functionality in real-world conditions while minimizing the impact of any potential issues or bugs. If the new version performs well and meets the desired criteria, it can then be gradually rolled out to a larger user base or all servers. By exposing the new functionality or features to a small set of users initially, canary deployment helps in identifying any unforeseen issues, gathering feedback, and ensuring the stability and reliability of the application before a full deployment.

Question 3

Question Type: MultipleChoice

(CHK_4>2) Which TWO statements are NOT valid regarding the Oracle Cloud Infrastructure (OCI) Streaming service? (Choose two.)

Options:

A- OCI Streaming stores all data for 24 hours by default, but that can be extended up to 7 days.B

B- Although OCI Streaming automatically encrypts all data while in transit, it is the developer's responsibility to encrypt data at rest, if needed.

C- The throughput of a stream is defined by a partition. A partition provides 1 MB/sec data input and 2 MB/sec data output.

D- A stream can be configured with either a public or a private endpoint with support for customer managed encryption keys.

E- OCI Streaming can support up to 2,000 requests per second to each partition.

Answer:

D, E

Explanation:

The two statements that are NOT valid regarding the Oracle Cloud Infrastructure (OCI) Streaming service are: A stream can be configured with either a public or a private endpoint with support for customer managed encryption keys. This statement is not valid because the OCI Streaming service currently supports only private endpoints. Customer managed encryption keys are not currently supported for OCI Streaming. OCI Streaming can support up to 2,000 requests per second to each partition. This statement is not valid because the throughput of a stream is not defined by the partition in terms of requests per second. The throughput of a stream is defined in terms of data input and output rates. Each partition provides 1 MB/sec data input and 2 MB/sec data output, but it does not correspond to a specific number of requests per second. The other statements are valid: OCI Streaming stores all data for 24 hours by default, but that can be extended up to 7 days. Although OCI Streaming automatically encrypts all data while in transit, it is the developer's responsibility to encrypt data at rest, if needed.

Question 4

Question Type: MultipleChoice

(CHK_4>3) Your development team decides to create and deploy some business logic to serverless Oracle Functions. You are asked to help facilitate the monitoring, logging, and tracing of these services. Which is NOT valid about troubleshooting Oracle Functions?

Options:

A- Oracle Functions invocation is enabled by default

B- Oracle Functions invocation logs are enabled at the application level.

C- Oracle Functions metrics are available at both the function and application level.

D- Oracle Functions tracing is enabled at the function level.

Answer:

D

Explanation:

The option that is NOT valid about troubleshooting Oracle Functions is: 'Oracle Functions tracing is enabled at the function level.' In Oracle Functions, tracing is not enabled at the function level. Instead, tracing is enabled at the application level. When you enable tracing for an application, it applies to all the functions within that application. Tracing allows you to capture detailed information about the execution flow and performance of the functions, helping you analyze and debug issues. The other options mentioned are valid: Oracle Functions invocation logs are enabled at the application level. Invocation logs provide visibility into the details of function invocations, including input, output, duration, and any error messages. These logs are generated and stored by Oracle Functions, and you can access them for troubleshooting and monitoring purposes. Oracle Functions invocation is enabled by default. Once you deploy a function, it becomes invocable by default. You can configure different triggers to invoke the function, such as HTTP requests, scheduled events, or events from other Oracle Cloud Infrastructure services. Oracle Functions metrics are available at both the function and application level. Metrics provide insights into the usage, performance, and behavior of functions. They can include metrics such as invocations per minute, average duration, and error counts. These metrics can be viewed in the Oracle Cloud Infrastructure Console or accessed programmatically through APIs. It's important to note that the specific configuration and behavior of monitoring, logging, and tracing in Oracle Functions may depend on the version, configuration, and options you have chosen. It is recommended to refer to the Oracle Functions documentation and consult the official documentation for accurate and up-to-date information on troubleshooting and

monitoring Oracle Functions.

Question 5

Question Type: MultipleChoice

From a DevOps process standpoint, it is a good practice to keep changes to an application under version control. Which of the following allows changes to a Docker image to be stored in a version control system?

Options:

A- Updating docker-compose.yml

B- Executing docker commit

C- Executing docker save

D- Updating Dockerfile

Answer:

B

Explanation:

The option that allows changes to a Docker image to be stored in a version control system is: docker commit The docker commit command is used to create a new image from a container's changes. It takes a running container as input, captures the changes made to it, and creates a new image with those changes. This new image can then be tagged and pushed to a registry, or saved locally. By using docker commit, you can effectively capture the changes made to a container as a new image and store it in a version control system along with the Dockerfile and other project files. This allows for reproducibility and traceability of changes to the Docker image over time.

Question 6

Question Type: MultipleChoice

Which is NOT a valid option to execute a function deployed in Oracle Functions?

Options:

A- Invoke from the Docker CLI.

B- Send signed HTTP requests to the function's invoke endpoint.

C- Invoke from the Fn Project CLI.

D- Trigger by an event in the Oracle Cloud Infrastructure (OCI) Events service.

E- Invoke from the OCI CLI.

Answer:

A

Explanation:

The correct answer is: Invoke from the Docker CLI. Executing a function deployed in Oracle Functions is typically done using the following options: Invoke from the Fn Project CLI: The Fn Project CLI provides a command-line interface specifically designed for interacting with Oracle Functions. You can use commands like fn invoke to invoke a function. Trigger by an event in the Oracle Cloud Infrastructure (OCI) Events service: You can configure events in OCI to trigger your function based on various criteria, such as object storage events, resource state changes, or scheduled events. Invoke from the OCI CLI: The OCI CLI (Command Line Interface) allows you to interact with various services in Oracle Cloud Infrastructure, including Oracle Functions. You can use the fn invoke command to invoke a function. Send signed HTTP requests to the function's invoke endpoint: Oracle Functions provides an HTTP endpoint that can be used to invoke functions. You can send signed HTTP requests to this endpoint using tools or programming languages that support making HTTP requests. On the other hand, invoking a function deployed in Oracle Functions using the Docker CLI is not a valid option. The Docker CLI is primarily used for managing Docker containers and images, and it does not provide a direct mechanism for invoking functions in Oracle Functions.

Question 7

Question Type: MultipleChoice

Which TWO are part of the Cloud Native Computing Foundation (CNCF) container runtime? (Choose two.)

Options:

A- rkt-o

B- runc

C- getcd

D- containerd

Answer:

B, D

Explanation:

The two components that are part of the Cloud Native Computing Foundation (CNCF) container runtime are: containerd: containerd is an open-source container runtime that provides a runtime environment for containers, including managing container images, executing containers, and handling container lifecycle events. It is designed to be lightweight and extensible, providing the necessary functionality

to run containers efficiently. runc: runc is a lightweight container runtime that serves as a reference implementation of the Open Container Initiative (OCI) runtime specification. It is responsible for launching and managing containers based on OCI specifications, including handling container isolation, namespaces, cgroups, and other low-level container operations. These two components, containerd and runc, are widely used in the container ecosystem and are part of the CNCF's efforts to promote and develop open-source technologies for cloud-native computing.

Question 8

Question Type: MultipleChoice

Which statement accurately describes the Oracle Cloud Infrastructure (OCI) Load Balancer integration with OCI Container Engine for Kubernetes (OKE)?

Options:

A- OKE service provisions an OCI Load Balancer instance for each Kubernetes service with LoadBalancer type in the YAML configuration.

B- OKE service provisions a single OCI Load Balancer instance shared with all the Kubernetes services with LoadBalancer type in the YAML configuration.

C- OCI Load Balancer instance provisioning is triggered by the OCI Events service for each Kubernetes service with LoadBalancer type

in the YAML configuration.

D- OCI Load Balancer instance must be manually provisioned for each Kubernetes service that requires traffic balancing.

Answer:

A

Explanation:

The statement that accurately describes the Oracle Cloud Infrastructure (OCI) Load Balancer integration with OCI Container Engine for Kubernetes (OKE) is: 'OKE service provisions an OCI Load Balancer instance for each Kubernetes service with LoadBalancer type in the YAML configuration.' When you define a Kubernetes service in your YAML configuration with the LoadBalancer type, the OKE service automatically provisions an OCI Load Balancer instance specifically for that service. This Load Balancer instance is dedicated to the Kubernetes service and provides traffic balancing functionality. Each Kubernetes service that requires load balancing will have its own OCI Load Balancer instance provisioned by OKE.

Question 9

Question Type: **MultipleChoice**

You are developing a distributed application and you need a call to a path to always return a specific JSON content deploy an OCI API Gateway with the below API deployment specification. What is the correct value for type? { "routes" : [{ "path" : "/hello", "methods" : ["Get"), "backend" : { "type" : " ---------- ", "status" : 200, "headers" : [{ "name" : "Content-Type", "value" : "application/json" }] "body" : "{\"myjson\": \"consistent response\"}" }}]}

Options:

A- STOCK_RESPONSE_BACKEND

B- CONSTANT_BACKEND

C- JSON_BACKEND

D- HTTP_BACKEND

Answer:

A

Explanation:

The correct value for the 'type' field in the API deployment specification is 'STOCK_RESPONSE_BACKEND'. By setting the 'type' to 'STOCK_RESPONSE_BACKEND', you are indicating that the backend for the specified route should return a pre-defined response. This type of backend is commonly used when you want a specific response to be returned consistently, regardless of the actual backend service implementation. In this case, the API deployment specification is configured to have a single route with the path '/hello' and the method 'GET'. The backend section specifies the type as 'STOCK_RESPONSE_BACKEND'. Additionally, it defines the response status

code as 200, sets the 'Content-Type' header to 'application/json', and provides the JSON content in the 'body' field. Using this configuration, any request to the '/hello' path with the 'GET' method will always receive a consistent JSON response with the content '{'myjson': 'consistent response'}'.