PassQueen
a.k.a dumpsqueen.com

HashiCorp Certified: Vault Associate (002)

HashiCorp Vault-Associate

Version Demo

Total Demo Questions: 7

Total Premium Questions: 77

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Vault Architecture	13
Topic 2, Authentication Methods	9
Topic 3, Vault Policies	12
Topic 4, Secrets Engines	16
Topic 5, Tokens	13
Topic 6, Leases	4
Topic 7, Vault CLI	5
Topic 8, Vault API	1
Topic 9, Vault Agent	3
Topic 10, Mix Questions	1
Total	77

QUESTION NO: 1

A security team needs to determine whether a token can read a secret and list available secret names in a KV version 2 mount at `secret/`. Which policy path and capability combinations are appropriate? Choose two correct answers.

A. `path "secret/data/apps/*" { capabilities = ["read"] }`

B. `path "secret/data/apps/*" { capabilities = ["list"] }`

C. `path "secret/metadata/apps/*" { capabilities = ["list"] }`

D. `path "secret/metadata/apps/*" { capabilities = ["read"] }`

E. `path "secret/apps/*" { capabilities = ["sudo"] }`

ANSWER: A C

Explanation:

For KV version 2, reading secret data requires the `read` capability on the `data/` API path. Listing keys requires the `list` capability on the `metadata/` API path. These are separate permissions because listing names does not require permission to read the secret values.

Applying `list` to the data path does not grant listing access for KV v2, and applying `read` to the metadata path does not grant access to the stored secret value. A policy should grant only the operations the client requires.

QUESTION NO: 2

An operator has a policy that grants the `update` capability on `sys/mounts/team-kv`. The operator still receives permission denied when attempting to enable a secrets engine at that path because the endpoint is root protected. What policy change is required?

A. Add the sudo capability to sys/mounts/team-kv.

B. Replace update with read on sys/mounts/team-kv.

C. Add the list capability to secret/data/team-kv/*.

D. Attach the root policy to the operator token.

ANSWER: A

Explanation:

The policy must also grant the `sudo` capability on the root-protected path. Vault uses `sudo` as an additional authorization requirement for selected sensitive system endpoints. It does not replace the normal operation capability; enabling a mount requires the appropriate write capability as well as `sudo`.

Granting `read` or `list` would not authorize the update operation. Assigning the root policy would work but would violate least privilege because it grants unrestricted access rather than only the required capability for the specific mount path.

QUESTION NO: 3

Vault supports which type of configuration for source limited token?

A. Cloud-bound tokens

B. Domain-bound tokens

C. CIDR-bound tokens

D. Certificate-bound tokens

ANSWER: C

Explanation:

Vault supports **CIDR-bound tokens** to restrict where a token may be used based on the client source IP address. A CIDR, or Classless Inter-Domain Routing range, represents either an individual IP address or a network range. When a token has CIDR restrictions, Vault validates the IP address of the request against its configured bound CIDR list. The token is accepted only when the request originates from an allowed address or range, providing an additional security boundary if a token is inadvertently exposed.

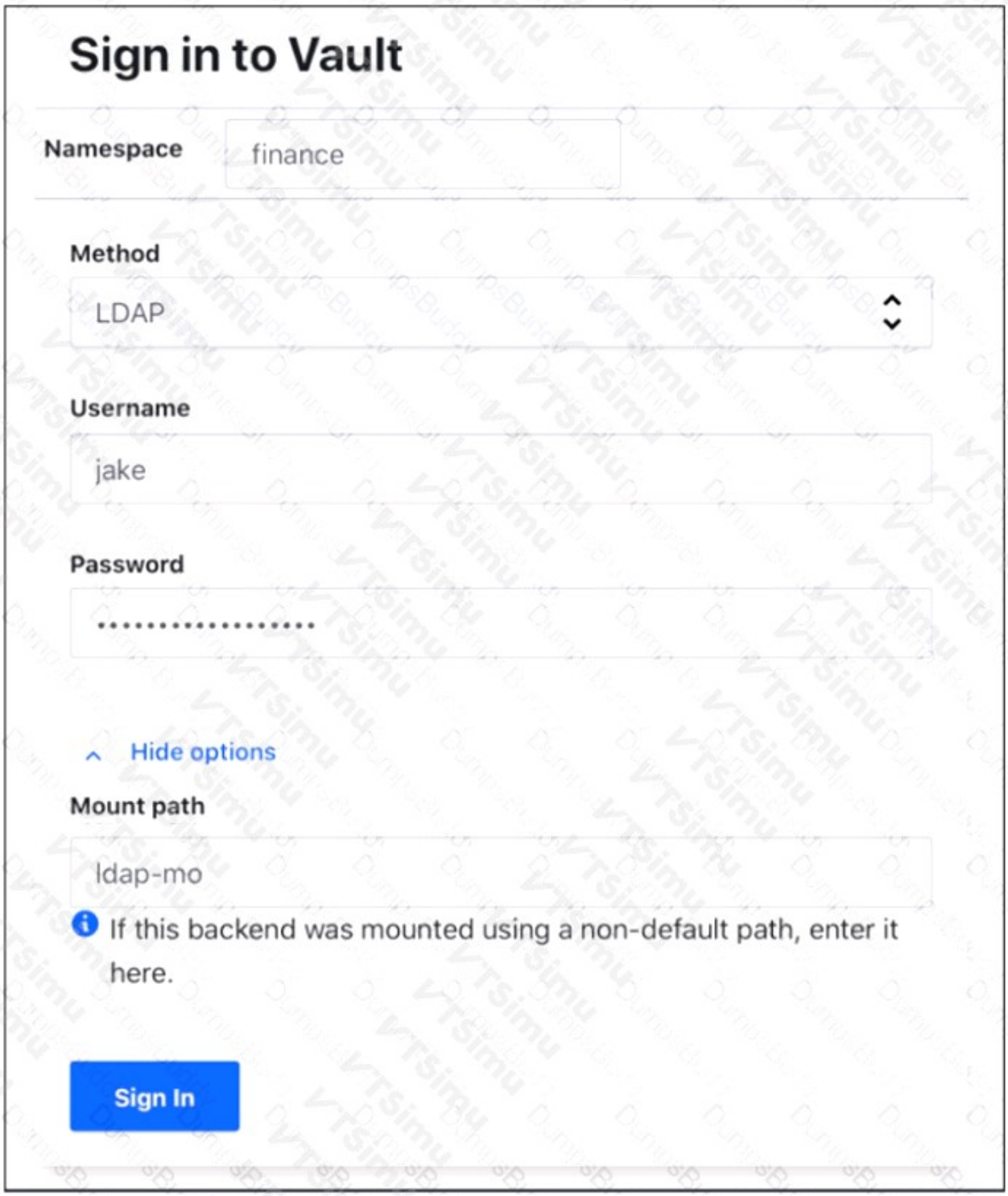
CIDR restrictions can be applied directly when creating a token with the `-bound-cidr` flag, or configured through token roles using the `bound_cidr_list` parameter. This control is particularly useful for limiting privileged tokens to known administrative networks, private application subnets, or tightly scoped automation environments. The restriction is enforced at token use time, rather than merely serving as descriptive metadata. HashiCorp documents the token creation option and the token role CIDR binding setting in its command and API documentation: vault token create and Token auth method API.

QUESTION NO: 4 - (SIMULATION)

Where do you define the Namespace to log into using the Vault UI?

To answer this question

Use your mouse to click on the screenshot in the location described above. An arrow indicator will mark where you have clicked. Click the "Answer" button once you have positioned the arrow to answer the question. You may need to scroll down to see the entire screenshot.



ANSWER: See the explanation for the answer

Explanation:

Click the **Namespace** input field at the top of the Vault sign-in form (the field currently showing `finance`), approximately near coordinates `(290, 112)` in the screenshot.

The Namespace is defined directly in this dedicated **Namespace** field—not in the Advanced Options **Mount path** field, which specifies the authentication method mount path.

QUESTION NO: 5

An application obtains database credentials from the database secrets engine. Which statements are true about the generated credentials? Choose two correct answers.

A. They can be issued with a lease and a time-to-live.

B. They can be revoked through the configured database plugin.

C. They are always stored as static values in the KV secrets engine.

D. They allow the application to authenticate to Vault without a token.

E. They remain valid until the database administrator manually changes them.

ANSWER: A B

Explanation:

The database secrets engine can generate credentials dynamically according to a configured role. The response normally includes a lease, which gives the credentials a controlled lifetime and lets Vault track their lifecycle.

When a dynamic database credential is revoked or expires, Vault invokes the configured database revocation behavior to remove or invalidate the corresponding database access. This is different from storing a shared database password in KV, where Vault cannot automatically create a unique database principal or revoke it at the database based on a lease.

QUESTION NO: 6

An organization would like to use a scheduler to track & revoke access granted to a job (by Vault) at completion. What auth-associated Vault object should be tracked to enable this behavior?

A. Token accessor

B. Token ID

C. Lease ID

D. Authentication method

ANSWER: A

Explanation:

Token accessor is correct because it is the non-secret management identifier Vault assigns to a token. A scheduler can retain the accessor when a job receives its Vault token, then use that accessor at job completion to look up, renew where appropriate, or revoke the associated token without storing or exposing the token value itself. Revoking the token immediately removes the job's ability to make authenticated requests to Vault, which is the appropriate lifecycle action when the job has finished.

This design supports least privilege and safer automation: the job uses its token to authenticate during execution, while the scheduler or control plane tracks only the accessor for administrative lifecycle operations. Vault's token API supports revocation using an accessor through the `auth/token/revoke-accessor` endpoint. HashiCorp documents that accessors identify tokens without granting the ability to authenticate as those tokens, making them suitable for auditing and token management workflows. See the Vault token accessor documentation and the Token API documentation for revoking an accessor.

QUESTION NO: 7

An administrator enables the LDAP auth method at both `ldap-engineering/` and `ldap-contractors/`. Which statements are true? Choose two correct answers.

A. The LDAP auth method can be enabled at both paths.

B. Roles configured at one mount are independent of roles at the other mount.

C. Both mounts must issue tokens with the same policies.

D. Enabling the second mount disables the first LDAP mount.

E. LDAP auth methods can be enabled only at the default ldap/ path.

ANSWER: A B

Explanation:

Vault auth methods are mount-based, so the same auth method type can be enabled at more than one path. Separate mounts allow independent configuration for different populations or use cases.

Roles, configuration, and mappings are scoped to their auth mount. Therefore, a role configured at `ldap-engineering/` is not automatically available through `ldap-contractors/`. This separation allows the administrator to apply different LDAP settings, policies, token settings, or group mappings at each mount.