VA-002-P Q&As

HashiCorp Certified: Vault Associate

Pass HashiCorp VA-002-P Exam with 100% Guarantee

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

https://www.leads4pass.com/va-002-p.html

100% Passing Guarantee
100% Money Back Assurance

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

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

QUESTION 1

What feature of Terraform Cloud and/or Terraform Enterprise can you publish and maintain a set of custom modules which can be used within your organization?

A. custom VCS integration

B. remote runs

C. private module registry

D. Terraform registry

Correct Answer: C

You can use modules from a private registry, like the one provided by Terraform Cloud. Private registry

modules have source strings of the form

///. This is the same format as the public registry,

but with an added hostname prefix.

QUESTION 2

While Terraform is generally written using the HashiCorp Configuration Language (HCL), what another syntax can Terraform be expressed in?

A. JSON

B. XML

C. TypeScript

D. YAML

Correct Answer: A

The constructs in the Terraform language can also be expressed in JSON syntax, which is harder for humans to read and edit but easier to generate and parse programmatically.

QUESTION 3

What command is used to renew a token, if permitted?

A. vault operator token renew

B. vault token update

C. vault new

D. vault update token

E. vault token renew

F. vault renew token

Correct Answer: E

In order to renew a token, a user can issue a vault token renew command to extend the TTL. The token can also be renewed using the API

QUESTION 4

True or False: When encrypting data with the transit secrets engine, Vault always stores the ciphertext in a dedicated KV store along with the associated encryption key.

A. False

B. True

Correct Answer: A

Vault doesn\\'t store the data sent to the secrets engine. The transit secrets engine handles cryptographic functions on data-in-transit. It can also be viewed as "cryptography as a service" or "encryption as a service". The transit secrets engine can also sign and verify data; generate hashes and HMACs of data; and act as a source of random bytes. Reference link:- https://www.vaultproject.io/docs/secrets/transit

QUESTION 5

An administrator wants to create a new KV mount for individual users to maintain their own secrets but needs a way to simplify the policy so they don\\'t need to write a new one for each new user? With the requirements listed below, what would such a policy look like? Requirement: Each user can perform all operations on their allocated key/value secret path

A. path "user-kv/data/{{identity.entity.name}}/*" { capabilities = ["create", "update", "read", "delete", "list"] }

B. path "user-kv/data/{{identity.entity.id.name}}/*" { capabilities = ["create", "update", "read", "delete", "list"] }

C. path "user-kv/data/{{identity.entity.aliases..id}}/*" { capabilities = ["create", "update", "read", "delete", "list"] }

D. path "user-kv/data/{{user}}/*" { capabilities = ["create", "update", "read", "delete", "list"] }

Correct Answer: A

Everything in the Vault is path-based, and policies are no exception. Policies provide a declarative way to grant or forbid access to certain paths and operations in Vault. The policy template makes it very flexible to customize the environment. By using parameters within your template, you can have Vault "insert" a value into the path based upon things like identity values, group membership, and metadata associated with either the user\\'s identity or group they are a member of. Using the parameter, the path user-kv/data/{{identity.entity.name}}/* converts to user- kv/data/student01/*

QUESTION 6

What does the command terraform fmt do?

A. formats the state file in order to ensure the latest state of resources can be obtained

B. updates the font of the configuration file to the official font supported by HashiCorp

C. rewrite Terraform configuration files to a canonical format and style

D. deletes the existing configuration file

Correct Answer: C

The terraform fmt command is used to rewrite Terraform configuration files to a canonical format and style.

This command applies a subset of the Terraform language style conventions, along with other minor

adjustments for readability.

Other Terraform commands that generate Terraform configuration will produce configuration files that

conform to the style imposed by terraform fmt, so using this style in your own files will ensure consistency.

QUESTION 7

By default, where does Terraform store its state file?

A. shared directory

B. current working directory

C. Amazon S3 bucket D. remotely using Terraform Cloud

Correct Answer: B

By default, the state file is stored in a local file named "terraform.tfstate", but it can also be stored remotely, which works better in a team environment.

QUESTION 8

True or False:

When using the transit secrets engine, setting the min_decryption_version will determine the minimum key

length of the data key (i.e., 2048, 4096, etc.)

A. False

B. True

Correct Answer: A

The Transit engine supports the versioning of keys. Key versions that are earlier than a key\\'s specified min_decryption_version gets archived, and the rest of the key versions belong to the working set. This is a performance

consideration to keep key loading fast, as well as a security consideration: by disallowing decryption of old versions of keys, found ciphertext corresponding to obsolete (but sensitive) data can not be decrypted by most users, but in an emergency, the min_decryption_version can be moved back to allow for legitimate decryption. Reference link:- https://www.vaultproject.io/docs/secrets/transit

QUESTION 9

Which command is used to initialize Vault after first starting the Vault service?

A. vault create key

B. vault operator init

C. vault operator initialize keys

D. vault start

E. vault operator unseal

Correct Answer: B

The vault operator init command initializes a Vault server. Initialization is the process by which Vault\\'s

storage backend is prepared to receive data.

This only happens once when the server is started against a new backend that has never been used with

Vault before.

Reference link is below:- https://www.vaultproject.io/docs/commands/operator/init

QUESTION 10

Unsealing Vault creates the encryption keys, which is used to unencrypt the data on the storage backend.

A. FALSE

B. TRUE

Correct Answer: A

Unsealing is the process of obtaining the plaintext master key necessary to read the decryption key to decrypt the data, allowing access to the Vault. The master key is used to decrypt the encryption key which can unencrypt the data on the storage backend.

QUESTION 11

Which of the following Vault features is available only in the Enterprise version? (select three)

A. MFA

B. dynamic credentials

C. cloud auto unseal

D. replication

E. auto unseal with HSM

Correct Answer: ADE

Most of the important features of Vault are available in the open-source version, however, some of the features which are generally required by large organizations are only available in the Enterprise version such as:

-MFA - Multi-factor Authentication

-Replication

-Auto unseal with HSM and many more.

Check all the features at the below link.

Reference link:- https://www.hashicorp.com/products/vault/pricing/

QUESTION 12

Which of the following storage backends are supported by HashiCorp technical support? (select four)

A. Filesystem

B. Consul

C. In-Memory

D. Raft

E. DynamoDB

F. MySQL

Correct Answer: ABCD

Just to clarify, "HashiCorp supported" means, it is supported by HashiCorp\\'s technical support, it doesn\\'t mean that Vault supports the platform as a storage backend. For example, DynamoDB is a valid storage backend, but it is not officially supported by HashiCorp technical support but it has got the community support. In-Memory - HashiCorp Supported MySQL - Community Supported Raft - HashiCorp Supported Dynamo DB - Community Supported Consul - HashiCorp Supported Filesystem - HashiCorp Supported Check more details on below link:- https://www.vaultproject.io/docs/configuration/storage/in- memory

QUESTION 13

True or False? Each Terraform workspace uses its own state file to manage the infrastructure associated with that particular workspace.

A. False

B. True

Correct Answer: B

The persistent data stored in the backend belongs to a workspace. Initially, the backend has only one workspace, called "default", and thus there is only one Terraform state associated with that configuration.

QUESTION 14

When multiple arguments with single-line values appear on consecutive lines at the same nesting level, HashiCorp recommends that you:

A. place a space in between each line type = "A" ttl = "300" zone_id = aws_route53_zone.primary.zone_id

B. align their equals signs ami = "abc123" instance_type = "t2.micro"

C. place all arguments using a variable at the top ami = var.aws_ami instance_type = var.instance_size subnet_id = "subnet-0bb1c79de3EXAMPLE" tags = { Name = "HelloWorld" }

D. put arguments in alphabetical order name = "www.pythonfanclub.com" records = [aws_eip.lb.public_ip] type = "A" ttl = "300" zone_id = aws_route53_zone.primary.zone_id

Correct Answer: B

HashiCorp style conventions suggest you that align the equals sign for consecutive arguments for easing

readability for configurations

ami = "abc123"

instance_type = "t2.micro"

QUESTION 15

In regards to Terraform state file, select all the statements below which are correct: (select four)

A. storing state remotely can provide better security

B. the Terraform state can contain sensitive data, therefore the state file should be protected from unauthorized access

C. Terraform Cloud always encrypts state at rest

D. using the mask feature, you can instruct Terraform to mask sensitive data in the state file

E. when using local state, the state file is stored in plain-text

F. the state file is always encrypted at rest

Correct Answer: ABCE

Terraform state can contain sensitive data, depending on the resources in use and your definition of "sensitive." The

state contains resource IDs and all resource attributes. For resources such as databases, this may contain initial passwords. When using local state, state is stored in plain-text JSON files. If you manage any sensitive data with Terraform (like database passwords, user passwords, or private keys), treat the state itself as sensitive data. Storing Terraform state remotely can provide better security. As of Terraform 0.9, Terraform does not persist state to the local disk when remote state is in use, and some backends can be configured to encrypt the state data at rest.