Nutanix.NCP-CN.v2026-03-26.q51

Exam Code:	NCP-CN
Exam Name:	Nutanix Certified Professional - Cloud Native v6.10
Certification Provider:	Nutanix
Free Question Number:	51
Version:	v2026-03-26
# of views:	154
# of Questions views:	510
https://www.freepdfdumps.com/Nutanix.NCP-CN.v2026-03-26.q51.html	

NEW QUESTION: 1
A Platform Engineer is deploying a new NKP cluster that has internet connectivity. Now, a Cloud Administrator and Security Administrator are discussing the security of communications between the NKP Kubernetes cluster and the container registry. The engineer proposes to have an on-prem private registry.
What is the most significant reason that the engineer should create a private registry instead of configuring a secure connection between the NKP cluster and Github (SaaS)?
A. Private registry license is included with NKP.
B. NKP requires specific registry versions.
C. NKP cannot connect to public clouds.
D. Private registry provides security and privacy.
Answer: D (LEAVE A REPLY)
The primary benefit of a private registry is to ensure security and privacy for container images, especially when dealing with sensitive data and compliance requirements (such as financial or government use cases).
While secure connections to public registries like DockerHub or GitHub container registries are possible, using a private registry ensures full control over image access and auditing.
Key reference:
"Private registries ensure images remain within the security and compliance boundaries of the enterprise, avoiding potential risks associated with public SaaS registries." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Private Registry Use Cases and Benefits" NCP-CN 6.10 Study Guide - "Security and Compliance Considerations"

NEW QUESTION: 2
When deploying NKP using the Nutanix provisioning method (CAPX), what are the supported OS platforms?
A. CentOS and Rocky Linux

B. Rocky Linux and Ubuntu
C. Flatcar, Rocky Linux, and Ubuntu
D. CentOS and Ubuntu
Answer: B (LEAVE A REPLY)
The NKPA course specifies the supported operating systems for NKP clusters deployed using the Nutanix provisioning method (CAPX), which leverages Cluster API for Nutanix (CAPX) to provision clusters on Nutanix AHV. The supported OS platforms for CAPX are Rocky Linux and Ubuntu, as these distributions are tested and optimized for Nutanix infrastructure and Kubernetes requirements.
Rocky Linux is a CentOS replacement adopted by Nutanix after CentOS 8's end-of-life in 2021, providing a stable, enterprise-grade OS. Ubuntu, particularly LTS versions like 20.04 or 22.04, is widely supported due to its compatibility with Kubernetes and Nutanix AHV. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "When deploying NKP with the Nutanix provisioning method (CAPX), the supported OS platforms are Rocky Linux and Ubuntu, ensuring compatibility with Nutanix AHV and Kubernetes." These OS images are typically prepared using NKP Image Builder (NIB) to include necessary components like kubeadm and containerd.
Incorrect Options:
* A. CentOS and Rocky Linux: CentOS 8 is no longer supported post-2021, and Nutanix has shifted to Rocky Linux.
* C. Flatcar, Rocky Linux, and Ubuntu: Flatcar Container Linux is not a supported OS for CAPX in NKP deployments.
* D. CentOS and Ubuntu: CentOS is not supported, as noted above.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Nutanix Provisioning with CAPX.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on NKP Deployment Prerequisites.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 3
A Platform Engineer is getting started with NKP and has created a bastion host with all needed prerequisites.
How should the engineer install Kommander?
A. AWS CLI
B. Terraform
C. Ansible
D. NKP CLI
Answer: D (LEAVE A REPLY)
As per the NKPA 6.10 documentation, the standard and supported approach for installing Kommander on an NKP cluster is using the NKP CLI. The CLI provides commands for deploying Kommander and associated platform components as part of the cluster lifecycle management workflow.

Key Reference:
"The recommended approach to deploy Kommander is by using the nkp CLI, which ensures compatibility and streamlined installation with the rest of the Nutanix Kubernetes Platform stack."
Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Installing Kommander with NKP CLI"
NCP-CN 6.10 Study Guide - "Kommander Installation and Management"

NEW QUESTION: 4
Prior to implementing NKP, a company had created a number of Kubernetes (K8s) clusters using kubeadm.
While they are deploying new managed clusters via NKP, the company does not wish to migrate workloads from these pre-existing native K8s clusters over to new NKP clusters just yet.
What are the requirements to have these clusters attached to their NKP management cluster?
A. The version of the K8s clusters must be within N - 1 of the Kubernetes version of the NKP management cluster.
B. The NKP management cluster must be able to reach the services and api-server of the target cluster.
C. The version of the K8s clusters must match the Kubernetes version of the NKP management cluster.
D. An NKP management cluster admin account must be established on the K8s clusters.
Answer: B (LEAVE A REPLY)
As per the NKPA 6.10 documentation, the primary requirement for attaching external (self-managed) Kubernetes clusters to NKP is network connectivity. Specifically, the NKP management cluster must be able to communicate with the Kubernetes API server and relevant services of the target cluster. This allows NKP to collect metrics, perform health checks, and manage the attached cluster through Kommander and associated tools.
Exact extract:
"For attaching existing Kubernetes clusters, ensure the NKP management cluster can reach the Kubernetes API server of the target cluster and that the kubeconfig used has sufficient permissions." There is no requirement that the Kubernetes versions be exactly matched or within N-1, nor that an NKP admin account be directly established on the target clusters; the connectivity and valid kubeconfig file are the essential requirements.
Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Attaching External Clusters" NCP-CN 6.10 Study Guide - "External Cluster Integration"

NEW QUESTION: 5
A company was using a test application called temp-shop developed in the temp-ecommerce NKP Starter cluster. Now, the cluster has just been taking up valuable resources that could be used for other projects, so the development team has decided to remove it.
Before proceeding, they verified that they had the cluster configuration file temp-ecommerce.conf.

What command should the development team execute to delete the cluster with its nodes and application?

A. nkp delete cluster --all

B. nkp delete cluster --application-name=temp-shop --self-managed --kubeconfig=temp-ecommerce.conf

C. nkp delete cluster --cluster-name=temp-ecommerce --self-managed --kubeconfig=temp-ecommerce.conf

D. nkp delete cluster --cluster-name=temp-shop --self-managed --kubeconfig=temp-shop.conf

Answer: C (LEAVE A REPLY)

As per the NKPA 6.10 documentation, the correct approach for deleting an entire NKP-managed cluster (including nodes and deployed applications) is to use the nkp delete cluster command with the cluster name and --kubeconfig parameter. The --self-managed flag ensures the deletion of resources provisioned by the cluster.

Exact extract:

"Use nkp delete cluster --cluster-name=<name> --self-managed --kubeconfig=<config> to delete a cluster and its associated resources." Reference:

Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Cluster Deletion Workflow" NCP-CN 6.10 Study Guide - "Deleting a Managed Cluster"

=======

NEW QUESTION: 6

A company has been modernizing on cloud-native platforms for the past few years and has been running some small consumer support utilities on their production NKP cluster. After a thorough testing and QA cycle with simulated workloads on a development cluster, the company is ready to bring their online retail application into the fold. While they have sufficient system resources to scale the NKP cluster properly from a performance standpoint, they also want to ensure they properly scale their monitoring stack's resource settings to retain a sufficient amount of data to see how overall system resource utilization trends for the NKP cluster over several months' time with the added workloads. Which NKP Platform Application component should the company be most concerned with adjusting, and how should their Platform Engineer adjust it?

A. Adjust the number of replicas for the Fluent Bit deployment, as well as increase the amount of storage available for use by the NKP cluster.

B. Adjust the number of replicas for the Prometheus deployment, as well as increase the amount of storage available for use by the NKP cluster.

C. Adjust the resource settings for Fluent Bit by increasing its container resource limits and memory settings, as well as its storage.

D. Adjust the resource settings for Prometheus by increasing its container resource limits and memory settings, as well as its storage.

Answer: D (LEAVE A REPLY)

The NKPA course explains that NKP's monitoring stack includes Prometheus for metrics collection and storage, and Fluent Bit for log collection and forwarding. To retain system resource

utilization data over several months, the company must focus on Prometheus, as it is responsible for storing time-series metrics data, such as CPU, memory, and network utilization, which are critical for long-term trend analysis.
To handle the increased workload from the online retail application, the Platform Engineer should adjust Prometheus by increasing its container resource limits and memory settings to ensure it can process and store more metrics, and increasing its storage to retain data for several months.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "For long-term metrics retention in NKP, scale the Prometheus deployment by increasing its resource limits (CPU and memory) and expanding its persistent storage to accommodate larger time-series data." This involves editing the Prometheus deployment's resource settings (e.
g., via kubectl edit deployment) and updating the PersistentVolumeClaim (PVC) to allocate more storage.
Incorrect Options:
* A. Adjust the number of replicas for Fluent Bit: Fluent Bit handles logs, not metrics. Increasing replicas does not address long-term metrics storage.
* B. Adjust the number of replicas for Prometheus: Increasing replicas improves availability but does not directly address storage or resource needs for metrics retention.
* C. Adjust the resource settings for Fluent Bit: Fluent Bit is for log collection, not metrics storage, and is not relevant for this use case.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Monitoring and Platform Applications.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com
Prometheus Documentation: https://prometheus.io

NEW QUESTION: 7

A Platform Engineer sees no insight in all workspaces, and it is a critical feature to control all the alerts on all company's Kubernetes clusters. What does the engineer need to do to begin generating NKP Insights?
A. Acquire the NKP Insights Add-on license.
B. Install nkp-insights in every Kubernetes cluster with kubectl apply -f nkp-insights-1.2.2 --kubeconfig=<cluster>.conf
C. Create a persistent volume claim and assign it to nkp-insights; this application requires volumes to save logs and data. Install nkp-insights with nkp create appdeployment nkp-insights --app nkp-insights-1.2.2

--workspace kommander-workspace

Answer: (SHOW ANSWER)

The exhibit shows an NKP UI message stating "No insights identified," indicating that NKP Insights, the platform's predictive analytics and observability feature, is not active. The NKPA course explains that NKP Insights is a platform application that provides anomaly detection and alerting for Kubernetes clusters but requires deployment and configuration to generate insights. The correct approach (Option C) involves two steps:

* Create a persistent volume claim (PVC) and assign it to nkp-insights: NKP Insights requires persistent storage to save logs, metrics, and historical data for analysis. The course specifies that a PVC must be created to provide this storage, ensuring the application can retain data for generating insights.
* Install nkp-insights with nkp create appdeployment nkp-insights --app nkp-insights-1.2.2 --workspace kommander-workspace: This command deploys the NKP Insights application across all clusters in the specified workspace (kommander-workspace). The --app flag specifies the application version, and the deployment ensures that insights are generated for all clusters in the workspace.

The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To enable NKP Insights, create a PersistentVolumeClaim for storage and deploy the application using nkp create appdeployment nkp-insights -- app <version> --workspace <workspace-name> to begin generating insights across clusters." This aligns with the need to control alerts across all company clusters, as NKP Insights provides centralized observability.

Incorrect Options:

* A. Acquire the NKP Insights Add-on license: NKP Insights is a platform application included in higher-tier licenses (e.g., NKP Ultimate), not a separate add-on. The course does not indicate a separate license requirement.
* B. Install nkp-insights in every cluster with kubectl apply: This method is manual and inefficient. NKP's appdeployment command automates deployment across all clusters in a workspace.

:

Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Observability and Insights.

Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.

Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 8

Refer the exhibit.

After selecting the Production workspace and selecting View Details for the cluster prod-01, a Platform Engineer wanted to enable the NKP Insights application. This application is under the Observability category, but this category doesn't appear in the list.

Which action should the engineer take to be able to deploy the NKP Insights application in the Kubernetes cluster for this workspace?

A. Select Clusters in the left menu, select Applications, and select Enable in the NKP Insights three-dot menu.

B. Select Applications in the left menu, press the three-dot menu in the NKP Insights application option, and select Enable.

C. Select Insights in the left menu and select the Enable button.

D. Select Clusters in the left menu, select View Details for the cluster prod-01, then in the Application Dashboard, select Enable in the NKP Insights three-dot menu.

Answer: D (LEAVE A REPLY)

As per the NKPA 6.10 documentation under "Day 2 Operations: Managing Applications", the recommended procedure to enable an application (like NKP Insights) involves accessing the cluster's Application Dashboard. The NKP Insights application does not appear in the general category list if the cluster does not have the correct context or if the application category is not globally enabled.

The specific procedure to enable NKP Insights includes:

* Navigate to the Clusters section from the left-side menu.
* Select View Details for the target cluster (prod-01 in this case).
* In the Application Dashboard of that specific cluster, locate the NKP Insights application.
* Click the three-dot menu (ellipsis) for NKP Insights and select Enable.

Reference:

Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Enabling Applications in a Specific Cluster" NCP-CN 6.10 Study Guide - "Application Deployment in Workspaces" This exact approach ensures that NKP Insights is deployed in the proper cluster-level context, circumventing the missing "Observability" category in the global Applications view.

=======

NEW QUESTION: 9
A company has a new DevOps team that needs to be provided cloud native computing resources. This team will need to have access to multiple NKP clusters for development, testing and validation of an in-house application. However, they also need to be restricted to a specific namespace and a consistent level of access across the clusters within this namespace so that they do not adversely impact the environment of other user groups or the clusters themselves. As a part of this consideration, the new team also needs to be limited to the amount of storage, CPU and memory they can consume on the clusters. A Platform Engineer has been tasked with providing the appropriate level of access to the team on these multiple NKP clusters.
How should the engineer best accomplish this task?
A. 1.Create an NKP project and assign the NKP clusters to it.
2.Create quotas for the NKP project.
3.Create an NKP group for the DevOps team.
4.Assign that NKP group the proper RBAC roles within that NKP project.
B. 1.Create an NKP workspace and assign the NKP clusters to it.
2.Create quotas for the NKP workspace.
3.Create an NKP group for the DevOps team.
4.Assign that NKP group the proper RBAC roles within that NKP workspace.
C. 1.Create an NKP group for the DevOps team.
2.Assign that NKP group the proper RBAC roles in the NKP UI.
3.Select the export to yaml option once the RBAC role assignment to that NKP group is complete.
4.Apply the exported manifest to each of the NKP clusters using kubectl.
D. 1.Enable Gatekeeper on the NKP clusters.
2.Create an NKP group for the DevOps team.
3.Assign Gatekeeper quota and authorization policies to that NKP group.
Answer: C (LEAVE A REPLY)
The NKP documentation recommends using a combination of RBAC assignments and exported manifests to manage access across multiple clusters. By exporting the role assignments to YAML, the engineer can consistently apply these settings across the different environments, ensuring the new team has the necessary resources and limits. This approach is especially useful for environments with multiple clusters and standardized configurations.
References: Nutanix Kubernetes Platform Administration Guide - RBAC Management Across Clusters

NEW QUESTION: 10
Some time ago, an EKS cluster was attached to be managed with NKP (Fleet Management). Now, a Platform Engineer has been asked to disconnect the EKS cluster from NKP for licensing reasons. After disconnecting the cluster, the developers realized that application changes are still being reflected in the EKS cluster, despite the fact that the EKS cluster was successfully detached from NKP. How should the engineer resolve this issue?

A. Forcefully detach EKS cluster: nkp detach cluster -c detached-cluster-name --force
B. Detached cluster must also be deleted from NKP: nkp delete cluster -c detached-cluster-name
C. Developers must have some bad configuration in the deployment config files. Ask for revision or call AWS technical support.
D. Detached cluster's Flux installation must be manually disconnected from the management Git repository: kubectl -n kommmander-flux patch gitrepo management -p '{"spec":{"suspend":true}}' -- type merge
Answer: D (LEAVE A REPLY)
When an Amazon EKS cluster is attached to NKP for fleet management, NKP uses GitOps principles, leveraging Flux (a GitOps operator) to synchronize application deployments and configurations from a management Git repository to the attached cluster. The NKPA course explains that detaching a cluster from NKP removes it from the NKP management plane, but the Flux installation on the cluster may continue to reconcile with the Git repository, causing application changes to persist post-detachment.
To resolve this, the engineer must manually disconnect the Flux installation by suspending the Git repository reconciliation. The correct command, as per the NKPA course, is: kubectl -n kommmander-flux patch gitrepo management -p '{"spec":{"suspend":true}}' --type merge. This command suspends Flux's synchronization with the management Git repository, stopping further application updates. The Nutanix Cloud Native (NCP- CN) 6.10 Study Guide states: "After detaching an external cluster from NKP, the Flux GitOps operator may continue to apply changes unless its GitRepo resource is suspended using kubectl patch in the kommmander- flux namespace." Incorrect Options:
* A. Forcefully detach EKS cluster: The --force flag is not a standard option for the nkp detach cluster command, and forceful detachment does not address the Flux reconciliation issue.
* B. Detached cluster must also be deleted from NKP: The cluster is already detached, and deletion is not necessary to stop GitOps updates. The issue lies with Flux, not NKP's state.
* C. Developers must have some bad configuration: The issue is not with developer configurations but with Flux's ongoing synchronization, as explained in the NKPA course.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Fleet Management and GitOps.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Detaching Clusters.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com Flux Documentation: https://fluxcd.io

NEW QUESTION: 11
In an effort to control cloud cost consumption, auto-scale is configured to meet demands as needed.
What is the behavior for when nodes are scaled down?
A. Node is changed to a status of Hibernate.

B. Node is CAPI deleted from its infrastructure provider, effectively removing it from its hypervisor.
C. Node is changed to a status of Power-Off for stand-by.
D. Node is paused in Kubernetes and the infrastructure continues to consume the resources at the current level.
Answer: B (LEAVE A REPLY)
As per the NKPA 6.10 documentation and cluster autoscaler behavior, when nodes are scaled down in NKP (or any CAPI-managed environment), the node is deleted from the infrastructure provider (vSphere, AWS, Nutanix, etc.). This effectively removes it from both the cluster and the underlying hypervisor or cloud provider, thus freeing up resources and reducing costs.
Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Cluster Autoscaler Node Deletion Behavior" NCP-CN 6.10 Study Guide - "Autoscaler Impact on Infrastructure Resources"

NEW QUESTION: 12
Which CAPI provisioning method requires creating an inventory file of the servers to become NKP nodes?
A. AWS (CAPA)
B. Nutanix (CAPX)
C. Pre-provisioned (CAPPP)
D. vSphere (CAPV)
Answer: C (LEAVE A REPLY)
According to the NKPA 6.10 documentation, the Pre-provisioned infrastructure provider (CAPPP) requires the user to provide an inventory file of the servers that will become the cluster nodes. This inventory file contains connection information (IP addresses, credentials, etc.) for each pre-provisioned node, enabling the Cluster API to connect and configure these servers directly during cluster creation.
Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "CAPPP (Pre-Provisioned) Overview" NCP-CN 6.10 Study Guide - "Inventory Files for Pre-Provisioned Infrastructure"
=======

NEW QUESTION: 13
A Platform Engineer will be deploying an NKP cluster in a dark site with no Internet access. The Cloud Administrator has provided a Linux VM for this purpose, so the engineer needs to prepare this VM to be used as a bastion host. Which two actions should the engineer take to complete this task? (Choose two.)
A. Install LDAP Server.
B. Get or create SSH Keys.
C. Install Docker.
D. Enable NTP Service.

Answer: B,D (LEAVE A REPLY)
A bastion host in a dark site environment serves as a secure entry point for managing the NKP deployment, providing access to the cluster infrastructure without direct Internet connectivity. The NKPA course outlines the prerequisites for preparing a Linux VM as a bastion host, focusing on secure access and time synchronization, which are critical for air-gapped Kubernetes deployments.
* Get or create SSH Keys (Option B):The bastion host requires SSH keys to enable secure, passwordless access to the NKP cluster nodes and other infrastructure components (e.g., Nutanix AHV hosts). The NKPA course specifies that SSH keys must be generated or obtained and configured on the bastion host to facilitate secure communication during deployment and management tasks. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "For a bastion host in an NKP dark site deployment, ensure SSH keys are created or obtained to enable secure access to cluster nodes and infrastructure." The engineer can generate SSH keys using ssh-keygen and distribute the public key to the target systems.
* Enable NTP Service (Option D):Time synchronization is essential in Kubernetes clusters to ensure consistent logging, certificate management, and scheduling. In a dark site with no Internet access, the bastion host must be configured to synchronize time with an internal NTP (Network Time Protocol) server or act as an NTP server itself. The NKPA course emphasizes enabling the NTP service on the bastion host to maintain accurate time across the air-gapped environment. The NCP-CN 6.10 Study Guide notes: "Enable the NTP service on the bastion host to ensure time synchronization in a dark site NKP deployment, as Kubernetes requires accurate time for proper operation." The engineer can enable NTP using commands like systemctl enable ntpd and configure it to use an internal time source.
Incorrect Options:
* A. Install LDAP Server: LDAP is used for centralized authentication, but it is not a requirement for a bastion host in an NKP dark site deployment. The course focuses on SSH access instead.
* C. Install Docker: While Docker is needed on Kubernetes nodes for container runtimes, the bastion host's role is to provide secure access and management, not to run containers.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Preparing for Dark Site Deployments.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on NKP Deployment Prerequisites.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 14
A company is required by NIST to follow FIPS guidelines for compliance.
What is the first step for enabling FIPS in NKP?
A. Run the command export FIPS_ENABLED=true
B. Run the command nkp cluster create <provisioner> <options> --fips
C. Follow the OS vendor's instructions to ensure that the OS or OS images are prepared for operating in FIPS mode.

D. Click Enable in the NKP Kommander Web UI, Global Workspace -> Settings -> FIPS menu.
Answer: C (LEAVE A REPLY)
According to the NKPA 6.10 documentation under the "Preparing for FIPS Compliance" section, the first step for enabling FIPS in an NKP environment is to ensure that the underlying operating system (OS) or OS images are correctly configured for FIPS mode. The Nutanix Kubernetes Platform can only leverage FIPS- compliant cryptographic modules if the underlying OS is already configured for FIPS operation.
Specifically, the documentation states:
"Before enabling FIPS in NKP, ensure that the OS or OS images are prepared and configured to operate in FIPS mode, following the vendor's guidelines. Once the OS is in FIPS mode, the NKP cluster components can be provisioned with FIPS support." This foundational step is critical to ensure cryptographic consistency and compliance throughout the NKP stack.
Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "FIPS Compliance" NCP-CN 6.10 Study Guide - "Preparing OS Images for FIPS"
=======

NEW QUESTION: 15
A Platform Engineer has deployed NKP and wants to utilize its OOB data storage feature.
What should the engineer enable to support backups within the NKP environment?
A. MinIO
B. Rook Ceph
C. Volumes iSCSI
D. Objects S3
Answer: (SHOW ANSWER)
According to the NKPA 6.10 documentation, Rook Ceph is the recommended out-of-the-box storage solution integrated with NKP for persistent data storage, backups, and object storage within the Kubernetes environment.
Key Reference:
"Rook Ceph is integrated with NKP for providing persistent storage and backup support within the cluster." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Storage Architecture in NKP" NCP-CN 6.10 Study Guide - "Rook Ceph for OOB Storage"

NEW QUESTION: 16
A technology company has decided to migrate its infrastructure to NKP to improve the scalability and management of its applications. After a successful initial implementation, the operations team faces a new challenge of validating the HelmReleases to ensure that all applications are running correctly and avoid problems in production. Which command should the company execute to know the right status of their HelmReleases?
A. kubectl get namespaces

B. kubectl get helmreleases -n ${PROJECT_NAMESPACE}
C. kubectl edit helmreleases -n ${PROJECT_NAMESPACE}
D. kubectl apply -f fluent-bit-overrides.yaml
Answer: B (LEAVE A REPLY)
NKP uses Helm and Flux for application deployment, where applications are managed as HelmReleases, a custom resource that defines Helm chart deployments. The NKPA course explains that to validate the status of HelmReleases, administrators can use the kubectl command to query these resources. The correct command is kubectl get helmreleases -n ${PROJECT_NAMESPACE}, which lists all HelmReleases in the specified project namespace, showing their status (e.g., Ready, Failed) and other details like chart version and reconciliation status.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To check the status of HelmReleases in NKP, use kubectl get helmreleases -n <namespace> to view the current state of application deployments managed by Flux." This command helps the operations team verify that applications are running correctly and identify any issues in production.
Incorrect Options:
* A. kubectl get namespaces: This lists namespaces, not HelmReleases or their status.
* C. kubectl edit helmreleases -n ${PROJECT_NAMESPACE}: This edits HelmRelease resources, not displays their status.
* D. kubectl apply -f fluent-bit-overrides.yaml: This applies a configuration for Fluent Bit, unrelated to HelmRelease status.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Application Management.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com Flux Documentation: https://fluxcd.io/docs/components/helm

NEW QUESTION: 17
A company recently deployed NKP. A Platform Engineer was asked to attach the existing Amazon EKS. A workspace and project were created accordingly, and resource requirements were met. What does the engineer need to do first to prepare the EKS clusters?
A. Configure a ConfigMap according to EKS configuration.

B. Create a service account with cluster-admin permissions.
C. Configure HAProxy to get connected to EKS clusters.
D. Deploy cert-manager in the EKS clusters.
Answer: (SHOW ANSWER)
Attaching an existing Amazon EKS cluster to NKP for fleet management involves integrating the cluster into NKP's management plane, which requires specific preparatory steps. The NKPA course outlines that the first step is to create a service account with cluster-admin permissions in the EKS cluster. This service account is used by NKP to authenticate and manage the cluster, enabling operations like monitoring, scaling, and application deployment.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide explains: "To attach an external Kubernetes cluster, such as Amazon EKS, to NKP, a service account with cluster-admin role bindings must be created to allow NKP to interact with the cluster's API server." The service account is configured with a token that NKP uses to authenticate requests. The NKPA course provides detailed steps, including creating the service account, assigning the cluster-admin ClusterRole, and generating a token for NKP integration. This step is critical to ensure NKP has the necessary permissions to manage the EKS cluster.
Incorrect Options:
* A. Configure a ConfigMap according to EKS configuration: While ConfigMaps may be used for specific configurations, they are not the first step for attaching an EKS cluster. The NKPA course prioritizes service account creation.
* C. Configure HAProxy to get connected to EKS clusters: HAProxy is a load balancer, not required for attaching EKS clusters to NKP. EKS uses AWS-native load balancers, and NKP connects via the Kubernetes API.
* D. Deploy cert-manager in the EKS clusters: Cert-manager is used for certificate management, not a prerequisite for attaching EKS clusters. The NKPA course does not list it as a required step.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Fleet Management.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Attaching External Clusters.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com Amazon EKS Documentation: https://docs.aws.amazon.com/eks

NEW QUESTION: 18
A Platform Engineer is attaching existing Kubernetes clusters to NKP, but a particular Kubernetes Amazon EKS cluster is getting errors with application deployments. These errors are related to persistent volumes.
What could be the issue, and what can the engineer do?
A. The storage appliance is having issues. The storage engineer should be contacted to take a look.
B. There is no compatible storage to be attached to the EKS cluster. Ask for compatible storage.
C. There is no default StorageClass. Storage classes should be reviewed, and only one should be set as default.

D. There could be a misconfiguration in the ConfigMap. It should be adjusted to NKP requirements.

Answer: C (LEAVE A REPLY)

When attaching an Amazon EKS cluster to NKP for fleet management, persistent volume (PV) errors during application deployments often indicate issues with storage configuration. The NKPA course identifies a common cause: the absence of a default StorageClass. Kubernetes requires a default StorageClass to automatically provision PVs for PersistentVolumeClaims (PVCs) when none is specified. Without a default, applications fail to bind PVCs, resulting in deployment errors. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "For attached EKS clusters, ensure a default StorageClass is configured to support dynamic provisioning of persistent volumes. Review existing StorageClasses and set one as default using the storageclass.kubernetes.io/is-default-class annotation." The engineer should run kubectl get storageclass to list available StorageClasses, verify their configurations, and set one as default by patching it with kubectl patch storageclass <name> -p '{"metadata": {"annotations": {"storageclass.kubernetes.io/is-default-class":"true"}}}'. In EKS, the default StorageClass is typically backed by Amazon EBS (e.g., gp2 or gp3).

Incorrect Options:

* A. The storage appliance is having issues: This assumes a hardware issue, which is not indicated by PV errors. The NKPA course suggests checking Kubernetes configurations first.
* B. There is no compatible storage: EKS supports EBS and EFS, which are compatible with NKP. The issue is likely configuration, not compatibility.
* D. There could be a misconfiguration in the ConfigMap: ConfigMaps are not directly related to PV provisioning. The NKPA course points to StorageClass issues for PV errors.

:

Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Fleet Management.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Storage Configuration.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com Amazon EKS Storage Documentation: https://docs.aws.amazon.com/eks

NEW QUESTION: 19

A Kubernetes administrator needs to deploy a new Kubernetes cluster into a new workspace. This cluster requires a predictive analytics solution that detects current and future anomalies. Which option does the administrator need to deploy after the cluster is ready?

A. NKP Insights

B. NCM Intelligent Ops

C. Nutanix Pulse

D. NKP AI Navigator

Answer: A (LEAVE A REPLY)

The NKPA course highlights that NKP Insights is the platform's integrated solution for observability and analytics, providing predictive analytics to detect current and future anomalies in Kubernetes clusters. NKP Insights leverages machine learning to analyze metrics, logs, and

events, identifying performance issues and potential failures proactively. After deploying a new cluster, the administrator can enable NKP Insights as a platform application in the workspace to meet the predictive analytics requirement.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "NKP Insights provides predictive analytics and anomaly detection for Kubernetes clusters, enabling administrators to monitor and optimize cluster performance." This is deployed via the NKP UI or CLI by enabling the Insights application in the workspace.
Incorrect Options:
* B. NCM Intelligent Ops: Nutanix Cloud Manager (NCM) Intelligent Ops is for broader infrastructure management, not Kubernetes-specific analytics.
* C. Nutanix Pulse: Pulse is a telemetry service for Nutanix support, not for analytics.
* D. NKP AI Navigator: This is not a recognized NKP component in the NKPA documentation.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Observability and Insights.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 20
In a telecom company, two teams were working on the development of two different applications:
* ApplicationA
* ApplicationBApplicationA's development team was excited about the release of their new functionality. However, upon deploying their application, they noticed that performance was slow. After investigating, they discovered that the ApplicationB team was consuming the majority of the cluster's resources, affecting all other teams. How can this problem be mitigated?
A. Implementing Quotas and Limit Ranges
B. Setting up Network Policies
C. Configuring RBAC
D. Implementing Continuous Deployment (CD)
Answer: A (LEAVE A REPLY)
The NKPA course addresses resource contention in Kubernetes clusters, where one application (e.g., ApplicationB) consumes excessive resources, impacting others (e.g., ApplicationA). To mitigate this, the course recommends implementing Quotas and Limit Ranges.
* Resource Quotas restrict the total amount of CPU, memory, and other resources a namespace can consume, ensuring fair resource allocation across teams.
* Limit Ranges set minimum and maximum resource limits for individual pods and containers within a namespace, preventing any single application from monopolizing resources.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To prevent resource contention in NKP clusters, apply Resource Quotas to limit namespace resource usage and Limit Ranges to enforce pod and container resource boundaries." For example, the administrator can create a

ResourceQuota to cap ApplicationB's namespace resource usage and a LimitRange to restrict its pod resource requests/limits, ensuring ApplicationA has sufficient resources.
Incorrect Options:
* B. Setting up Network Policies: Network Policies control network traffic, not resource usage.
* C. Configuring RBAC: RBAC manages access permissions, not resource allocation.
* D. Implementing Continuous Deployment (CD): CD automates deployments but does not address resource contention.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Resource Management.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com
Kubernetes Documentation: https://kubernetes.io/docs/concepts/policy/resource-quotas

NEW QUESTION: 21
An infrastructure team has configured a Backup Storage Location on an existing AWS bucket and created a backup named testbackup. What command can the team use to view the backup?
A. kubectl get backupstoragelocations -n ${testbackup} -o yaml
B. velero backup describe aws-velero-testbackup
C. velero backup describe testbackup
D. kubectl get backupstoragelocations -n ${WORKSPACE_NAMESPACE} -o yaml
Answer: C (LEAVE A REPLY)
The Nutanix Kubernetes Platform (NKP) integrates Velero, an open-source tool, for backup and restore operations as part of its Day 2 operations. The NKPA course explains that after configuring a Backup Storage Location (e.g., an AWS S3 bucket) and creating a backup, administrators can view details of the backup using the Velero CLI. The correct command to view the details of a backup named testbackup is velero backup describe testbackup. This command provides a detailed description of the backup, including its status, resources included, and storage location.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To inspect a Velero backup in NKP, use the velero backup describe <backup-name> command to display detailed information about the backup, such as its creation time, expiration, and included resources." The backup name (testbackup) is specified as created by the team, and no prefix like aws-velero- is indicated in the question, making option C the correct choice.
Incorrect Options:
* A. kubectl get backupstoragelocations -n ${testbackup} -o yaml: This command retrieves Backup Storage Location objects, not backup details. The namespace ${testbackup} is also incorrect, as Velero resources are typically in a specific namespace (e.g., velero).
* B. velero backup describe aws-velero-testbackup: The backup name is testbackup, not aws-velero- testbackup. The NKPA course does not indicate any prefix for the backup name.
* D. kubectl get backupstoragelocations -n ${WORKSPACE_NAMESPACE} -o yaml: This retrieves Backup Storage Locations, not the backup itself, and does not provide backup details.

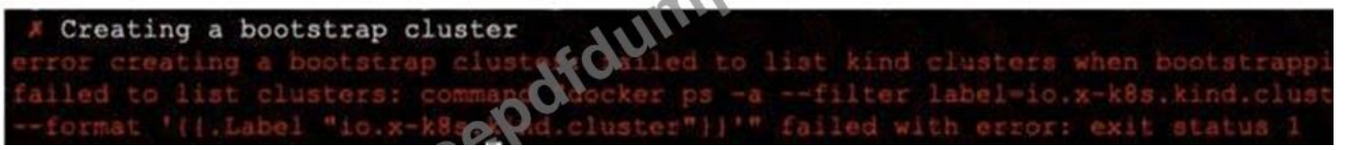
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Backup and Restore with Velero.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com Velero Documentation: https://velero.io

NEW QUESTION: 22
A Platform Engineer is deploying NKP into a highly secure vSphere environment. The NKP cluster will be air-gapped, and must also be FIPS compliant. The OS platform to be used for NKP cluster nodes is RHEL.
What must the engineer do to properly prep an OS image and have it be deployed as a FIPS compliant NKP cluster node?
A. Verify the OS itself has been placed in fips mode.
When performing the NKP image build operation, be sure to include the offline fips override.
When performing the NKP cluster deploy operation, be sure to include the FIPS version references for kubernetes and etcd.
B. Verify the OS itself has been placed in fips mode.
When performing the NKP image build operation, be sure to include the offline fips and fips overrides.
When performing the NKP cluster deploy operation, be sure to include the FIPS version references for kubernetes and etcd.
C. Verify the OS itself has been placed in fips mode.
When performing the NKP image build operation, be sure to include the offline fips and fips overrides.
When performing the NKP cluster deploy operation, be sure to include the FIPS version references for kubernetes, kubectl and etcd.
D. Verify the OS itself has been placed in fips mode.
When performing the NKP cluster deploy operation, be sure to include the FIPS version references for kubernetes, kubectl and etcd.
Answer: B (LEAVE A REPLY)
To create a FIPS-compliant NKP deployment in an air-gapped environment, the following must be done:
* Verify the OS (RHEL) itself is in FIPS mode.
* When building the image, include both the offline fips and fips overrides to ensure the image and components are built in compliance.
* When deploying the cluster, include FIPS-specific references for both Kubernetes and etcd components.
Exact extract:
"For FIPS deployments, ensure the OS is in FIPS mode, and include both offline fips and fips overrides during image creation to ensure compliant images and deployment." Reference:

Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Deploying NKP in FIPS Environments" NCP-CN 6.10 Study Guide - "FIPS Compliance Best Practices"
=======

NEW QUESTION: 23

Refer to the exhibit.



A Platform Engineer is trying to create a new NKP cluster and is getting the error shown in the exhibit.

What is the most likely cause of this error?

A. A docker compatible runtime is not running

B. Informatting in the Ansible playbook

C. Inpermissions to the NKP binary

D. An inHelm chart repo was referenced

Answer: A (LEAVE A REPLY)

The error message:

python

Copy

failed to list clusters: command "docker ps -a --filter label=io.x-k8s.kind.cluster --format '{{.Label "io.x-k8s.

kind.cluster"}}'" failed with error: exit status 1

indicates that the command failed while attempting to list docker containers with the given filter. The kind tool (used for creating a bootstrap cluster) depends on Docker as the container runtime.

Root cause:

If Docker is not running or a compatible container runtime is not available (such as containerd), the kind tool cannot list or interact with the required containers, resulting in this error.

Key Reference:

"When creating bootstrap clusters, the kind tool requires Docker to be installed and running. If Docker is not running, errors will be encountered listing or interacting with containers." Reference: Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Bootstrap Cluster Creation Requirements" NCP-CN 6.10 Study Guide - "Bootstrap Cluster Creation Dependencies"

NEW QUESTION: 24

A development team is working on a new application that requires access to certain cluster resources. The team needs to ensure that they have limited permissions to avoid unauthorized changes in other environments.

Among the tasks they will perform are the following:

* Deploy new versions of the application to their specific namespace.

* Scale deployments according to demand.
* View logs and metrics of their applications to monitor performance.When using the NKP GUI, what type of access should the team configure?

A. NKP Role

B. Cluster Role

C. Cluster Admin

D. Kommander Role

Answer: A (LEAVE A REPLY)

The NKPA course explains that NKP provides a role-based access control (RBAC) system to manage permissions within its platform, in addition to Kubernetes-native RBAC. For a development team needing limited permissions to perform specific tasks (deploying applications, scaling deployments, viewing logs and metrics) within a specific namespace, the appropriate access type in the NKP GUI is an NKP Role.

NKP Roles are predefined or custom roles within the NKP platform that map to Kubernetes RBAC permissions but are managed through the NKP UI for ease of use. They allow granular control over actions within a workspace or namespace, ensuring the team can perform their tasks (e.g., deploy, scale, get logs) without having access to other environments or cluster-wide resources. For example, an NKP Role like

"Developer" or a custom role can be configured to grant edit permissions in the team's namespace while restricting access elsewhere. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "In the NKP GUI, configure an NKP Role to grant limited permissions to a development team, allowing actions like deploying applications and viewing logs within their namespace while preventing unauthorized changes in other environments." Incorrect Options:

* B. Cluster Role: A Kubernetes Cluster Role grants permissions across all namespaces, which is too broad for the team's limited access requirement.
* C. Cluster Admin: This grants full administrative access to the entire cluster, far exceeding the team's needs and violating the principle of least privilege.
* D. Kommander Role: Kommander is a management component in NKP, but "Kommander Role" is not a specific access type in the NKP GUI for this purpose.

:

Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Access Control and RBAC.

Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.

Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 25

An administrator has been tasked with deploying NKP as the Kubernetes platform and needs to deploy their first cluster with the following requirements:

* Dark site (no Internet connectivity)
* Nutanix-provided Rocky Linux VM image
* AHV-based clusterWhat are two prerequisites to accomplish the deployment? (Choose two)

A. Konvoy Image Builder
B. Air-Gapped Bundle
C. Existing local container registry
D. Self-managed AWS cluster
Answer: B,C (LEAVE A REPLY)
The Nutanix Kubernetes Platform (NKP) is designed to simplify the deployment and management of Kubernetes clusters on Nutanix infrastructure, including on-premises AHV-based clusters in dark site environments with no Internet connectivity. The requirements specified in the question- dark site, Nutanix- provided Rocky Linux VM image, and AHV-based cluster-point to a deployment scenario where the environment must be self-contained and rely on Nutanix-specific tools and resources to meet the air-gapped constraints.
According to the Nutanix Kubernetes Platform Administration (NKPA) course, deploying NKP in a dark site environment requires specific prerequisites to ensure all necessary components, such as container images, dependencies, and configuration files, are available without Internet access. The course emphasizes the use of an Air-Gapped Bundle and an existing local container registry as critical components for such deployments.
* Air-Gapped Bundle (Option B):
* The NKPA course explains that for dark site deployments, Nutanix provides an Air-Gapped Bundle, which is a comprehensive package containing all the software components, container images, and dependencies required to deploy and manage an NKP cluster without Internet connectivity. This bundle includes the Kubernetes binaries, NKP platform applications (e.g., Rook Ceph, monitoring tools), and other necessary artifacts.
* The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide specifically states: "For air-gapped environments, the Nutanix Air-Gapped Bundle is required to provide all dependencies, including container images and installation files, to deploy NKP clusters." This bundle is typically downloaded from the Nutanix Support Portal in an Internet-connected environment and then transferred to the dark site for deployment.
* The bundle ensures that the Nutanix-provided Rocky Linux VM image, which serves as the base operating system for the Kubernetes nodes, can be provisioned with all required software components. The NKPA course further notes that the Air-Gapped Bundle is tailored for AHV-based clusters, ensuring compatibility with the Nutanix hypervisor.
* Existing Local Container Registry (Option C):
* In a dark site environment, a local container registry is a prerequisite to store and distribute container images required by the NKP cluster. The NKPA course highlights that NKP relies on container images for Kubernetes components, platform applications, and user workloads. In an air-gapped setup, these images cannot be pulled from public registries like Docker Hub or Quay. io.
* The course instructs administrators to set up a local container registry (e.g., Harbor, Nexus, or a Nutanix-managed registry) and populate it with the container images included in the Air-Gapped Bundle. The Nutanix Cloud Bible reinforces this, stating: "In air-gapped deployments, a local

container registry must be pre-configured to host all required images, which are provided as part of the Nutanix Air-Gapped Bundle."
* The local container registry ensures that the Kubernetes nodes, running on the Nutanix-provided Rocky Linux VM image, can access the necessary images during cluster bootstrapping and operation. The NKPA course provides guidance on configuring the registry and integrating it with the NKP deployment process.
Incorrect Options:
* Konvoy Image Builder (Option A):
* Konvoy Image Builder is a tool associated with D2iQ's Konvoy platform, used to create custom machine images for Kubernetes deployments. While it can be used to build images for Kubernetes nodes, it is not a Nutanix-specific tool nor a prerequisite for NKP deployments. The NKPA course and NCP-CN 6.10 Study Guide do not mention Konvoy Image Builder, as NKP uses the Nutanix-provided Rocky Linux VM image, which is pre-configured for AHV-based clusters. This option is irrelevant to the Nutanix ecosystem.
* Self-managed AWS cluster (Option D):
* A self-managed AWS cluster is unrelated to the requirements of deploying NKP on an AHV-based cluster in a dark site. The question specifies an AHV-based cluster, which is Nutanix's Acropolis Hypervisor running on-premises, not a cloud-based AWS environment. The NKPA course focuses on Nutanix infrastructure (AHV, Prism Central) for NKP deployments and does not include AWS as a supported platform for this scenario. This option is incorrect as it contradicts the deployment environment.
Deployment Context:
* The Nutanix-provided Rocky Linux VM image is a pre-configured operating system image optimized for NKP deployments on AHV. The NKPA course notes that this image includes the necessary kernel settings, drivers, and configurations to run Kubernetes nodes efficiently on Nutanix infrastructure.
* The AHV-based cluster requirement indicates that the deployment leverages Nutanix's hypervisor, managed through Prism Central, to provision and manage the Kubernetes nodes. The Air-Gapped Bundle and local container registry ensure that all software components are available in the dark site, aligning with the NKPA course's guidelines for air-gapped deployments.
References:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Preparing the Environment for NKP Deployment.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Air-Gapped Deployments.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com Nutanix Support Portal, Air-Gapped Bundle Documentation: https://portal.nutanix.com Nutanix Kubernetes Platform Deployment Guide: https://www.nutanix.com
=======

NEW QUESTION: 26

A Platform Engineer for an organization needs to deploy NKP into AWS while using custom credentials for authenticating. Which flag should the engineer use when starting to bootstrap the cluster installation?
--aws-profile=<my-profile>
 B. --cloud-credentials=<my-profile>
 C. --with-aws-bootstrap-credentials=true
 D. --aws-access-key=<aws access="" key=""> --aws-secret-key=<aws secret="" key=""
></aws></aws></my-profile></my-profile>

Answer:

When deploying NKP to AWS, the bootstrap process requires AWS credentials to interact with AWS APIs for provisioning resources like EC2 instances. The NKPA course specifies that the nkp CLI supports the -- aws-profile flag to specify a custom AWS profile for authentication. This profile, defined in the AWS credentials file (~/.aws/credentials), contains the access key and secret key for the desired AWS account.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To deploy NKP on AWS with custom credentials, use the --aws-profile=<profile-name> flag during the nkp create bootstrap command to reference a specific AWS profile." This approach leverages the AWS CLI's profile management, ensuring secure and flexible credential handling. For example, the command would be nkp create bootstrap --aws-profile=my- profile --kubeconfig bootstrap-cluster.conf.
Incorrect Options:
* B. --cloud-credentials=<my-profile></my-profile>: This flag is not used by the nkp CLI. The NKPA course specifies --aws-profile for AWS.
* C. --with-aws-bootstrap-credentials=true: This flag does not exist in the NKPA documentation for NKP bootstrap.
* D. --aws-access-key=<aws access="" key=""> --aws-secret-key=<aws secret="" key=""></aws><
/aws>: While these flags may be used in some tools, the NKPA course recommends using --aws-profile to avoid hardcoding sensitive credentials.
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on AWS Deployment.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Bootstrap Configuration.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com AWS CLI Documentation: https://docs.aws.amazon.com/cli

NEW QUESTION: 27

A dev team needed to optimize their logging system to be more robust, because the CPU and memory limits were insufficient, which caused delays in log collection and processing during times of high demand.
After a deep performance analysis, they decided to increase the CPU limits from 1 to 4 and the memory from
1000Mi to 4Gi.
Which ConfigMap should the development team run with custom resource requests and limit values for fluentd?

A. bash
Copy
[nutanix@nkp-boot ~]$ cat <<EOF > configmap.yaml
apiVersion v1
kind ConfigMap
metadata
name logging-operator-logging-overrides
namespace kommander
data
values.yaml |
fluentd
resources
limits
cpu 1
memory 1000Mi
requests
cpu 4
memory 4Gi
EOF
B. bash
Copy
[nutanix@nkp-boot ~]$ cat <<EOF > configmap.yaml
apiVersion: v1
kind: ConfigMap
metadata:
name: logging-operator-logging-overrides
namespace: kommander
data:
values.yaml: |
fluentd:
resources:
limits:
cpu: 4
memory: 4Gi
requests:
cpu: 4
memory: 4Gi
EOF
C. bash
Copy
[nutanix@nkp-boot ~]$ cat <<EOF > configmap.yaml

apiVersion v1
kind ConfigMap
metadata
name logging-operator-logging-overrides
namespace kommander
data
values.yaml |
fluentd
resources
limits
cpu 4
EOF
values.yaml |
fluentd
resources
limits
cpu 4
memory 4Gi
requests
cpu 1
memory 1000Mi
EOF

D. bash

Copy

[nutanix@nkp-boot ~]$ cat <<EOF > configmap.yaml
apiVersion: v1
kind: ConfigMap
metadata:
name: logging-operator-logging-overrides
namespace: kommander
data:
values.yaml: |
fluentd
resources
limits
cpu 4
memory 1000Mi
requests
cpu 1
memory 4Gi
EOF

Answer: (SHOW ANSWER)
As outlined in the NKPA 6.10 documentation under "Customizing Resource Requests and Limits for Logging," to override the default resource values for the logging operator, a ConfigMap named logging- operator-logging-overrides in the kommander namespace is used. The values.yaml in the ConfigMap should precisely define fluentd resource limits and requests in a valid YAML format.
The correct YAML format is:
yaml
Copy
apiVersion: v1
kind: ConfigMap
metadata:
name: logging-operator-logging-overrides
namespace: kommander
data:
values.yaml: |
fluentd:
resources:
limits:
cpu: 4
memory: 4Gi
requests:
cpu: 4
memory: 4Gi
This ensures that the desired CPU and memory resources are correctly applied for the fluentd daemon, avoiding parsing errors and meeting the high-demand logging needs.
Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Configuring Logging Resources"
NCP-CN 6.10 Study Guide - "Overriding Fluentd Resources Using ConfigMap"
=======

NEW QUESTION: 28
A Cloud Engineer is deploying an NKP Cluster in AWS. The environment is for testing purposes only, so the AWS team has requested it be deployed to use a minimal set of system resources to reduce cloud subscription fees. Which two parameters should be specified when initializing a Kommander installation, using the nkp install kommander command set? (Choose two.)
A. --request-timeout
B. --wait-timeout
C. --yaml
D. --init
Answer: (SHOW ANSWER)

The NKPA course details the deployment of an NKP Management cluster on AWS using the nkp install kommander command, which installs the Kommander component responsible for fleet management. For a testing environment with minimal resource usage, the engineer can optimize the installation process by adjusting parameters that control timeouts and initialization settings, reducing overhead and ensuring the deployment completes efficiently on smaller infrastructure. The two relevant parameters are:
* --wait-timeout (Option B): This parameter sets the maximum time the nkp install kommander command waits for the Kommander components to become ready. In a testing environment with minimal resources, components may take longer to start due to limited CPU and memory. Reducing the wait timeout (e.g., --wait-timeout=10m) ensures the command does not hang indefinitely, allowing the engineer to troubleshoot if needed. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "Use
--wait-timeout with nkp install kommander to adjust the waiting period for component readiness, useful in low-resource environments."
* --init (Option D): This parameter initializes the Kommander installation with default settings optimized for minimal resource usage, suitable for a testing environment. It ensures that Kommander deploys with a lightweight configuration, reducing the resource footprint (e.g., fewer replicas, lower resource requests). The NKPA course notes: "The --init flag with nkp install kommander sets up a minimal configuration for testing purposes, minimizing resource usage on AWS." Incorrect Options:
* A. --request-timeout: This parameter is not relevant to nkp install kommander. It is typically used for API request timeouts, not installation optimization.
* C. --yaml: This parameter would specify a custom YAML configuration file, but the question asks for minimal resource usage, which is better achieved with --init for default lightweight settings.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Deploying Kommander on AWS.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Building NKP Clusters.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 29
An administrator has been trying to deploy an initial AHV-based NKP cluster in a dark site (no Internet connectivity) environment using the command shown in the question.
nkp create cluster nutanix \
--cluster-name=$CLUSTER_NAME \
--control-plane-prism-element-cluster=$PE_NAME \
--worker-prism-element-cluster=$PE_NAME \
--control-plane-subnets=$SUBNET_ASSOCIATED_WITH_PE \
--worker-subnets=$SUBNET_ASSOCIATED_WITH_PE \
--control-plane-endpoint-ip=$AVAILABLE_IP_FROM_SAME_SUBNET \
--csi-storage-container=$NAME_OF_YOUR_STORAGE_CONTAINER \

--endpoint=$PC_ENDPOINT_URL \
--control-plane-vm-image=$NAME_OF_OS_IMAGE_CREATED_BY_NKP_CLI \
--worker-vm-image=$NAME_OF_OS_IMAGE_CREATED_BY_NKP_CLI \
--registry-url=${REGISTRY_URL} \
--registry-mirror-username=${REGISTRY_USERNAME} \
--registry-mirror-password=${REGISTRY_PASSWORD} \
--kubernetes-service-load-balancer-ip-range $START_IP-$END_IP \
--self-managed
Which missing attribute needs to be added in order for the deployment?
A. --airgapped
B. --insecure
C. --registry-url
D. --registry-username
Answer: (SHOW ANSWER)
For deployments in air-gapped environments where there is no external Internet connectivity, the --airgapped parameter is required. This instructs the NKP deployment to rely solely on internal resources, using pre- staged images and local container registries, ensuring that no external network dependencies cause deployment failures.
References: Nutanix Kubernetes Platform Administration Guide - Air-gapped Cluster Deployment Requirements

NEW QUESTION: 30
A Cloud Engineer is deploying an NKP management cluster and plans to deploy multiple NKP workload clusters from it. The management cluster will be on Nutanix infrastructure, but the NKP workload clusters may be deployed in multiple provisioning environments, such as:
* Nutanix
* AWS
* AzureWhen the engineer deploys an NKP workload cluster in AWS, which two default behaviors will be performed by NKP on this newly-deployed cluster? (Choose two.)
A. The NKP workload cluster will receive all of the GitOps sources that have been assigned to the NKP workspace.
B. The NKP workload cluster will receive all of the NKP RBAC policy that has been assigned to this NKP workspace.
C. The NKP workload cluster will also be assigned to all of the NKP projects that have been created within the NKP workspace.
D. The NKP workload cluster will be deployed all of the applications that have been enabled on this NKP workspace.
Answer: (SHOW ANSWER)
The NKPA course outlines the default behaviors of NKP when deploying workload clusters within a workspace, regardless of the provisioning environment (e.g., AWS, Nutanix, Azure). When a

workload cluster is deployed in AWS, NKP ensures consistency across the workspace by applying the following default behaviors:
* The NKP workload cluster will receive all of the GitOps sources that have been assigned to the NKP workspace (Option A): NKP uses GitOps with Flux to manage cluster configurations and applications. The course explains that GitOps sources (e.g., Git repositories) assigned to a workspace are automatically applied to all clusters in that workspace, ensuring consistent configuration and application deployment. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "New workload clusters inherit all GitOps sources configured for the workspace, enabling Flux to synchronize configurations from the specified repositories."
* The NKP workload cluster will be deployed all of the applications that have been enabled on this NKP workspace (Option D): NKP Platform Applications (e.g., Prometheus, Rook Ceph) enabled in the workspace are automatically deployed to new workload clusters. The NKPA course notes: "When a workload cluster is created, NKP deploys all platform applications enabled in the workspace to ensure consistent functionality across clusters." Incorrect Options:
* B. The NKP workload cluster will receive all of the NKP RBAC policy: RBAC policies are defined at the workspace or project level and applied to users or groups, not automatically to clusters. Clusters inherit RBAC through user access, not as a default deployment behavior.
* C. The NKP workload cluster will also be assigned to all of the NKP projects: Clusters are assigned to specific projects manually, not automatically to all projects in a workspace.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Workload Cluster Deployment.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Workspace Management.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 31
Refer to Exhibit:

Using an NKP Ultimate license, a Platform Engineer has created a new workspace and needs to create a new Kubernetes cluster within this workspace. However, the engineer discovers that the

Create Cluster option is grayed out, as shown in the exhibit. How should the engineer resolve this issue?

A. Create the cluster only using YAML and not the GUI.

B. Attach existing clusters instead of creating a new cluster.

C. Create an Infrastructure provider for the workspace.

D. Ensure NKP is upgraded to a minimum version of 2.12.

Answer: (SHOW ANSWER)

The exhibit shows the NKP UI with the "Create Cluster" option grayed out in a new workspace, indicating that a prerequisite for cluster creation is missing. The NKPA course explains that to create a new Kubernetes cluster in a workspace, an Infrastructure Provider must be configured for that workspace. An Infrastructure Provider defines the underlying infrastructure (e.g., Nutanix AHV, AWS, vSphere) where the cluster will be provisioned, and without it, the Create Cluster option remains disabled in the UI.

The NKP Ultimate license supports creating clusters across various infrastructures, but the workspace must be associated with an Infrastructure Provider to enable cluster creation. The Nutanix Cloud Native (NCP-CN)

6.10 Study Guide states: "Before creating a cluster in a new workspace, ensure an Infrastructure Provider is configured for the workspace in the NKP UI; otherwise, the Create Cluster option will be grayed out." The engineer should navigate to the Infrastructure Providers section in the NKP UI (typically under the Global or Administration view), create a provider (e.g., for Nutanix AHV, AWS, or vSphere), and associate it with the workspace. Once this is done, the Create Cluster option will become available.

Incorrect Options:

* A. Create the cluster only using YAML and not the GUI: The issue is not with the GUI itself but with the missing Infrastructure Provider, which affects both GUI and CLI cluster creation. YAML alone does not resolve this.
* B. Attach existing clusters instead of creating a new cluster: Attaching existing clusters is an alternative but does not address the requirement to create a new cluster.
* D. Ensure NKP is upgraded to a minimum version of 2.12: The NKP Ultimate license implies a recent version, and the course does not indicate that version 2.12 is specifically required for this functionality. The issue is configuration-related, not version-related.

:

Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Cluster Creation Prerequisites.

Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Workspace Configuration.

Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 32
By default, a full cluster backup is occurring on a daily basis on an NKP-managed cluster. However, the Victory Finance team has requested that their applications be backed up four times a day. The user group has been assigned to the victory-finance-apps NKP workspace, and the applications have been deployed to the Kubernetes namespace victory-finance.
What is the procedure for scheduling the team's application backups?
A. Access the NKP UI
Switch to the victory-finance NKP namespace
Click on the Applications menu panel and select Velero
Edit the Velero application configuration at the cluster level for the specific NKP managed cluster (not all clusters in that NKP workspace) Set the backup frequency to four times daily
B. Install the Velero CLI
Use the Velero CLI to create a backup schedule that includes the following parameters:
- Backup name
- Kubernetes namespace to backup
- Backup frequency (every six hours)
- Managed NKP cluster's kubeconfig
C. Install the Velero CLI
Use the Velero CLI to create a backup schedule that includes the following parameters:
- Backup name
- NKP workspace to backup
- Backup frequency (every six hours)
- Managed NKP cluster's kubeconfig
D. Access the NKP UI
Switch to the victory-finance-apps NKP workspace
Click on the Applications menu panel and select Velero
Edit the Velero application configuration at the cluster level for the specific NKP managed cluster (not all clusters in that NKP workspace) Set the backup frequency to four times daily
Answer: B (LEAVE A REPLY)
As per the NKPA 6.10 documentation under "Application Backup and Recovery with Velero," Velero is the out-of-the-box backup and restore solution integrated with NKP. To schedule application backups at a specific frequency (in this case, every six hours, equivalent to four times daily), the recommended approach is to use the Velero CLI to create a backup schedule.
The procedure involves:
* Using the Velero CLI to create a scheduled backup targeting the specific Kubernetes namespace (victory-finance) rather than the entire workspace.
* Specifying the frequency as --schedule="0 */6 * * *" (every six hours).

* Providing the cluster's kubeconfig for authentication.
Exact extract from the documentation:
"Velero's CLI can be used to create backup schedules for application-specific workloads. Use the namespace parameter to target the namespace where the applications are deployed, and set the cron-like schedule to the desired frequency." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Using Velero CLI to Create Backup Schedules" NCP-CN 6.10 Study Guide - "Application-Level Backup Best Practices"
=======

NEW QUESTION: 33
A company has 30 Edge devices with lightweight Kubernetes, and developers need to push the application to every edge device. An NKP administrator has the NKP Ultimate license tier configured and has access to all kubeconfig files for the 30 edge devices. What is the most efficient way that the administrator can lifecycle manage the application deployments?
A. Create a GitHub configuration, deploy it to the 30 edge devices, and configure them to use a GitHub account.
B. Create a script to automate the deployment to every edge device.
C. Ask the developers to delete the lightweight Kubernetes and deploy new Kubernetes clusters with NKP.
D. Create a new Workspace and attach the 30 edge devices to this workspace with Attach Cluster.
Answer: D (LEAVE A REPLY)
The NKPA course emphasizes that NKP's fleet management capabilities allow administrators to manage multiple Kubernetes clusters, including lightweight clusters on edge devices, by attaching them to an NKP workspace. With the NKP Ultimate license tier, the administrator has access to advanced fleet management features, including GitOps-based application deployment across attached clusters.
The most efficient way to lifecycle manage application deployments across the 30 edge devices is to create a new workspace and attach the 30 edge devices to this workspace using the Attach Cluster functionality (Option D). Once attached, NKP can use GitOps (via Flux) to push applications to all clusters in the workspace simultaneously, ensuring consistent deployment and lifecycle management. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "For managing edge devices with lightweight Kubernetes, attach the clusters to an NKP workspace using the Attach Cluster feature, enabling centralized application deployment and lifecycle management via GitOps." The administrator can use the kubeconfig files to attach each cluster via the NKP UI or CLI (e.g., nkp attach cluster).
Incorrect Options:
* A. Create a GitHub configuration for each device: This is manual and inefficient compared to NKP's centralized GitOps management.
* B. Create a script to automate deployment: Scripting is error-prone and lacks NKP's built-in fleet management capabilities.

* C. Delete lightweight Kubernetes and deploy new clusters: This is unnecessary, as NKP can manage existing clusters, including lightweight ones.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Fleet Management.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Attaching Clusters.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 34
A development team has decided to implement an efficient logging system and use AWS S3 as storage to manage large volumes of logs in a scalable way.
The team followed these steps:
* Set the WORKSPACE_NAMESPACE variable to the namespace copied in the previous step.
* Created a config that overrode ConfigMap to update the storage configuration.
* Updated the grafana-loki AppDeployment to apply the configuration override.However the implementation failed.What should the team do to be able to manage log storage in AWS?
A. Configure an IP address corresponding to AWS storage.
B. Create a secret containing the static AWS S3 credentials.
C. Create a tenant on AWS.
D. Configure a new IAM role specifically for NKP.
Answer: B (LEAVE A REPLY)
As stated in the NKPA 6.10 documentation, when using external storage (such as AWS S3) with Loki for log storage, AWS credentials must be provided securely. This typically involves creating a Kubernetes Secret containing the static AWS credentials (access key ID and secret access key), which are referenced in the override ConfigMap to authenticate Loki's S3 storage backend.
Key reference from documentation:
"For Loki to store logs in an S3 bucket, AWS credentials must be created as a Kubernetes secret and referenced in the storage configuration." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Loki External Storage Configuration"
NCP-CN 6.10 Study Guide - "Using External Storage Backends with Logging"

NEW QUESTION: 35
A development team decided to employ an efficient monitoring system with Grafana-logging, which was successfully implemented as can be seen in the following output:
AppDeployment "kommander-default-workspace/grafana-logging" created in namespace "kommander- default-workspace".
Which command did the team execute to complete this task?
A. nkp create appdeployment grafana-logging --app grafana-logging-6.57.4 --workspace default-workspace
B. export WORKSPACE_NAMESPACE=kommander-default-workspace
appdeployment
nkp create package-bundle grafana-logging

C. kubectl get appdeployment -n kommander-default-workspace
D. kubectl get helmreleases grafana-logging -n kommander-default-workspace -w
Answer: (SHOW ANSWER)
The output indicates that an AppDeployment resource named grafana-logging was created in the kommander- default-workspace namespace, which corresponds to an NKP workspace. The NKPA course explains that NKP Platform Applications, such as Grafana Logging, are deployed using the nkp create appdeployment command. This command creates an AppDeployment resource that deploys the specified application (e.g., Grafana Logging) across all clusters in a workspace.
The command in Option A, nkp create appdeployment grafana-logging --app grafana-logging-6.57.4 -- workspace default-workspace, matches the output:
* grafana-logging is the name of the AppDeployment.
* --app grafana-logging-6.57.4 specifies the application and version.
* --workspace default-workspace targets the workspace, which corresponds to the namespace kommander-default-workspace in the output (the kommander- prefix is a standard NKP namespace convention).
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To deploy a platform application like Grafana Logging, use nkp create appdeployment <app-name> --app <app-id>-<version> --workspace
<workspace-name>, which creates an AppDeployment resource in the workspace namespace." This command aligns with the output provided.
Incorrect Options:
* B. export WORKSPACE_NAMESPACE and nkp create package-bundle: There is no nkp create package-bundle command in NKP for deploying applications. The correct command is nkp create appdeployment.
* C. kubectl get appdeployment: This retrieves the status of an AppDeployment, not creates it. The output indicates creation, not retrieval.
* D. kubectl get helmreleases: This retrieves HelmRelease resources, not AppDeployments, and does not create the Grafana Logging deployment.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Platform Application Deployment.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 36
A Platform Engineer is a member of an IT team that provides Kubernetes clusters for three groups within a company named Fin Group, Inc.:
* Fin VD
* Fin Insurance

* Fin TravelThe engineer created workspaces for each group. Fin Group Inc. has its own Active Directory implementation, while each group uses their own Identity Provider. Now, the engineer needs to assign the Tenant Administrators role for each workspace. How will the engineer complete this task?

A. Configure a dedicated identity provider for each group to access their own workspace.

B. Create a role named admin-tenant-X, where X is the name of the group, and assign that role to manage the corresponding workspace.

C. Create a role binding and assign it to manage the corresponding workspace.

D. Configure the global Active Directory and assign a workspace admin user to each group.

Answer: A (LEAVE A REPLY)

The NKPA course emphasizes that NKP supports multi-tenancy through workspaces, each of which can be configured with its own Identity Provider (IdP) for authentication. In this scenario, each group (Fin VD, Fin Insurance, Fin Travel) has its own IdP, and the engineer has created separate workspaces for them. To assign the Tenant Administrators role, the engineer must configure a dedicated IdP for each group's workspace, enabling users to authenticate via their group-specific IdP and assume the Tenant Administrator role.

The course details that NKP uses Dex as the OIDC provider to integrate with external IdPs. For each workspace, the engineer configures a Dex connector to the group's IdP, maps the IdP groups to NKP roles (e.

g., Tenant Administrator), and assigns permissions via role bindings. The Nutanix Cloud Native (NCP-CN)

6.10 Study Guide states: "To assign Tenant Administrator roles in NKP, configure a dedicated IdP for each workspace using Dex connectors, mapping IdP groups to the appropriate roles for workspace management." This ensures that each group's administrators can access only their designated workspace.

Incorrect Options:

* B. Create a role named admin-tenant-X: NKP uses predefined roles like Tenant Administrator, not custom roles with specific naming conventions.

* C. Create a role binding and assign it: Role bindings are part of the process, but the primary step is configuring the IdP for authentication, as per the NKPA course.

* D. Configure the global Active Directory: The scenario specifies separate IdPs per group, not a global Active Directory for all groups.

:

Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Workspace and Role Management.

Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Authentication and Authorization.

Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 37

In which unit are NKP licenses able to be obtained?

A. Flash

B. CPU Sockets
C. TiBs
D. CPU Cores
Answer: D (LEAVE A REPLY)
The NKPA course clarifies that NKP licenses are based on the number of CPU cores in the infrastructure hosting the Kubernetes clusters. This licensing model applies to both on-premises (e.g., Nutanix AHV) and cloud-based deployments. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "NKP licenses are obtained based on the total number of CPU cores allocated to the clusters managed by the platform." This core-based licensing ensures flexibility across different infrastructure types while aligning with the resource consumption of Kubernetes workloads.
Incorrect Options:
* A. Flash: Flash storage is not a licensing unit for NKP.
* B. CPU Sockets: Nutanix licenses for other products may use sockets, but NKP uses cores.
* C. TiBs: Terabytes are used for storage licensing, not NKP.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Licensing.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on NKP Licensing.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 38
A Platform Engineer is deploying an NKP workload cluster using the nkp create cluster vsphere command.
The cluster will be utilized by the company's code-green team and the engineer has already created a code- green NKP workspace on the NKP management cluster.
After issuing the deploy command, the engineer monitored the build using the nkp describe cluster command and confirmed it completed successfully. However, a few hours later, after logging into the NKP UI, the engineer checked the code-green NKP workspace and saw that the NKP workload cluster was not there.
What is the likely reason the NKP workload cluster is not in the code-green NKP workspace?
A. The vSphere cluster cannot be displayed in the NKP UI unless its Kubernetes version is within 'N - 1' versions of the NKP management cluster's Kubernetes version.
B. The vSphere service account credentials had expired prior to the engineer's attempt to view the cluster in the NKP UI. Once the credentials are refreshed, the vSphere cluster will reappear in the NKP workspace.
C. The engineer did not supply the --namespace code-green parameter as part of the nkp create cluster vsphere command, therefore it was created in the default workspace and needs to be manually attached.
D. NKP vSphere clusters cannot be assigned NKP workspaces and instead are assigned the default NKP workspace. The cluster can be viewed from this workspace instead.
Answer: C (LEAVE A REPLY)

The NKPA 6.10 documentation clarifies that when creating a workload cluster using the nkp create cluster command, specifying the target workspace (namespace) is critical for properly associating the workload cluster with that workspace in the NKP UI. If the --namespace <workspace> parameter is omitted, the cluster is provisioned in the default workspace, not in the intended workspace (in this case, code-green).
Key documentation excerpt:
"If you do not specify the workspace (namespace) using the --namespace parameter when creating a cluster, the cluster will be created in the default workspace. It will not appear in custom workspaces until manually assigned." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Creating Workload Clusters" NCP-CN 6.10 Study Guide - "Namespace and Workspace Mapping for Workload Clusters"
=======

NEW QUESTION: 39
A Platform Engineer is deploying an NKP cluster within an air-gapped AWS environment. However, after an infrastructure planning session with the network team, it's been determined that the default CIDR block range that is used by pods on NKP clusters is already in use in their environment.
How can the engineer ensure there are no collisions between NKP pod traffic and the existing network using that subnet range?
A. Because the environment is air-gapped, there will be no network traffic collision concerns and no adjustment needs to be made to the pod network CIDR block range.
B. Create an NKP infrastructure provider for AWS in the NKP UI.
Select the Advanced Options button from the Network section of the Create Cluster page and specify a unique CIDR block range within the pod network field.
C. Create the NKP cluster's manifest using the nkp create cluster command set and include the pod CIDR block range parameter when generating the cluster manifest.
Deploy the NKP cluster manifest.
D. Create an NKP infrastructure provider for AWS in the NKP UI.When deploying the NKP cluster through the UI, specify a unique CIDR block range for the pod network field in the Network section of the Create Cluster page.
Answer: C (LEAVE A REPLY)
The NKPA 6.10 documentation outlines that in air-gapped environments where default pod network CIDR conflicts exist, the cluster manifest (created via the nkp create cluster command) should be updated to specify an alternate, non-conflicting pod CIDR block range before deployment.
Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Customizing Pod Network CIDR Blocks in Air- gapped Clusters" NCP-CN 6.10 Study Guide - "Networking Configuration for Air-gapped Environments"
=======

NEW QUESTION: 40
A Platform Engineer would like to install some NKP applications, but with a few modifications to the default configuration specs of some of the components. Additionally, Velero itself can be disabled, as the company already utilizes a different backup utility for Kubernetes.
Which procedure would the engineer utilize to accomplish these goals when deploying the applications?
A. Execute nkp install kommander --init to an output file.
Set the custom specs for the components to be modified in the output file.
Deploy the NKP applications using the nkp install kommander command, specifying the output file.
Once the NKP applications install has completed, execute kubectl delete hr -n kommander velero.
B. Execute nkp install kommander --init to an output file.
Disable Velero in the output file and set the custom specs for the components to be modified.
Deploy the NKP applications using the nkp install kommander command, specifying the output file.
C. Execute nkp config kommander --init to an output file.
Disable Velero in the output file and set the custom specs for the components to be modified.
Deploy the applications using the helm install command, specifying the output file.
D. Execute nkp config kommander.
Disable Velero in the resulting output file and set the custom specs for the components to be modified.
Deploy the NKP applications using the nkp install kommander command, specifying the output file.
Answer: B (LEAVE A REPLY)
The NKPA 6.10 documentation confirms that the proper method to customize application deployment (including disabling Velero) is by generating a configuration file using the nkp install kommander --init command. The file can then be modified to set custom specs and disable Velero, and finally deployed using the nkp install kommander command.
Exact extract:
"Generate a custom configuration file using --init, modify it for your specific environment (including disabling Velero), and deploy with the same nkp install kommander command." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Customizing Kommander Installation"
NCP-CN 6.10 Study Guide - "Deploying and Customizing NKP Applications"
=======

NEW QUESTION: 41
A Cloud Engineer is deploying an NKP cluster into an AWS environment. By default, when deploying NKP on AWS infrastructure, it generates the supporting infrastructure necessary for the cluster (VPC, subnets, ELBs). However, the AWS team has insisted that the NKP cluster be deployed on existing AWS infrastructure. How can the engineer meet this requirement?

A. When using the nkp adopt infrastructure aws command set, include the valid parameters with the pre- existing VPC, subnets, and ELB to use. Deploy the NKP cluster using the nkp create cluster aws command set.
B. Create an overrides file with the pre-existing VPC, subnets, and ELB to use. When using the nkp create cluster aws command set, include the overrides parameter with the overrides file that was created.
C. Create an NKP infrastructure provider for AWS in the NKP UI. When deploying the NKP cluster through the UI, specify the pre-existing VPC, subnets, and ELB to use in the appropriate fields of the
'Infrastructure' section of the Create Cluster page.
D. When using the nkp create cluster aws command set, include the valid parameters with the pre-existing VPC, subnets, and ELB to use.
Answer: B (LEAVE A REPLY)
The NKPA course explains that by default, NKP creates new AWS infrastructure (VPC, subnets, ELBs) when deploying a cluster on AWS. However, NKP supports deploying clusters on existing AWS infrastructure by providing custom configurations. The recommended method using the NKP CLI is to create an overrides file specifying the pre-existing VPC, subnets, and ELB, and then pass this file to the nkp create cluster aws command using the --overrides parameter.
The overrides file (e.g., aws-infra-overrides.yaml) contains details like vpcID, subnetIDs, and loadBalancerIDs, which NKP uses to deploy the cluster on the specified infrastructure instead of creating new resources. For example:
yaml
CollapseWrap
Copy
aws:
vpcID: vpc-12345678
subnetIDs:
- subnet-12345678
- subnet-87654321
loadBalancerIDs:
- elb-12345678
The engineer then runs: nkp create cluster aws --overrides aws-infra-overrides.yaml.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To deploy an NKP cluster on existing AWS infrastructure, create an overrides file with the pre-existing VPC, subnets, and ELB details, and use the -- overrides parameter with the nkp create cluster aws command to apply the custom configuration." This method ensures the AWS team's requirement is met while leveraging NKP's CLI for deployment.
Incorrect Options:
* A. nkp adopt infrastructure aws: There is no nkp adopt infrastructure command in NKP for this purpose.

* C. Use the NKP UI: While the UI allows specifying infrastructure details, the question focuses on the CLI-based deployment, and the UI method is less relevant here.
* D. Include parameters directly in nkp create cluster aws: The nkp create cluster aws command does not support direct parameters for VPC, subnets, and ELB; it requires an overrides file.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on AWS Cluster Deployment.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Building NKP Clusters.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

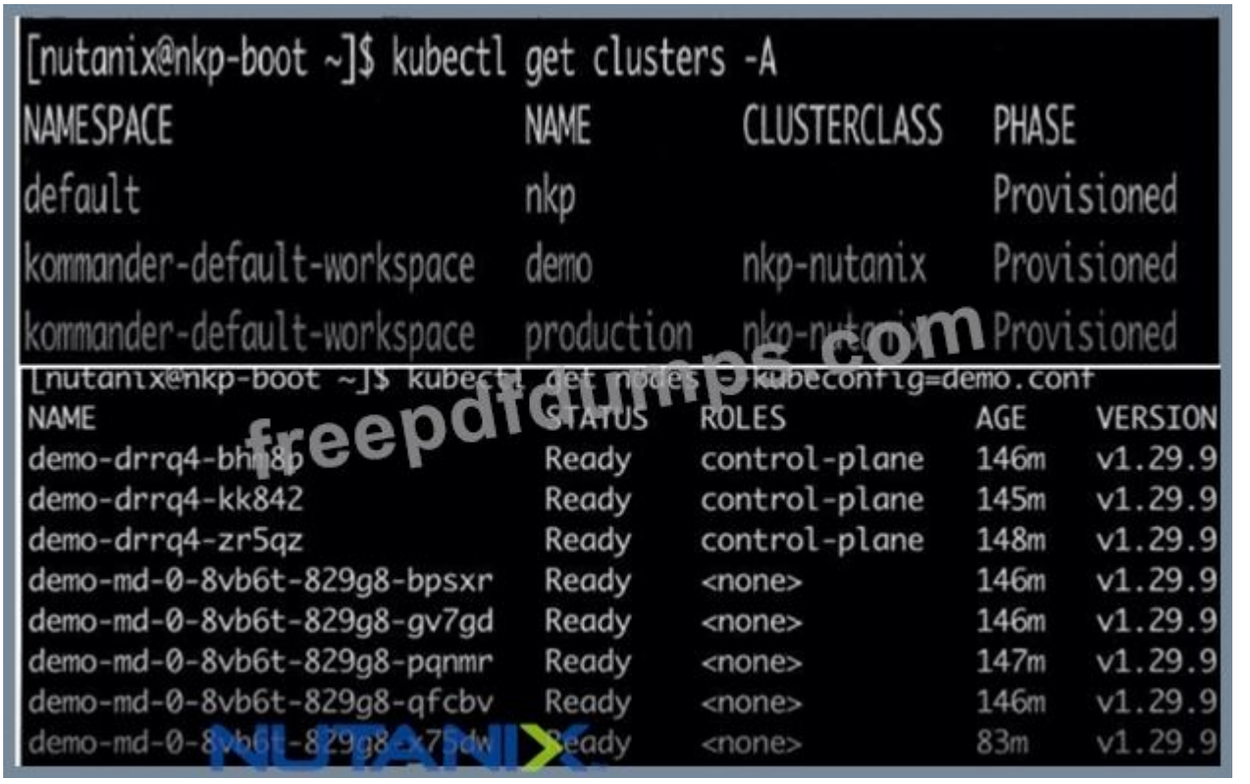
NEW QUESTION: 42
A company is developing a new web application consisting of several microservices, including:
* Authentication service
* User management service
* Payment processing service Each microservice is developed by different teams and requires an isolated environment for testing and development. To facilitate development and testing, the team decides to create a specific workspace in NKP. What should the team do to start this new creation?
A. From the Cluster selection, select Add Cluster.
B. From the workspace selection dropdown list in the top menu bar, select Create Workspace.
C. From the workspace selection dropdown list in the top menu bar, select Add Workspace.
D. From the Administration selection dropdown list in Infrastructure Providers, select Add Infrastructure Provider.
Answer: (SHOW ANSWER)
The Nutanix Kubernetes Platform (NKP) uses workspaces to provide isolated environments for different teams or projects, allowing each team to manage its own clusters, applications, and resources independently.
According to the NKPA course, creating a new workspace is a key Day 2 operation to support multi-tenancy and isolated development environments, such as those required for the microservices in this scenario.
The course specifies that to create a new workspace, users must navigate to the workspace selection dropdown list in the top menu bar of the NKP user interface (UI) and select Create Workspace. This action opens a form where administrators can define the workspace name, description, and associated resources (e.
g., clusters, users, and policies). The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To create a new workspace in NKP, go to the workspace selection dropdown in the UI and select 'Create Workspace' to configure an isolated environment for a team or project." This process ensures that each microservice team has its own isolated environment for development and testing, with access restricted to their specific workspace.
Incorrect Options:

* A. From the Cluster selection, select Add Cluster: Adding a cluster creates a new Kubernetes cluster within an existing workspace, not a new workspace. The NKPA course distinguishes between cluster and workspace creation.
* C. From the workspace selection dropdown list in the top menu bar, select Add Workspace: The NKPA course and UI use "Create Workspace" as the standard terminology, not "Add Workspace."
* D. From the Administration selection dropdown list in Infrastructure Providers, select Add Infrastructure Provider: This option is for configuring infrastructure providers (e.g., AWS, vSphere) for NKP, not for creating workspaces.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Workspace Management.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 43
To keep an NKP cluster and applications healthy and drive productivity forward, a Platform Engineer needs to stay informed of all events occurring within the cluster. What component of kube-prometheus-stack will help the engineer to stay informed of these events in NKP?
A. prometheus-operator
B. service monitors
C. alertmanager
D. node-exporter
Answer: C (LEAVE A REPLY)
The kube-prometheus-stack is a key component of NKP's monitoring stack, providing tools for metrics collection, visualization, and alerting. The NKPA course explains that Alertmanager, a component of the kube-prometheus-stack, is responsible for handling alerts generated from Prometheus metrics. It aggregates, deduplicates, and routes notifications to the appropriate channels (e.g., email, Slack, PagerDuty), ensuring that the Platform Engineer stays informed of critical events and anomalies in the NKP cluster, such as node failures, resource exhaustion, or application errors.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "Alertmanager in the kube-prometheus-stack processes alerts from Prometheus, enabling administrators to stay informed of cluster events through configured notification channels." By configuring Alertmanager with appropriate routing rules and receivers, the engineer can receive real-time notifications about cluster events, driving proactive management and productivity.
Incorrect Options:
* A. prometheus-operator: The operator manages Prometheus and related resources but does not directly handle event notifications.
* B. service monitors: Service monitors define how Prometheus scrapes metrics, not how events are communicated.

* D. node-exporter: Node-exporter collects node-level metrics, not event notifications.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Monitoring and Alerting.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com
Prometheus Documentation: https://prometheus.io/docs/alerting/alertmanager

NEW QUESTION: 44
Refer to the exhibits.
A Cloud Administrator had provisioned a Kubernetes cluster named demo that is no longer actively being used. A Quick review from the Systems Engineer confirms that the following VMs are part of the demo Kubernetes cluster



How should the demo cluster be properly deleted?
A. Run nkp delete cluster -c demo -n kommander-default-workspace
B. Delete all the VMs and inform of the results.
C. Run acli vm.delete demo* from a CVM.
D. Run kubectl config delete-cluster demo -n kommander-default-workspace and delete the VMs.
Answer: (SHOW ANSWER)
Comprehensive and Detailed Explanation:
The correct procedure for deleting an NKP cluster involves using the nkp delete cluster command with the appropriate cluster name and namespace. This ensures that not only the Kubernetes resources but also the corresponding NKP resources (e.g., nodepools, Kommander integration) are deleted cleanly and consistently.

Simply deleting the VMs does not clean up the associated NKP management objects. This approach is detailed in the NKP documentation for cluster lifecycle management, emphasizing the need to use the provided CLI commands for full removal.
References: NCP-CN-6.10 Course Material - Cluster Deletion Commands

NEW QUESTION: 45
Which procedure should a Platform Engineer follow for setting up user authentication into an NKP cluster?
A. Enable Gatekeeper and create a connector to the user base's identity provider.
B. Disable native NKP authentication, enable Traefik, and create a connector to the user base's identity provider.
C. Create a MetalLB connector to the user base's identity provider.
D. Create a Dex connector to the user base's identity provider.
Answer: D (LEAVE A REPLY)
The NKPA course covers user authentication for NKP clusters as part of Day 2 operations, emphasizing integration with external identity providers (IdPs) to manage user access securely. NKP uses Dex, an OpenID Connect (OIDC) identity provider, to facilitate authentication by acting as a connector between the Kubernetes cluster and external IdPs, such as LDAP, SAML, or OAuth-based systems.
The course explains that to set up user authentication, a Platform Engineer must configure a Dex connector to the user base's identity provider. Dex integrates with the Kubernetes API server to enable OIDC-based authentication, allowing users to log in using their IdP credentials. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "NKP supports user authentication through Dex, which provides OIDC integration with external identity providers, enabling single sign-on (SSO) for cluster access." The process involves deploying Dex as a platform application, configuring the IdP connector (e.g., specifying client IDs, secrets, and endpoints), and updating the Kubernetes API server to use OIDC authentication.
Incorrect Options:
* A. Enable Gatekeeper and create a connector to the user base's identity provider: Gatekeeper is a Kubernetes policy engine used for enforcing admission control policies, not for authentication. The NKPA course does not associate Gatekeeper with user authentication.
* B. Disable native NKP authentication, enable Traefik, and create a connector to the user base's identity provider: Traefik is an ingress controller for managing external traffic, not authentication. Disabling native authentication is unnecessary, as NKP supports OIDC alongside native methods. The NKPA course does not mention Traefik in the context of authentication.
* C. Create a MetalLB connector to the user base's identity provider: MetalLB is a load balancer for bare-metal Kubernetes clusters, not an authentication component. This option is irrelevant, as per the NKPA course.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on User Authentication and Authorization.

Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on NKP Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com Dex Documentation: https://dexidp.io

NEW QUESTION: 46
At a national defense company, protecting sensitive data is their top priority. With the increase in cyber- attacks, they have decided to implement an air-gapped Kubernetes environment to manage their critical applications, ensuring that no information could leak to the outside. The Kubernetes environment has three clusters deployed for their applications with centralized management. What type of licensing is required to enable this environment?
A. NKP Starter
B. NKP Ultimate
C. NKP Pro
D. NKP UI
Answer: (SHOW ANSWER)
The NKPA course specifies that air-gapped deployments and centralized fleet management of multiple clusters are advanced features of NKP, requiring the NKP Ultimate license tier. The Ultimate tier includes support for air-gapped environments (via Air-Gapped Bundles) and fleet management capabilities, such as attaching and managing multiple clusters under a single management plane, which is critical for the national defense company's scenario.
The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "The NKP Ultimate license tier is required for air-gapped deployments and centralized management of multiple Kubernetes clusters, providing the necessary tools for secure, isolated environments." This ensures the company can deploy and manage their three clusters in an air-gapped setup while maintaining strict data security.
Incorrect Options:
* A. NKP Starter: The Starter tier lacks air-gapped and fleet management features.
* C. NKP Pro: The Pro tier may support some advanced features but not air-gapped deployments or full fleet management.
* D. NKP UI: This is not a license tier; it's a UI component of NKP.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Licensing and Fleet Management.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Air-Gapped Deployments.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 47
A Platform Engineer is running a Kubernetes cluster version 1.28.1 on AWS that needs to be upgraded to version 1.29.9. This cluster was deployed with Nutanix NKP. Which two actions should the engineer take to complete this requirement? (Choose two.)
A. Upgrade Workers with nkp update nodepool aws ${NODEPOOL_NAME} --cluster- name= ${CLUSTER_NAME} --kubernetes-version=v1.29.9
B. Upgrade Control Planes with nkp update controlplane aws --cluster-name= ${CLUSTER_NAME} -- ami AMI_ID --kubernetes-version=v1.29.9
C. Upgrade Workers with nkp upgrade nodepool aws ${NODEPOOL_NAME} --cluster- name= ${CLUSTER_NAME} --kubernetes-version=v1.29.9
D. Upgrade the Cluster with nkp update cluster aws --cluster-name=${CLUSTER_NAME} --ami AMI_ID
--kubernetes-version=v1.29.9
Answer: (SHOW ANSWER)
The NKPA course details the process for upgrading an NKP-managed Kubernetes cluster, such as one running on AWS from version 1.28.1 to 1.29.9. Upgrading a Kubernetes cluster involves two distinct steps:
upgrading the control plane nodes and upgrading the worker nodes, ensuring minimal disruption and maintaining compatibility. The NKP CLI provides specific commands to handle these upgrades separately for AWS clusters.
* Upgrade Control Planes with nkp update controlplane aws --cluster-name=${CLUSTER_NAME} --ami AMI_ID --kubernetes-version=v1.29.9 (Option B):The control plane must be upgraded first to the target Kubernetes version (1.29.9). The nkp update controlplane aws command updates the control plane nodes, specifying the cluster name, the new Kubernetes version, and an updated AMI (Amazon Machine Image) that supports the target version. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "To upgrade an NKP cluster on AWS, first update the control plane using nkp update controlplane aws --cluster-name <name> --ami <ami-id> --kubernetes-version <version> to ensure the control plane runs the desired Kubernetes version." The --ami flag is required to specify a compatible image for the upgraded control plane nodes.
* Upgrade Workers with nkp upgrade nodepool aws ${NODEPOOL_NAME} --cluster- name= ${CLUSTER_NAME} --kubernetes-version=v1.29.9 (Option C):After the control plane is upgraded, the worker nodes in each node pool must be upgraded to match the control plane version.
The nkp upgrade nodepool aws command updates the specified node pool to the target Kubernetes version (1.29.9). The NKPA course notes: "Upgrade worker nodes using nkp upgrade nodepool aws
<nodepool-name> --cluster-name <cluster-name> --kubernetes-version <version>, which performs a rolling update to ensure minimal downtime." This command automatically handles the

rolling update of worker nodes, replacing them with new nodes running the updated version. Note that the --ami flag is not required here, as NKP typically uses the same AMI as the control plane or retrieves a compatible one based on the version.
Incorrect Options:
* A. nkp update nodepool aws: The correct command is nkp upgrade nodepool, not nkp update nodepool. The NKPA course uses upgrade for version changes to node pools.
* D. nkp update cluster aws: This command is not the standard approach for upgrading Kubernetes versions in NKP. The course specifies separate commands for control plane and node pool upgrades to ensure a controlled process.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Cluster Upgrades.
Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Day 2 Operations.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 48
Refer to the exhibit.

An administrator is provisioning an NKP cluster. After the VM creation task, the error shown in the exhibit is produced.
What could be the reason?
A. Private registry software or version is not the recommended.
B. VM does not have the Linux version.
C. VM doesn't have communication to the registry.
D. NKP Software is not loaded in the registry.
Answer: C (LEAVE A REPLY)
The error states:
pgsql
Copy
error upgrading CAPI components: unable to upgrade CAPI components: deployment "capp-controller- manager" is not ready after 10m0s: failed to connect to the management cluster: context deadline exceeded This clearly points to connectivity issues between the VM (or nodes) and the management cluster, typically caused by registry communication issues in air-gapped or

private environments. When the VM cannot connect to the registry to pull required images or configuration, the CAPI (Cluster API) components cannot be initialized, causing a timeout.
Key Reference:
* Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Air-Gapped and Registry Communication Issues"
* NCP-CN 6.10 Study Guide - "Cluster API Upgrade Process and Network Prerequisites"
=======

NEW QUESTION: 49
A Platform Engineer is attempting to delete an attached cluster from the NKP UI, but it is stuck in a 'deleting' state and does not get removed. How can the engineer resolve this attempt to detach the cluster so that it is removed from the UI and no longer managed by NKP?
A. Run the kubectl delete cluster command in the context of the NKP management cluster.
B. Run the nkp delete kommandercluster command in the context of the NKP attached cluster.
C. Run the kubectl delete kommandercluster command in the context of the NKP management cluster.
D. Run the nkp delete cluster command in the context of the NKP attached cluster.
Answer: C (LEAVE A REPLY)
When an attached cluster (e.g., an external cluster like EKS) is stuck in a 'deleting' state in the NKP UI, it indicates an issue with the reconciliation process in the NKP management cluster. The NKPA course explains that attached clusters are represented in NKP as KommanderCluster custom resources in the management cluster. To resolve a stuck deletion, the engineer must manually delete the KommanderCluster resource using kubectl in the context of the NKP management cluster.
The correct command is kubectl delete kommandercluster, executed in the context of the NKP management cluster (not the attached cluster). For example: kubectl delete kommandercluster <cluster-name> -n
<namespace>. The Nutanix Cloud Native (NCP-CN) 6.10 Study Guide states: "If an attached cluster is stuck in a 'deleting' state, delete the corresponding KommanderCluster resource in the NKP management cluster using kubectl delete kommandercluster to remove it from management." This ensures the cluster is fully detached and removed from the UI.
Incorrect Options:
* A. kubectl delete cluster: There is no cluster resource type in this context; the correct resource is kommandercluster.
* B. nkp delete kommandercluster: The nkp CLI does not have a delete kommandercluster subcommand.
* D. nkp delete cluster in the attached cluster: This command is for deleting NKP-managed clusters, not detaching external clusters, and it should be run from the management cluster context.
:
Nutanix Kubernetes Platform Administration (NKPA) Course, Section on Fleet Management.

Nutanix Cloud Native (NCP-CN) 6.10 Study Guide, Chapter on Detaching Clusters.
Nutanix Cloud Bible, NutanixKubernetesPlatform Section: https://www.nutanixbible.com

NEW QUESTION: 50
A Kubernetes administrator has been tasked with deploying a new cluster to AWS. The administrator has received the following requirements for this deployment:
* Region us-east-1
* AMI rhel8.6What is a requirement for deploying a new cluster in AWS?
A. Use --dry-run parameter
B. Use --ami-format parameter
C. Set an export AWS_REGION
D. Set an export KUBECONFIG
Answer: C (LEAVE A REPLY)
For deploying NKP clusters in AWS, setting the AWS_REGION environment variable is a key prerequisite to ensure that the AWS CLI and underlying deployment scripts know which region to target. This is essential for provisioning instances using the specified AMI.
Exact extract:
"Set the AWS_REGION environment variable to the appropriate region prior to deploying clusters to ensure proper interaction with the AWS API." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "AWS Environment Configuration"
NCP-CN 6.10 Study Guide - "Preparing the AWS Environment for NKP"

NEW QUESTION: 51
There is a private registry for the NKP deployment and the company has an NKP Ultimate license. A Platform Engineer is using the Podman tool and is already logged in. Now, the engineer needs to send the private registry with the NKP Catalog Applications.
What command should the engineer use?
A. podman load -i ./container-images/nkp/catalog-applications-image-bundle-v2.12.0.tar
B. nkp push bundle --bundle ./container-images/nkp/catalog-applications-image-bundle-v2.12.0.tar --to- registry=${REGISTRY_URL} --to-registry-username=${REGISTRY_USERNAME} --to-registry- password=${REGISTRY_PASSWORD}
C. docker load -i ./container-images/nkp/catalog-applications-image-bundle-v2.12.0.tar
D. nkp apply bundle -f ./container-images/nkp/catalog-applications-image-bundle-v2.12.0.tar --to-registry=${REGISTRY_URL} --to-registry-username=${REGISTRY_USERNAME} --to-registry-password=${REGISTRY_PASSWORD}
Answer: B (LEAVE A REPLY)
To push the NKP Catalog Applications image bundle to a private registry, the official nkp push bundle command must be used with the specified parameters to authenticate and push the bundle to the registry.
Exact extract:

"Use the nkp push bundle command to upload the NKP catalog applications image bundle to the specified private registry, ensuring secure and complete image upload." Reference:
Nutanix Kubernetes Platform Administration (NKPA) 6.10 - "Pushing Catalog Applications to Private Registries" NCP-CN 6.10 Study Guide - "Private Registry Integration for NKP"