

VMware 3V0-21.25

Advanced VMware Cloud Foundation 9.0 Automation

For More Information – Visit link below:

<https://www.examsempire.com/>

Product Version

- 1. Up to Date products, reliable and verified.**
- 2. Questions and Answers in PDF Format.**



<https://examsempire.com/>

Visit us at: <https://www.examsempire.com/3v0-21-25>

Latest Version: 6.1

Question: 1

A VMware Cloud Foundation (VCF) Automation Administrator is tasked to enable VCF Automation with the following requirements:

- All companies are hosted within a single private cloud.
- RBAC (role-based access control) is enforced.
- Resource governance within companies.
- Segregation between companies.

What two actions must the VCF Automation Administrator perform to satisfy the requirements? (Choose two.)

- A. Deploy a vCenter instance with a Supervisor cluster per company.
- B. Ensure that the vCenter instance has a Supervisor cluster enabled.
- C. Deploy a VCF Operations Orchestrator server to enable multi-tenancy.
- D. Create and configure an AllApps Organization per company.
- E. Create and configure a VMApps Organization per company.

Answer: B, E

Explanation:

In VMware Cloud Foundation 9.0, multi-tenancy is structured around the concept of Organizations. To meet the requirement of hosting multiple companies within a single private cloud with strict segregation and governance, the administrator must utilize the VMApps Organization model. Unlike the AllApps model, which is highly optimized for modern containerized and VPC-driven workloads, the VMApps Organization is specifically designed for environments requiring traditional VM-centric segregation and access control while sharing underlying physical infrastructure. Enabling a Supervisor cluster on the vCenter instance is a foundational prerequisite for these advanced automation capabilities. The Supervisor provides the necessary integration between the vSphere compute layer and the VCF Automation control plane, allowing for the instantiation of the Namespace and Organization constructs that enforce RBAC and resource quotas. By configuring a VMApps Organization per company, the administrator ensures that each tenant has a distinct administrative boundary, private catalog, and isolated resource allocation, effectively satisfying the requirements for hard tenancy within a consolidated private cloud environment.

Question: 2

An administrator is responsible for managing a VMware Cloud Foundation (VCF) fleet and the administrator has been tasked with the following:

- Create DNS records before each virtual machine (VM) is deployed using VCF Automation. The administrator has already completed the following tasks:
- Created two VCF Operations Orchestrator Workflows with corresponding Event Subscriptions:
Create DNS Record
Delete DNS Record
- Created a new blueprint to deploy a VM:
Added two string inputs, hostname and domainName
Added hostname: '\${input.hostname}' as a custom property of the Virtual Machine resource.
Added domainName: '\${input.domainName}' as a custom property of the Virtual Machine resource.

What should the administrator configure within the Event subscription to ensure that the DNS record is only created when the hostname is provided?

- A. Add the Delete DNS Record workflow as the Recovery Workflow of the Create DNS Record subscription.
- B. Add the event.data.customproperties['domainName'] != null condition to the Create DNS Record and Delete DNS Record subscriptions.
- C. Add the event.data.customproperties['hostname'] != null condition to the Create DNS Record and Delete DNS Record subscriptions.
- D. Enable the Block execution of events in topic option in the Create DNS Record and Delete DNS Record subscriptions.

Answer: C

Explanation:

VCF Automation 9.0 utilizes an Event Broker Service (EBS) to trigger extensibility workflows during the lifecycle of a deployment. For a DNS integration to function correctly and reliably, the event subscription must be "scoped" to prevent it from firing when essential metadata is missing. In this scenario, the administrator has mapped the user input hostname to a custom property of the virtual machine. By adding the condition event.data.customproperties['hostname'] != null to the subscription, the platform evaluates the payload before invoking the Operations Orchestrator workflow. If the consumer leaves the hostname field empty (assuming it is not marked as mandatory in the blueprint), the condition will evaluate to false, and the DNS creation workflow will not be triggered, preventing "empty" or invalid records from being sent to the DNS provider. This logic must be applied to both the creation and deletion subscriptions to maintain parity throughout the VM's lifecycle. Using the customproperties array within the event.data payload is the standard method for referencing blueprint-specific inputs within the VCF 9.0 extensibility framework.

Question: 3

An administrator is tasked with configuring a multi-tenant environment within the organization's VMware Cloud Foundation (VCF) deployment. Requirements are:

- isolated network spaces for development and testing.
- dedicated resources.

- role-based access control.

Which of the actions satisfies the requirement?

- A. Define and allocate IP blocks and quotas at the organization level, which will then be selected and consumed by users at deployment.
- B. Configure firewall rules at the datacenter edge level to isolate traffic between departments.
- C. Create an NSX backed VLAN segment per department.
- D. Define and allocate IP blocks and quotas at the organization level, which will then be consumed by the NSX VPCs created by the different departments.
- E. Grant full administrative access for one responsible user in each department to create and manage their own network configurations.

Answer: D

Explanation:

The introduction of Virtual Private Clouds (VPCs) in VCF 9.0 is the primary mechanism for achieving isolated networking in a multi-tenant environment. The correct administrative workflow involves the Provider or Organization Administrator defining IP Blocks and Quotas at the Organization level. These resource allocations act as a "bucket" of capacity from which individual departments or projects can draw. When a department creates an NSX VPC, it automatically consumes these pre-allocated IP ranges for its internal segments and services, ensuring that development and testing environments remain logically isolated at the network layer without requiring manual intervention for every subnet creation. This "consumption-based" model ensures that while departments have the flexibility to manage their own VPC resources, they remain constrained by the governance and RBAC policies defined at the higher Organization level. This approach significantly reduces the operational overhead compared to traditional VLAN segment management or complex edge firewall rules.

Question: 4

A customer has a requirement to register a resource with an external service during provisioning in a VMApps Organization. The requirements are:

- the registration cannot cause provisioning to fail.
- the registration or failure will only be logged in the external service.

What two constructs satisfy the requirements? (Choose two.)

- A. Cloud-init stanza in the blueprint.
- B. ABX action.
- C. Non-blocking event broker subscription.
- D. Blocking event broker subscription.
- E. Bash script on the template.

Answer: B, C

Explanation:

To satisfy the requirement that a registration task must not impact the success of the overall deployment, a Non-blocking event broker subscription must be used. In VCF Automation 9.0, a "Blocking" subscription pauses the provisioning process until the extensibility task completes, allowing the workflow to fail the deployment if the task returns an error. Conversely, a "Non-blocking" subscription operates asynchronously; the platform fires the event and immediately continues with the VM lifecycle regardless of the task's outcome. An Action-Based Extensibility (ABX) action is the ideal lightweight serverless function to execute this registration logic, as it can be easily configured to run in response to the event trigger without the overhead of a full orchestrator workflow. By combining these two, the administrator ensures that the external registration is attempted, and any successes or failures are handled purely within the context of that action and the external service, fulfilling the customer's logging and failure-tolerance requirements.

Question: 5

An administrator has been tasked to enable developers to utilize Terraform to configure resources within VMware Cloud Foundation (VCF) Automation. The solution must:

- enable developers to configure Content Libraries.
- enable developers to configure Cloud Zones.
- enable developers to create flavor and image mappings.

What solution satisfies the requirements?

- A. Terraform provider for VCF Automation.
- B. System Administrator role.
- C. Organization Administrator role.
- D. Terraform configuration in VCF Automation.

Answer: A

Explanation:

The Terraform provider for VCF Automation is the specific tool designed to allow Infrastructure-as-Code (IaC) workflows to interact with the VCF 9.0 API surface. In VCF 9.0, the provider has been expanded to support the newer Organization and Region-based architecture. By utilizing this provider, developers can declare Content Libraries, Cloud Zones, and Flavor/Image Mappings within their HCL (HashiCorp Configuration Language) files. While specific RBAC roles (like Organization Administrator) are necessary for the credentials used by the Terraform runner, the solution itself is the provider that translates Terraform commands into the correct REST API calls for the VCF Automation engine. This enables a consistent developer experience where infrastructure configuration is versioned in Git and applied programmatically, aligning with modern DevOps practices supported by the VCF 9.0 platform.

Thank You for Trying Our Product

Special 16 USD Discount Coupon: NSZUBG3X

Email: support@examsempire.com

**Check our Customer Testimonials and ratings
available on every product page.**

Visit our website.

<https://examsempire.com/>