SECRET-SEN Q&As

CyberArk Sentry - Secrets Manager

Pass CyberArk SECRET-SEN Exam with 100% Guarantee

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

https://www.pass2lead.com/secret-sen.html

100% Passing Guarantee
100% Money Back Assurance

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

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

QUESTION 1

A customer requires high availability in its AWS cloud infrastructure.

What is the minimally viable Conjur deployment architecture to achieve this?

A. one Follower in each AZ. load balancer for the region

B. two Followers in each region, load balanced for the region

C. two Followers in each AZ. load balanced for the region

D. two Followers in each region, load balanced across all regions

Correct Answer: A

According to the CyberArk Sentry Secrets Manager documentation, Conjur is a secrets management solution that consists of a leader node and one or more follower nodes. The leader node is responsible for managing the secrets, policies,

and audit records, while the follower nodes are read-only replicas that can serve secrets requests from applications. To achieve high availability in AWS cloud infrastructure, the minimally viable Conjur deployment architecture is to have one

follower in each availability zone (AZ) and a load balancer for the region. This way, if one AZ fails, the applications can still access secrets from another AZ through the load balancer. Having two followers in each region, load balanced for the

region, is not enough to ensure high availability, as a regional outage can affect both followers. Having two followers in each AZ, load balanced for the region, is more than necessary, as one follower per AZ can handle the secrets requests.

Having two followers in each region, load balanced across all regions, is not feasible, as Conjur does not support cross-region replication.

References: 1: Conjur Architecture 2: Deploying Conjur on AWS

QUESTION 2

When working with Credential Providers in a Privileged Cloud setting, what is a special consideration?

A. If there are installation issues, troubleshooting may need to involve the Privileged Cloud support team.

B. Credential Providers are not supported in a Privileged Cloud setting.

C. The AWS Cloud account number must be defined in the file main appprovider.conf.. found in the AppProviderConf Safe.

D. Debug logging for Credential Providers deployed in a Privileged Cloud setting can inadvertently exhaust available disk space.

Correct Answer: A

Credential Providers are tools that enable applications to securely retrieve credentials from CyberArk Secrets Manager without hard-coding or storing them in files. Credential Providers can be installed on application servers or on a central

server that acts as a proxy for multiple applications. Credential Providers can integrate with Privileged Cloud, which is a cloud-based solution that provides privileged access management as a service. Privileged Cloud integrates with Secrets Manager Credential Providers to manage application credentials as privileged accounts within Privileged Cloud. When working with Credential Providers in a Privileged Cloud setting, a special consideration is that if there are installation issues, troubleshooting may need to involve the Privileged Cloud support team. This is because the installation of Credential Providers in a Privileged Cloud setting requires some additional steps and configurations that are performed by the Privileged Cloud support team. For example, the Privileged Cloud support team needs to configure the connection between Privileged Cloud and Credential Providers, and provide the necessary certificates and keys for secure communication. Therefore, if there are any problems or errors during the installation process, the Privileged Cloud support team may need to assist with the troubleshooting and resolution. The other options are not correct. Credential Providers are supported in a Privileged Cloud setting, as described in the Secrets Manager Credential Providers integration documentation1. The AWS Cloud account number does not need to be defined in the file main appprovider.conf.. found in the AppProviderConf Safe. This file is used to configure the Credential Provider settings, such as the Privileged Cloud URL, the application ID, and the SSL options. The AWS Cloud account number is not relevant for this file. Debug logging for Credential Providers deployed in a Privileged Cloud setting can be enabled or disabled by the Privileged Cloud support team, as described in the Credential Provider installation documentation2. Debug logging can help with troubleshooting and diagnostics, but it does not necessarily exhaust available disk space, as the log files can be rotated and archived. References: Secrets Manager Credential Providers integration; Credential Provider installation

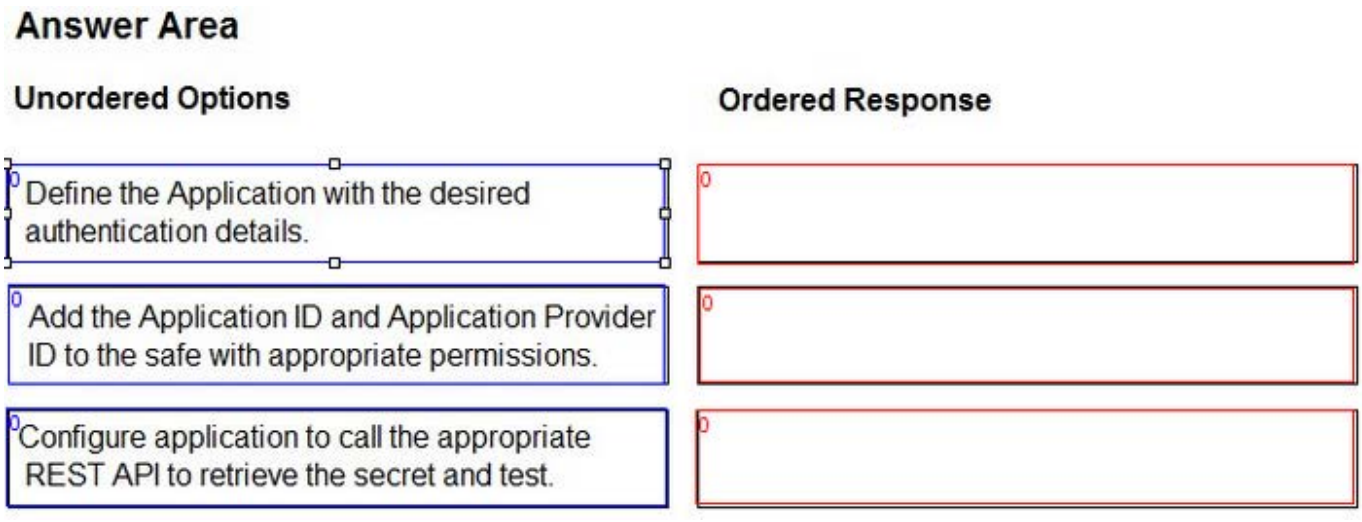
QUESTION 3

DRAG DROP

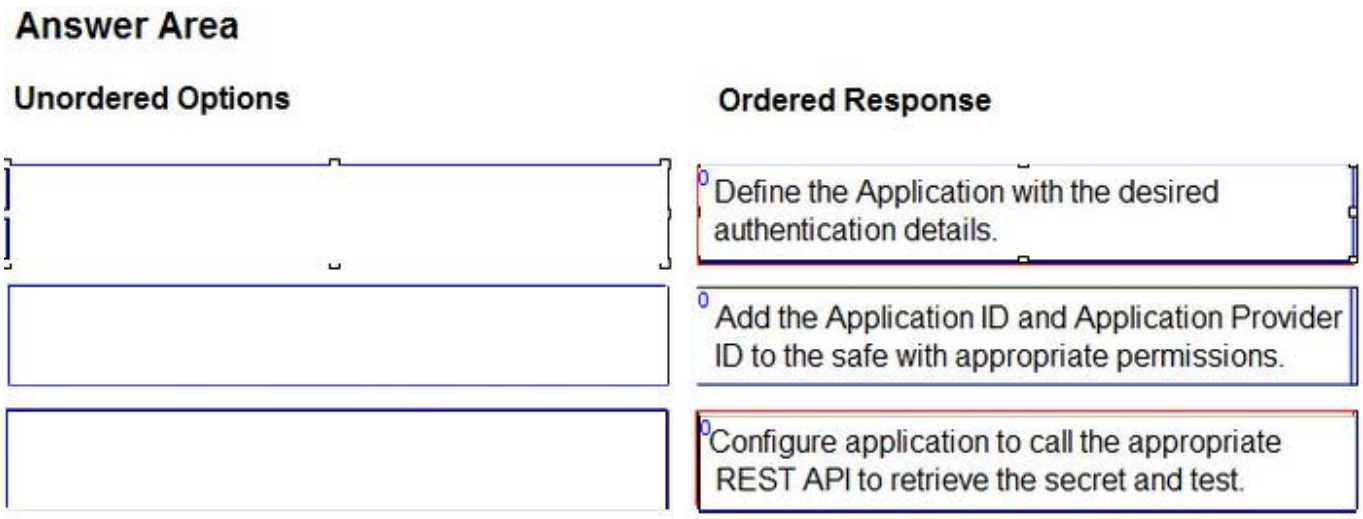
You want to allow retrieval of a secret with the CCP. The safe and the required secrets already exist.

Assuming the CCP is installed, arrange the steps in the correct sequence.

Select and Place:



Correct Answer:



The correct order of the steps is: Define the Application with the desired authentication details Add the Application ID and Application Provider ID to the safe with appropriate permissions Configure application to call the appropriate REST API to retrieve the secret and test To allow an application to retrieve a secret with the CCP, the following steps are required: Define the Application with the desired authentication details: This step involves creating an Application object in the Vault with a unique Application ID and an Application Provider ID. The Application Provider ID is used to identify the CCP instance that will serve the request. The Application object also defines the authentication method and parameters that the application will use to connect to the CCP, such as certificate, password, or AppRole. Add the Application ID and Application Provider ID to the safe with appropriate permissions: This step involves granting the Application object the necessary permissions to access the safe and the secret that it needs. The Application ID and the Application Provider ID are added as members of the safe with at least List and Retrieve permissions. The secret name or ID can also be specified as a restriction to limit the access to a specific secret within the safe. Configure application to call the appropriate REST API to retrieve the secret and test: This step involves configuring the application to send a REST API request to the CCP endpoint with the required parameters, such as the Application ID, the Application Provider ID, the safe name, and the secret name or ID. The application should also provide the authentication credentials or token that match the method defined in the Application object. The application should receive a JSON response from the CCP with the secret value and other metadata. The application should test the connection and the secret retrieval before deploying to production. References: CyberArk Secrets Manager Sentry - Secrets Manager - Sample Items and Study Guide Sentry - Secrets Secrets Management Essentials for Developers

QUESTION 4

A customer has 100 .NET applications and wants to use Summon to invoke the application and inject secrets at run time.

Which change to the NET application code might be necessary to enable this?

A. It must be changed to include the REST API calls necessary to retrieve the needed secrets from the CCP.

B. It must be changed to access secrets from a configuration file or environment variable.

C. No changes are needed as Summon brokers the connection between the application and the backend data source through impersonation.

D. It must be changed to include the host API key necessary for Summon to retrieve the needed secrets from a Follower

Correct Answer: B

Summon is a utility that allows applications to access secrets from a variety of trusted stores and export them as environment variables to a sub-process environment. Summon does not require any changes to the application code to retrieve secrets from the CyberArk Central Credential Provider (CCP), as it uses a provider plugin that handles the communication with the CCP. However, the application code must be able to access secrets from a configuration file or environment variable, as these are the methods that Summon uses to inject secrets into the application. Summon reads a secrets.yml file that defines the secrets that the application needs and maps them to environment variables. Then, Summon fetches the secrets from the CCP using the provider plugin and exports them as environment variables to the application sub-process. The application can then read the secrets from the environment variables as if they were hard-coded in the configuration file. References: Summon-inject secrets, .NET Application Password SDK

QUESTION 5

You are setting up the Secrets Provider for Kubernetes to support rotation with Push-to-File mode.

Which deployment option should be used?

A. Init container

B. Application container

C. Sidecar

D. Service Broker

Correct Answer: C

According to the CyberArk Sentry Secrets Manager documentation, the Secrets Provider for Kubernetes can be deployed as an init container or a sidecar in Push- to-File mode. In Push-to-File mode, the Secrets Provider pushes Conjur secrets to one or more secrets files in a shared volume in the same Pod as the application container. The application container can then consume the secrets files from the shared volume. The deployment option that should be used to support rotation with Push-to-File mode is the sidecar, because the sidecar can run continuously and check for updates to the secrets in Conjur. If changes are detected, the sidecar can update the secrets files in the shared volume. The init container, on the other hand, runs to completion and does not support rotation. The application container and the service broker are not valid deployment options for the Secrets Provider for Kubernetes in Push-to-File mode. References: 1: Secrets Provider - Init container/Sidecar - Push-to-File mode 2: Secrets Provider - init container/sidecar - Push-to-File mode

SECRET-SEN VCE Dumps SECRET-SEN Practice Test SECRET-SEN Braindumps