Vendor: VMware

Exam Code: 3V0-24.25

Exam Name: Advanced VMware Cloud Foundation 9.0 vSphere Kubernetes Service

Version: DEMO

QUESTION 1
An administrator is upgrading an existing VMware vSphere Kubernetes Service (VKS) cluster and receives the following errors:

```
- kubectl get nodes fails with memcache.go and "server is currently
unable to handle the request"
- couldn't get resource list for stats.antrea.tanzu.vmware.com/v1alpha1
- yaml: mapping values are not allowed in this context
```

The administrator successfully updated the Supervisor, but an attempt to update the VKS cluster failed. Based on the scenario, what is the cause of the problem?

A. The administrator is in the wrong cluster context.
B. The Kubernetes version being upgraded is no longer supported.
C. There was an error pulling the update image from the catalog.
D. The administrator does not have the appropriate permissions to upgrade the cluster.

Answer: A
Explanation:
VCF 9.0 documentation emphasizes that administrators commonly work with multiple Kubernetes contexts (Supervisor context, namespace contexts, and workload-cluster contexts), and must switch to the correct context before running kubectl commands against a specific target. It explains that when you connect, the CLI generates and appends contexts into the kubeconfig, and you then switch between contexts using the context-management commands (for example, listing contexts and using a specific context).
It further describes the correct pattern for interacting with a VKS workload cluster: obtain or generate the workload-cluster kubeconfig/context, then run kubectl (such as kubectl get nodes) against that workload cluster context.
In this scenario, the errors appear while attempting cluster operations after a Supervisor update, which strongly indicates kubectl is not pointed at the intended endpoint/context (for example, still using a Supervisor /namespace context or an incorrect kubeconfig entry). Because VCF explicitly calls out context creation and context switching as required steps to target the correct cluster, the most verified cause among the options is being in the wrong cluster context.

QUESTION 2
What is the function of Contourin a VMware vSphere Kubernetes Service (VKS) cluster?

A. Providing an ingress controller to expose services to external users.
B. Monitoring the health and performance of the underlying infrastructure.
C. Managing the lifecycle and patching of VKS cluster nodes.
D. Providing persistent storage for stateful applications.

Answer: A
Explanation:
In VCF 9.0, ingress is described as part of VKS cluster networking. The documentation's VKS Cluster Networking table lists "Cluster ingress" and identifies its role asrouting inbound pod traffic. It further clarifies that this function is delivered by athird-party ingress controller, and explicitly names Contouras an example ("you can use any third-party ingress controller, such as Contour"). That mapping is exactly what option Adescribes: Contour is deployed to provideingresscapabilities so that inbound requests from outside the cluster can be routed to Kubernetes services and pods according to ingress rules. In other words, Contour is not a storage component (that would align to CSI/CNS/pvCSI), not a node lifecycle manager (that is handled by VKS/Cluster API/VM Service), and not an infrastructure health monitoring tool (that would be metrics/observability tooling). VCF 9.0 positions Contour specifically within

theingresspart of the networking feature set, makingAthe correct answer.

QUESTION 3
An administrator is deploying vSphere Kubernetes Service (VKS) on a VMware Cloud Foundation workload domain to support a new internal AI and data analytics platform. The environment must host both virtual machine (VM) applications and containerized workloads while maintaining a unified networking and security model through NSX. The design documentation outlines the requirements for the Supervisor infrastructure components.
What three components form the foundation of a VMware vSphere Kubernetes Service (VKS) Supervisor deployment? (Choose three.)

A. Cluster API
B. NSX Manager virtual machine
C. Supervisor control plane virtual machine
D. vCenter Virtual Distributed Switch
E. Virtual Machine Service
F. NSX Advanced Load Balancer Controller virtual machine

Answer: BCF
Explanation:
VCF 9.0 describes Supervisor networking with NSX as a model where NSX provides network connectivity to Supervisor control plane VMs, services, and workloads, and where the Supervisor can use either the NSX Load Balancer or the Avi Load Balancer. In the NSX + Avi design, the documentation identifies the Avi Load Balancer Controlleras a core infrastructure element: the Controller "interacts with vCenter to automate the load balancing for the VKS clusters," and is responsible for provisioning and coordinating Service Engines and exposing operational interfaces. Also, the Supervisor itself is anchored by the Supervisor control plane virtual machines. VCF 9.0 explains you deploy the Supervisor with one or three control plane VMs, and in a three-zone Supervisor the control plane VMs are distributed across zones for high availability. Finally, because the requirement explicitly calls for a unified networking/security model through NSX, the NSX Manager virtual machineis foundational to the NSX-based Supervisor design, as shown in the documented architecture and component descriptions for NSX-backed Supervisor deployments.

QUESTION 4
An administrator is upgrading to VKS 3.4 and encounters the following error during cluster creation using workload, yami:

```
Error from server (Forbidden): error when creating "workload.yaml": admission webhook
"capi.mutating.tanzukubernetescluster.run.tanzu.vmware.com" denied the request: Cluster and variable validation failed:
[spec.topology.variables[defaultStorageClass]: Invalid value: "\"vks-storage-policy\"": variable is not defined,
spec.topology.variables[nodePoolVolumes]: Invalid value: "[{\"capacity\":
{\"storage\":\"100Gi\"},\"mountPath\":\"/var/lib/containerd\",\"name\":\"containerd\",\"storageClass\":\"vks-storage-policy\"},
{\"capacity\":{\"storage\":\"100Gi\"},\"mountPath\":\"/var/lib/kubelet\",\"name\":\"kubelet\",\"storageClass\":\"vks-storage-
policy\"}]": variable is not defined, spec.topology.variables[trust]: Invalid value: "{\"additionalTrustedCAs\":
[{\"name\":\"additional-ca-1\"}]}": variable is not defined]
```

How should the administrator resolve this issue to successfully complete the upgrade?

A. Verify workload cluster versions to ensure compatibility
B. Remove the deprecated variables and apply the new workload, yaml.
C. Rename the vSphere storage policy and apply the new workload.yami.
D. Restart the Kubernetes services and restart the upgrade

Answer: B
Explanation:

The error shows an admission webhook denial where variable validation failed and multiple entries under spec. topology.variables[...] are reported as"variable is not defined". That message indicates the manifest is supplying variables that arenot part of the current Cluster API / topology schema enforced by the Supervisor during cluster creation. In VKS, cluster provisioning is declarative: you invoke the VKS API withkubectl + a YAML file, and "after the cluster is created, you update the YAML to update the cluster." When the API /schema changes between releases, older manifests can contain fields/variables that are no longer recognized, and the admission webhook blocks them to prevent creating an invalid cluster spec. This aligns with VMware's broader direction that the olderTanzuKubernetesCluster (TKC) API was deprecated and customers are encouraged to useCluster APIfor bootstrap/config/lifecycle management. In practice, to complete the upgrade/creation successfully, you must update the cluster manifest to match the supported schema: remove the deprecated/unknown topology variablesshown in the error (for example, the undefined storage-policy and trust variables) and re-apply the correctedworkload.yaml.

QUESTION 5
A cloud operations team is managing multiple Supervisor Clusters across two regions. Each region hosts its own vSphere Kubernetes clusters, integrated through a federated service mesh to enable consistent service connectivity and policy enforcement across environments. The application team wants to expose a multi-tier microservice named "GovApp", which includes front- end, API, and database services distributed between the two regions. Uniform traffic routing, identity, and security policies are also needed for these workloads regardless of the cluster or region in which they are deployed.
To meet these requirements, the architects decide to create a Global Namespace that spans both Supervisor Clusters.
Which two statements describe the requirements for a Global Namespace in a vSphere Kubernetes Service Mesh deployment? (Choose two.)

A. Applies a single policy to multiple namespaces across multiple clusters.
B. Defines an application boundary across multiple clusters.
C. Provides distributed Ingress and egress services to support multiple namespaces across multiple clusters.
D. Enables automatic placement of workloads to any global cluster based on traffic demand.
E. Provides centralized logging for Service Mesh traffic.

Answer: AB
Explanation:
A Global Namespace in a service-mesh-driven, multi-cluster design is fundamentally an organizational and policy boundary that lets platform teams treat multiple Kubernetes namespaces--across multiple clusters /regions--as one logical application domain. That is whyBis correct: it defines an application boundary spanning clusters, which is exactly what "GovApp" needs when its tiers are distributed across regions but must still behave as one application. It also explains why Ais correct: the value of a Global Namespace is that you can apply consistent identity, security, and traffic policies across all member namespaces and clusters, rather than configuring them separately per cluster. This supports uniform mTLS/authorization posture, consistent service-to-service access rules, and standardized routing behaviors regardless of where a service instance runs.
The other options describe outcomes that may be enabled by additional service-mesh features, but they are not the core requirements that define a Global Namespace: distributed ingress/egress is not guaranteed simply by the namespace construct, automatic workload placement is typically handled by schedulers/placement engines rather than the namespace boundary, and centralized logging is an observability capability outside the namespace requirement itself.

QUESTION 6
A cloud platform team is deploying multiple Supervisors in a VMware Cloud Foundation (VCF) environment to support a multi-tenant VKS architecture. Each Supervisor will host a distinct set of namespaces mapped to different organizational tenants such as Finance, Public Services, and R&D, where each tenant consumes resources from dedicated workload domains.
What is the purpose of the vcf context create command?

A. Provides authentication to the Supervisor cluster.
B. Creates a connection to VCF Operations for the current session.
C. Creates a named set of access parameters for a VKS cluster.
D. Creates a new storage profile for a cluster.

Answer: C
Explanation:
VCF 9.0 defines aVCF CLI contextas aconfiguration setfor a specific endpoint (for example, a Kubernetes cluster, vSphere Supervisor, or VCF Automation) and explains that a context stores the key connection parameters such asendpoint,context type,path to kubeconfig, andcredentials. This aligns directly with optionC, because vcf context create is used to create anamedcontext that encapsulates the access parameters needed to work with a specific Supervisor or VKS cluster and to switch between multiple environments without rewriting configuration each time. The documentation further shows vcf context create being used to connect either to a Supervisor endpointor to aworkload (VKS) cluster (by providing workload cluster name/namespace or by using a generated kubeconfig), which is exactly the multi-tenant operational pattern in the question. While authentication is part of what happens during context creation (for example, creating a configuration that includes a JWT for Kubernetes API access), the command's purpose is broader: it creates the reusablecontextobject that defineswhereandhowthe CLI connects.

QUESTION 7
After upgrading the vSphere Supervisor, an administrator notices that the vSphere Kubernetes Service, configured as a Core Supervisor Service, is stuck in a "Configuring" state.
What should the administrator do to finish the upgrade?

A. Restart the supervisor service.
B. Ensure the vSphere Kubernetes Service is a supported version.
C. Verify the administrator account can authenticate to the Supervisor service.
D. Verify the AppPlatform operator user account can authenticate with vCenter.

Answer: B
Explanation:
A Supervisor upgrade impacts the lifecycle and compatibility of Supervisor Services (including Core Supervisor Services). VMware guidance emphasizes validating compatibility for the components that depend on the Supervisor and remediating any incompatibilities as part of the overall upgrade process. If a Core Supervisor Service remains stuck in "Configuring" after the Supervisor is upgraded, a common and expected cause is that the service version is not compatible with the upgraded Supervisor/vCenter software state. In those cases, the upgrade workflow expects you to identify and update incompatible Supervisor Services to versions that match the new supported software level. Ensuring the vSphere Kubernetes Service is at asupportedversion (for the upgraded Supervisor) aligns with the documented approach: run compatibility checks and then update the versions of incompatible Supervisor Services so they can complete reconciliation and reach a healthy state.

QUESTION 8

Which type of storage is used by VMware vSphere Kubernetes Service (VKS) pods to store non-persistent data?

A. Container image storage
B. Ephemeral storage
C. Object storage
D. vSphere local storage

Answer: B
Explanation:
In Kubernetes terms,non-persistentpod data (for example, transient logs and scratch space) is handled byephemeral storage, meaning the data exists only for the lifetime of the pod/workload and is not meant to survive beyond it. In the VCF Workload Management documentation, this concept is described directly: a pod requiresephemeral storageto store transient Kubernetes objects such as "logs" and "emptyDir volumes," and this ephemeral (transient) storage "lasts as long as the pod continues to exist," disappearing when the pod reaches end of life.
While VKS clusters can also consumepersistent storagethrough storage classes and CSI integration for stateful needs, that is specifically for data that must be retained (persistent volumes/claims). The question asks specifically aboutnon-persistentpod data, which aligns with the documented ephemeral/transient storage behavior for pod runtime needs. Therefore, the correct choice isEphemeral storage.

QUESTION 9
A VMware vSphere Kubernetes Service (VKS) cluster exposes three layers of controllers to manage its lifecycle.
Which set identifies these layers?

A. Virtual Machine Service, Cluster API, and Cloud Provider Plug-in.
B. Cluster API, Node Problem Detector, CNI Plug-in.
C. API Gateway, StatefulSet Controller, Ingress Controller.
D. Virtual Machine Service, vCenter Server Appliance, Supervisor Service.

Answer: A
Explanation:
VCF 9.0 explicitly states: "The VKS exposes three layers of controllers to manage the lifecycle of a VKS cluster," and then enumerates those layers. The first layer is the set of components that integrate the workload cluster with Supervisor-backed resources, including aCloud Provider Plug-inthat integrates with the Supervisor and enables infrastructure integrations such as persistent volume requests being passed to the Supervisor (which is integrated with Cloud Native Storage). The second layer isCluster API, described as providing "declarative, Kubernetes-style APIs for cluster creation, configuration, and management," driven by resources that represent the cluster, the VMs making up the cluster, and cluster add-ons. The third layer is theVirtual Machine Service, which provides a declarative API for managing VMs and associated vSphere resources and is used to manage the lifecycle of the control plane and worker node VMs hosting a VKS cluster.

Thank You for Trying Our Product

Lead2pass Certification Exam Features:

★ More than **99,900** Satisfied Customers Worldwide.

★ Average **99.9%** Success Rate.

★ **Free Update** to match latest and real exam scenarios.

★ **Instant Download** Access! No Setup required.

★ Questions & Answers are downloadable in **PDF** format and **VCE** test engine format.

★ Multi-Platform capabilities - **Windows, Laptop, Mac, Android, iPhone, iPod, iPad**.

★ **100%** Guaranteed Success or **100%** Money Back Guarantee.

★ **Fast**, helpful support **24x7**.

View list of all certification exams: http://www.lead2pass.com/all-products.html

10% Discount Coupon Code: **ASTR14**