



Free Questions for VA-002-P by certscare

Shared by Bowers on 24-05-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

After running into issues with Terraform, you need to enable verbose logging to assist with troubleshooting the error. Which of the following values provides the MOST verbose logging?

Options:

A- ERROR

B- INFO

C- DEBUG

D- WARN

E- TRACE

Answer:

E

Explanation:

Terraform has detailed logs that can be enabled by setting the TF_LOG environment variable to any value. This will cause detailed logs to appear on stderr.

You can set TF_LOG to one of the log levels TRACE, DEBUG, INFO, WARN, or ERROR to change the verbosity of the logs. TRACE is the most verbose and it is the default if TF_LOG is set to something other than a log level name.

Question 2

Question Type: MultipleChoice

True or False: Provisioners should only be used as a last resort.

Options:

A- true

B- false

Answer:

A

Explanation:

Provisioners are used to execute scripts on a local or remote machine as part of resource creation or destruction. Provisioners can be used to bootstrap a resource, cleanup before destroy, run configuration management, etc. Even if the functionality you need is not available in a provider today, HashiCorp suggests that you consider local-exec usage as a temporary workaround and to open an issue in the relevant provider's repo to discuss adding first-class support.

Question 3

Question Type: MultipleChoice

True or False:

A list(...) may contain a number of values of the same type while an object(...) can contain a number of values of different types.

Options:

A- True

B- False

Answer:

A

Explanation:

A collection type allows multiple values of one other type to be grouped together as a single value. This includes a list, map, and set.

A structural type allows multiple values of several distinct types to be grouped together as a single value. This includes object and tuple.

Question 4

Question Type: MultipleChoice

What are some of the problems of how infrastructure was traditionally managed before Infrastructure as Code? (select three)

Options:

A- Requests for infrastructure or hardware required a ticket, increasing the time required to deploy applications

B- Traditional deployment methods are not able to meet the demands of the modern business where resources tend to live days to weeks, rather than months to years

C- Traditionally managed infrastructure can't keep up with cyclic or elastic applications

D- Pointing and clicking in a management console is a scalable approach and reduces human error as businesses are moving to a multi-cloud deployment model

Answer:

A, B, C

Explanation:

Businesses are making a transition where traditionally-managed infrastructure can no longer meet the demands of today's businesses. IT organizations are quickly adopting the public cloud, which is predominantly API-driven.

To meet customer demands and save costs, application teams are architecting their applications to support a much higher level of elasticity, supporting technology like containers and public cloud resources. These resources may only live for a matter of hours; therefore the traditional method of raising a ticket to request resources is no longer a viable option. Pointing and clicking in a management console is NOT scale and increases the change of human error.

Question 5

Question Type: MultipleChoice

What are the benefits of using Infrastructure as Code? (select five)

Options:

- A-** Infrastructure as Code easily replaces development languages such as Go and .Net for application development
- B-** Infrastructure as Code allows a user to turn a manual task into a simple, automated deployment
- C-** Infrastructure as Code is relatively simple to learn and write, regardless of a user's prior experience with developing code
- D-** Infrastructure as Code is easily repeatable, allowing the user to reuse code to deploy similar, yet different resources
- E-** Infrastructure as Code provides configuration consistency and standardization among deployments
- F-** Infrastructure as Code gives the user the ability to recreate an application's infrastructure for disaster recovery scenarios

Answer:

B, C, D, E, F

Explanation:

If you are new to infrastructure as code as a concept, it is the process of managing infrastructure in a file or files rather than manually configuring resources in a user interface. A resource in this instance is any piece of infrastructure in a given environment, such as a virtual machine, security group, network interface, etc.

At a high level, Terraform allows operators to use HCL to author files containing definitions of their desired resources on almost any provider (AWS, GCP, GitHub, Docker, etc) and automates the creation of those resources at the time of application.

Question 6

Question Type: MultipleChoice

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

Options:

- 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

Answer:

A, B, C, E

Explanation:

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.

Question 7

Question Type: MultipleChoice

What Terraform feature is shown in the example below?

```
1. resource "aws_security_group" "example" {  
2. name = "sg-app-web-01"  
3. dynamic "ingress" {  
4. for_each = var.service_ports  
5. content {  
6. from_port = ingress.value  
7. to_port = ingress.value  
8. protocol = "tcp"  
9. }  
10. }  
11. }
```

Options:

A- data source

B- dynamic block

C- local values

D- conditional expression

Answer:

B

Explanation:

You can dynamically construct repeatable nested blocks like ingress using a special dynamic block type, which is supported inside resource, data, provider, and provisioner blocks

To Get Premium Files for VA-002-P Visit

<https://www.p2pexams.com/products/va-002-p>

For More Free Questions Visit

<https://www.p2pexams.com/hashicorp/pdf/va-002-p>

