Free Questions for Vault-Associate by ebraindumps

Shared by Little on 12-12-2023

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

A user issues the following cURL command to encrypt data using the transit engine and the Vault AP:

```
curl \
--header "X-Vault-Token: c4f280f6-fdb2-18eb-89d3-589e2e834cdb" \
--request POST \<
--data @payload.json \
http://127.0.0.1:8200/v1/transit/encrypt/my-key
```

Which payload.json file has the correct contents?

A.

```
{
  "plaintext": "dGhlIHF1aWNrIGJyb3duIGZveA=="
}
```

B.

```
{
  "ciphertext": "vault:v1:abcdefgh"
}
```

C.

```
{
  "data": {
    "plaintext": "dGhlIHF1aWNrIGJyb3duIGZveA=="
  }
}
```

D.

```
{
  "data": {
    "ciphertext": "vault:v1:abcdefgh"
  }
}
```

Options:

A- Option A

B- Option B

C- Option C

D- Option D

Answer:

C

Explanation:

The payload.json file that has the correct contents is C. This file contains a JSON object with a single key, "plaintext", and a value that is the base64-encoded string of the data to be encrypted.This is the format that the Vault API expects for the transit encrypt endpoint1. The other files are not correct because they either have the wrong key name, the wrong value format, or the wrong JSON syntax.

Encrypt Data - Transit Secrets Engine | Vault | HashiCorp Developer

Question 2

Question Type: MultipleChoice

When an auth method is disabled all users authenticated via that method lose access.

Options:

A- True

B- False

Answer:

A

Explanation:

The statement is true. When an auth method is disabled, all users authenticated via that method lose access. This is because the tokens issued by the auth method are automatically revoked when the auth method is disabled. This prevents the users from performing any operation in Vault using the revoked tokens. To regain access, the users have to authenticate again using a different auth method that is enabled and has the appropriate policies attached.Reference:Auth Methods | Vault | HashiCorp Developer,auth disable - Command | Vault | HashiCorp Developer

Question 3

Question Type: MultipleChoice

A developer mistakenly committed code that contained AWS S3 credentials into a public repository. You have been tasked with revoking the AWS S3 credential that was in the code. This credential was created using Vault's AWS secrets engine and the developer received the following output when requesting a credential from Vault.

```
Key                Value
---                -----
lease_id           aws/creds/s3-access/f3e92392-7d9c-09c8-c921-575d62fe80d8
lease_duration     768h
lease_renewable    true
access_key         AKIAIOWQXTLW36DV7IEA
secret_key         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
```

Which Vault command will revoke the lease and remove the credential from AWS?

Options:

A- vault lease revoke aws/creds/s3-access/f3e92392-7d9c-99c8-c921-57Sd62fe89d8

B- vault lease revoke AKIAI0WQXTLW36DV7IEA

C- vault lease revoke f3e92392-7d9c-O9c8-c921-575d62fe80d8

D- vault lease revoke access_key-AKIAI0WQXTLW36DV7IEA

Answer:

A

Explanation:

The correct answer is A because the lease ID is the unique identifier for the credential. The lease ID is used to revoke the credential using the vault lease revoke command. This command will invalidate the credential immediately and prevent any further renewals.It will also delete the access key and secret key from AWS, rendering them useless1. The access key and secret key are not sufficient to revoke the credential, as they are not recognized by Vault. The lease ID is composed of the path of the secrets engine, the role name, and a random UUID. In this case, the path is aws/creds, the role name is s3-access, and the UUID is f3e92392-7d9c-99c8-c921-57Sd62fe89d8.

lease revoke - Command | Vault | HashiCorp Developer

Question 4

Question Type: MultipleChoice

When looking at Vault token details, which key helps you find the paths the token is able to access?

Options:

A- Meta

B- Path

C- Policies

D- Accessor

Answer:

C

Explanation:

When looking at Vault token details, the policies key helps you find the paths the token is able to access. Policies are a declarative way to grant or forbid access to certain paths and operations in Vault. Policies are written in HCL or JSON and are attached to tokens by name. Policies are deny by default, so an empty policy grants no permission in the system. A token can have one or more policies associated with it, and the effective policy is the union of all the individual policies. You can view the token details by using the vault token lookup command or the auth/token/lookup API endpoint. The output will show the policies key with a list of policy names that are attached to the token. You can also view the contents of a policy by using the vault policy read command or the sys/policy API endpoint. The output will show the rules key with the HCL or JSON representation of the policy.The rules will specify the paths and the capabilities (such as create, read, update, delete, list, etc.) that the policy allows or denies.Reference: https://developer.hashicorp.com/vault/docs/concepts/policies4, https://developer.hashicorp.com/vault/docs/commands/token/lookup5,

https://developer.hashicorp.com/vault/api-docs/auth/token#lookup-a-token6,
https://developer.hashicorp.com/vault/docs/commands/policy/read7, https://developer.hashicorp.com/vault/api-docs/system/policy8

Question 5

Question Type: MultipleChoice

You are performing a high number of authentications in a short amount of time. You're experiencing slow throughput for token generation. How would you solve this problem?

Options:

A- Increase the time-to-live on service tokens

B- Implement batch tokens

C- Establish a rate limit quota

D- Reduce the number of policies attached to the tokens

Answer:

B

Explanation:

Batch tokens are a type of tokens that are not persisted in Vault's storage backend, but are encrypted blobs that carry enough information to perform Vault actions. Batch tokens are extremely lightweight and scalable, and can improve the throughput for token generation. Batch tokens are suitable for high-volume and ephemeral workloads, such as containers or serverless functions, that require short-lived and non-renewable tokens. Batch tokens can be created by using the -type=batch flag in the vault token create command, or by configuring the token_type parameter in the auth method's role or mount options. Batch tokens have some limitations compared to service tokens, such as the lack of renewal, revocation, listing, accessor, and cubbyhole features.Therefore, batch tokens should be used with caution and only when the trade-offs are acceptable.Reference: https://developer.hashicorp.com/vault/tutorials/tokens/batch-tokens1, https://developer.hashicorp.com/vault/docs/commands/token/create2, https://developer.hashicorp.com/vault/docs/concepts/tokens#token-types3

Question 6

Question Type: MultipleChoice

The key/value v2 secrets engine is enabled at secret/ See the following policy:

```
path "secret/data/*" {
  capabilities = ["create", "read", "update", "delete", "list"]
}

path "secret/data/super-secret" {
  capabilities = ["deny"]
}
```

Which of the following operations are permitted by this policy? Choose two correct answers.

Options:

A- vault kv get secret/webapp1

B- vault kv put secret/webapp1 apikey-'ABCDEFGHI] K123M'

C- vault kv metadata get secret/webapp1

D- vault kv delete secret/super-secret

E- vault kv list secret/super-secret

Answer:

A, C

Explanation:

The policy shown in the image is:

path "secret/data/webapp1" { capabilities = ["create", "read", "update", "delete", "list"] }

path "secret/data/super-secret" { capabilities = ["deny"] }

This policy grants or denies access to the key/value v2 secrets engine mounted at secret/ according to the following rules:

The path "secret/data/webapp1" has the capabilities of "create", "read", "update", "delete", and "list". This means that the policy allows performing any of these operations on the secrets stored under this path.The data/ prefix is used to access the actual secret data in the key/value v2 secrets engine5.Therefore, the policy permits the operation of vault kv get secret/webapp1, which reads the secret data at secret/data/webapp16.

The path "secret/data/super-secret" has the capability of "deny". This means that the policy denies performing any operation on the secrets stored under this path. The policy overrides any other policy that might grant access to this path.Therefore, the policy does not permit the operations of vault kv delete secret/super-secret and vault kv list secret/super-secret, which delete and list the secret data at secret/data/super-secret respectively6.

The policy does not explicitly define any rules for the path "secret/metadata".The metadata/ prefix is used to access the metadata of the secrets in the key/value v2 secrets engine, such as the number of versions, the deletion status, the creation time, etc5.By default, if the policy grants any of the capabilities of "create", "read", "update", or "delete" on the data/ path, it also grants the same capabilities on the corresponding metadata/ path7.Therefore, the policy permits the operation of vault kv metadata get secret/webapp1, which reads the metadata of the secret at secret/metadata/webapp18.

Question 7

Question Type: MultipleChoice

Examine the command below. Output has been trimmed.

```
  $ vault write auth/approle/login \
    role_id="debb8f13-79ea-3e3d-8100-10711d85c1fb" \
    secret_id="31d52faa-5b0b-711d-2ea2-c197cff6081b"Key Value
---                   -----
token                 b.AAAAAQIlWH-DExezQvz-ZGWMhzsy8uWXEoQYHH6O...trimmed...
token_accessor        n/a
token_duration        1m
token_renewable       false
token_policies        ["shipping"]
identity_policies     []
policies              ["shipping"]
token_meta_role_name  shipping
```

Which of the following statements describe the command and its output?

Options:

A- Missing a default token policy

B- Generated token's TTL is 60 hours

C- Generated token is an orphan token which can be renewed indefinitely

D- Configures the AppRole auth method with user specified role ID and secret ID

Answer:

B, C

Explanation:

The command shown in the image is:

vault token create -policy=approle -orphan -period=60h

This command creates a new token with the following characteristics:

It has the policy "approle" attached to it, which grants or denies access to certain paths and operations in Vault according to the policy rules.The policy can be defined by using the vault policy write command or the sys/policy API endpoint12.

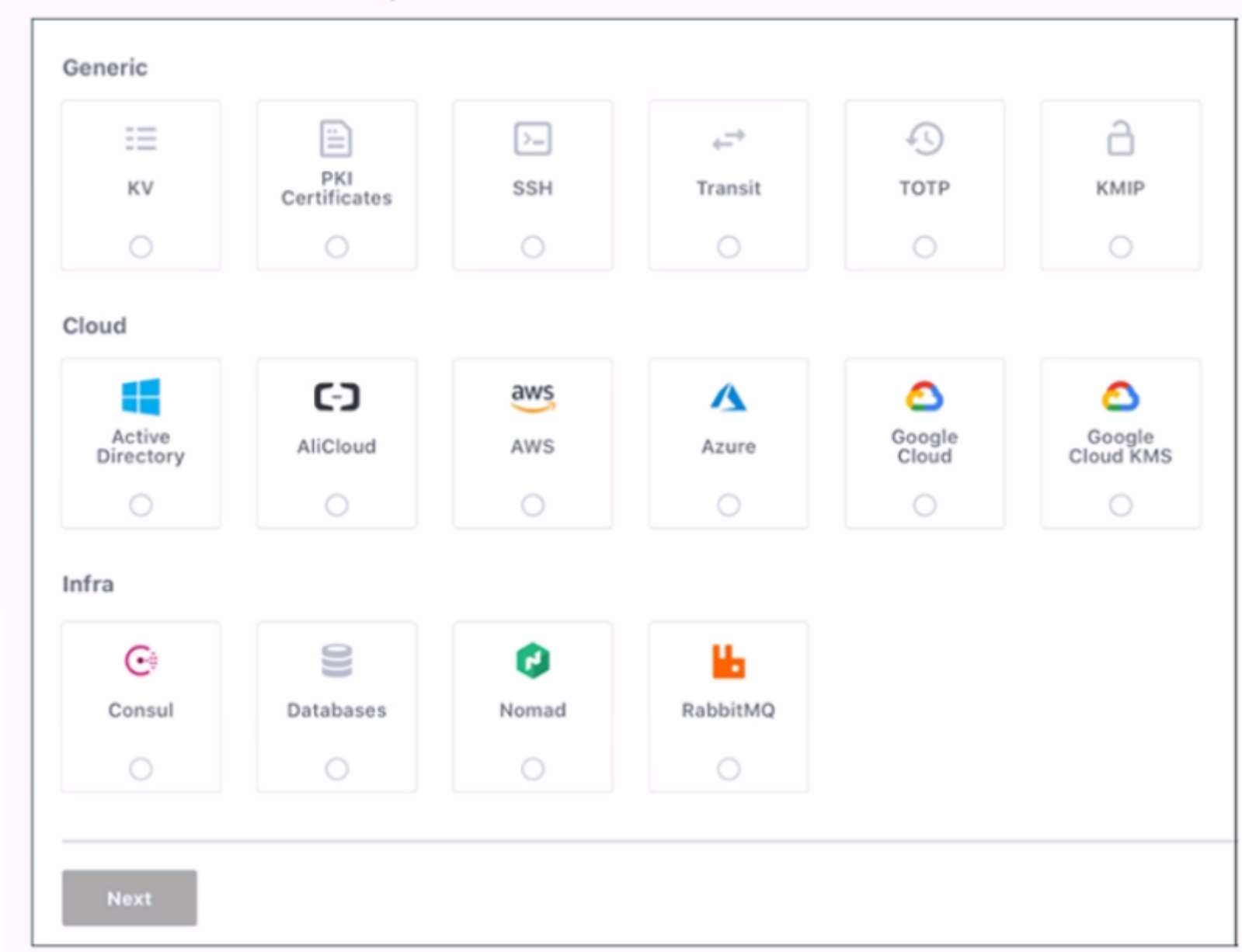
It is an orphan token, which means it has no parent token and it will not be revoked when its parent token is revoked.Orphan tokens can be useful for creating long-lived tokens that are not affected by the token hierarchy3.

It has a period of 60 hours, which means it has a renewable TTL of 60 hours. This means that the token can be renewed indefinitely as long as it does not go past the 60-hour mark from the last renewal time. The token's TTL will be reset to 60 hours upon each renewal.Periodic tokens are useful for creating tokens that have a fixed lifetime and can be easily revoked4.

Question 8

Question Type: MultipleChoice

Use this screenshot to answer the question below:


Generic
KV
PKI Certificates
SSH
Transit
TOTP
KMIP
Cloud
Active Directory
AliCloud
AWS
Azure
Google Cloud
Google Cloud KMS
Infra
Consul
Databases
Nomad
RabbitMQ
Next

When are you shown these options in the GUI?

Options:

A- Enabling policies

B- Enabling authentication engines

C- Enabling secret engines

D- Enabling authentication methods

Answer:

D

Explanation:

This screenshot is shown when you are enabling authentication methods in the GUI. Authentication methods are the ways users and applications authenticate with Vault. Vault supports many different authentication methods, including username and password, GitHub, and more. You can enable one or more authentication methods from the grid of options, which are divided into three categories: Generic, Cloud, and Infra. Each option has a name, a description, and a logo. You can also enable authentication methods using the Vault CLI or API.

Enabling policies, authentication engines, and secret engines are different tasks that are not related to this screenshot. Policies are rules that govern the access to Vault resources, such as secrets, authentication methods, and audit devices. Authentication engines are

components of Vault that perform authentication and assign policies to authenticated entities. Secret engines are components of Vault that store, generate, or encrypt data. These tasks have different GUI pages and options than the screenshot.

[Authentication | Vault | HashiCorp Developer]

[Policies | Vault | HashiCorp Developer]

[Authentication | Vault | HashiCorp Developer]

[Secrets Engines | Vault | HashiCorp Developer]

Question 9

Question Type: **MultipleChoice**

Which of the following are replication methods available in Vault Enterprise? Choose two correct answers.

Options:

A- Cluster sharding

B- Namespaces

C- Performance Replication

D- Disaster Recovery Replication

Answer:

C, D

Explanation:

The replication methods available in Vault Enterprise are performance replication and disaster recovery replication. These methods allow critical data to be replicated across clusters to support horizontally scaling and disaster recovery workloads.

Performance replication enables a primary cluster to replicate data to one or more secondary clusters, which can handle client requests and improve performance and availability. Performance replication replicates most Vault data, such as secrets, policies, auth methods, and leases, but not tokens. Performance secondaries generate their own tokens and leases, which are not replicated back to the primary. Performance replication also supports filtering, which allows selective replication of data based on namespaces or paths.

Disaster recovery replication enables a primary cluster to replicate data to one or more secondary clusters, which act as standby clusters in case of a failure or outage of the primary. Disaster recovery replication replicates all Vault data, including tokens and leases, and maintains the same configuration and state as the primary. Disaster recovery secondaries do not handle client requests, but they can be promoted to a primary in a disaster recovery scenario.Reference:Replication - Vault Enterprise | Vault | HashiCorp Developer,Performance Replication - Vault Enterprise | Vault | HashiCorp Developer,Disaster Recovery Replication - Vault Enterprise | Vault | HashiCorp Developer