

RedHat.EX432.v2026-08-12.q19

Exam Code:	EX432
Exam Name:	Red Hat Certified Specialist in OpenShift Advanced Cluster Management
Certification Provider:	RedHat
Free Question Number:	19
Version:	v2026-08-12
# of views:	112
# of Questions views:	205
https://www.freecram.net/torrent/RedHat.EX432.v2026-08-12.q19.html	

NEW QUESTION: 1

Label managed clusters for placement (environment=dev, region=us-east)

Answer:

See the solution below in Explanation.

Explanation:

* List clusters:

```
oc get managedcluster
```

* Label the target cluster(s):

```
oc label managedcluster cluster-dev environment=dev region=us-east --overwrite
```

* Verify labels:

```
oc get managedcluster cluster-dev --show-labels
```

Why this matters:

Labels drive Placement decisions used by policies and applications.

NEW QUESTION: 2

Create a namespace-scoped application deployment model using ClusterSetBinding + Placement

Answer:

See the solution below in Explanation.

* Bind the ClusterSet into the application namespace using ManagedClusterSetBinding.

* Create Placement in that namespace selecting the desired clusters.

* Deploy application resources tied to that placement (Subscription or GitOps pattern). Why this matters: This is the standard multi-tenant ACM pattern for teams.

NEW QUESTION: 3

Create a policy that enforces a specific Operator to be installed on all dev clusters

Answer:

See the solution below in Explanation.

Explanation:

- * Use a policy template approach (OperatorGroup + Subscription + desired namespace) in a ConfigurationPolicy.
- * Bind it to a Placement targeting dev clusters.
- * Verify operator subscription exists on managed clusters and CSV reaches Succeeded.

Why this matters:

Automating consistent operator installation at scale is a realistic governance task; community policy collections exist demonstrating this model.

NEW QUESTION: 4

Create a Placement that selects clusters by OpenShift version label and exclude a region

Answer:

See the solution below in Explanation.

Explanation:

- * Ensure clusters have labels like openshiftVersion=4.18 and region=us-west etc.
- * Create placement with matchExpressions (include one label, exclude another):

matchExpressions:

- key: openshiftVersion

operator: In

values: ["4.18"]

- key: region

operator: NotIn

values: ["us-west"]

- * Validate PlacementDecision.

NEW QUESTION: 5

Create a PolicySet and include multiple policies for a baseline

Answer:

See the solution below in Explanation.

Explanation:

- * Create PolicySet referencing existing policies:

```
cat < < 'EOF' | oc apply -f -
```

```
apiVersion: policy.open-cluster-management.io/v1beta1
```

```
kind: PolicySet
```

```
metadata:
```

```
name: baseline-dev
```

```
namespace: team-dev
```

```
spec:
```

```
policies:
```

- policy-ensure-audit-namespace

Add more policies here as you create them

EOF

* Verify:

oc get policyset -n team-dev

Why this matters:

PolicySets group policies to apply/track as a unit-common governance practice.

NEW QUESTION: 6

Troubleshoot a managed cluster stuck "NotReady" by checking klusterlet/agent components

Answer:

See the solution below in Explanation.

Explanation:

* Check cluster conditions:

oc describe managedcluster cluster-dev

* Check agent namespaces and pods on the managed cluster (common namespace names depend on deployment, but you're looking for ACM/klusterlet agents).

* On hub, check managedclusteraddons and addon health:

oc get managedclusteraddon -n cluster-dev

oc describe managedclusteraddon -n cluster-dev < addon-name >

* Typical fixes: missing pull secret, network/DNS issues, CSR approval issues, etc. (Exam expects you to identify from events/conditions).

NEW QUESTION: 7

Use clusteradm to create a token and join a managed cluster (CLI-based import pattern)

Answer:

See the solution below in Explanation.

Explanation:

* On hub, generate join command/token (varies by environment):

clusteradm get token

* On managed cluster, run the join/import command.

* Validate managed cluster appears and is Ready. Why this matters: CLI-based import is a practical alternative to console import and fits real exam workflows.

NEW QUESTION: 8

Validate ClusterSet RBAC: confirm User A can only administer production ClusterSet

Answer:

See the solution below in Explanation.

Explanation:

* Grant the ClusterSet role (admin on production).

* Log in as user-a and attempt:

- * list clusters in production vs development
 - * create policies in a namespace bound to production
 - * Confirm authorization errors when accessing development resources.
- (ClusterSet RBAC is explicitly part of cluster set management and access scope.)

NEW QUESTION: 9

Configure a maintenance window for deployments (time window concept) and validate drift

Answer:

See the solution below in Explanation.

- * Edit the subscription/app definition to include a time window (active or blocked window).
- * Attempt a change outside the window and observe behavior (no rollout until window).
- * Confirm rollout during window.

NEW QUESTION: 10

Detach (unimport) a managed cluster safely from the hub

Answer:

See the solution below in Explanation.

Explanation:

- * Identify the managed cluster:

```
oc get managedcluster
```

- * If required in your environment, remove dependent placements/policies/apps targeting it (exam often expects you to check "why deletion stuck").

- * Delete managed cluster:

```
oc delete managedcluster cluster-dev
```

- * Watch finalizers and cleanup:

```
oc get managedcluster cluster-dev -o yaml | grep finalizers -n
```

Why this matters:

Proper lifecycle operations (import/manage/detach) is core EX432/EX480 scope.

NEW QUESTION: 11

Create a Placement that selects clusters in a ManagedClusterSet only

Answer:

See the solution below in Explanation.

Explanation:

- * Ensure namespace has the correct ManagedClusterSetBinding.
- * Create placement in that namespace; only clusters visible via bound ClusterSet can be selected.
- * Verify placementdecision includes only clusters from that set.

This ties together ClusterSets (grouping) and Placement (selection).

NEW QUESTION: 12

Set TimeWindow for App (Web Console)

Task information: Add timeWindow to Subscription spec (active window Friday 09:00-17:00)

Answer:

See the solution below in Explanation.

Explanation:

- * In ACM console go to Applications .
- * Select the application you deployed.
- * Open the Editor view (YAML editor) for the Subscription object.
- * Under spec, add a timeWindow section similar to what the lab requests:
- * windowtype: active
- * day: Friday
- * hours: 09:00-17:00
- * Save/apply the changes.
- * Verify the Subscription YAML now contains the time window and observe that deployment actions respect the scheduling window.

Why this matters:

- * Time windows let you control when changes roll out (useful for change-management, business hours, and maintenance windows).

NEW QUESTION: 13

Install RHACS secured cluster services on managed clusters using a governance policy approach

Answer:

See the solution below in Explanation.

Explanation:

- * Use a policy-based deployment model that installs RHACS components consistently across clusters (operator + secured cluster).
 - * Bind policy to placement selecting target clusters.
 - * Validate RHACS Sensor/Collector present in expected namespace on managed clusters.
- Policy collections exist demonstrating RHACS installation patterns via policy framework.

NEW QUESTION: 14

Enable and verify ACM "Search" and query for a resource across clusters

Answer:

See the solution below in Explanation.

Explanation:

- * Ensure ACM hub components are installed and healthy.
- * In ACM console, open Search and query for:
- * a namespace name
- * a specific deployment
- * Validate results show cluster origin and resource details.

Why this matters:

Search is a core ACM operational workflow for multi-cluster visibility.

NEW QUESTION: 15

Deploy GitOps Application (Subscription YAML)

Answer:

See the solution below in Explanation.

Explanation:

- * Apply the subscription/application manifest:
- * `oc apply -f application-subscription.yaml`
- * Confirm resources were created:
- * `oc get applications.app.k8s.io -A`
- * `oc get subscriptions.apps.open-cluster-management.io -A`
- * `oc describe application.app.k8s.io my-app -n < namespace >`
- * Watch placement/propagation (varies by model used in the lab):
- * `oc get placement -A`
- * `oc get placementrule -A`

Why this matters:

- * ACM "application" deployment commonly uses Subscription/Placement (legacy model) or newer Placement resources, enabling multi-cluster rollout from Git sources.

NEW QUESTION: 16

Create a Placement for production clusters and limit placement to 2 clusters

Answer:

See the solution below in Explanation.

Explanation:

- * Label clusters for prod:
`oc label managedcluster cluster-prod1 environment=prod --overwrite`
`oc label managedcluster cluster-prod2 environment=prod --overwrite`
- * Create Placement with numberOfClusters: 2:
`cat < < 'EOF' | oc apply -f -`
`apiVersion: cluster.open-cluster-management.io/v1beta1`
`kind: Placement`
`metadata:`
`name: prod-two-clusters`
`namespace: team-prod`
`spec:`
`numberOfClusters: 2`
`predicates:`
`- requiredClusterSelector:`
`labelSelector:`
`matchLabels:`

environment: prod

EOF

* Validate placement decisions:

oc get placementdecision -n team-prod

Valid EX432 Dumps shared by EduDump.com for Helping Passing EX432 Exam!

EduDump.com now offer the **newest EX432 exam dumps**, the EduDump.com EX432 exam **questions have been updated** and **answers have been corrected** get the **newest** EduDump.com EX432 dumps with Test Engine here:

<https://www.edudump.com/exams/RedHat/EX432/premium/> (47 Q&As Dumps, **35%OFF**

Special Discount Code: freeexam)

NEW QUESTION: 17

Import Cluster (Web Console)

Answer:

See the solution below in Explanation.

Explanation:

- * In the hub cluster Web Console, go to Infrastructure # Clusters (ACM console navigation).
- * Click Import cluster .
- * Provide a name (the UI may request details like distribution/credentials depending on flow).
- * The wizard will provide a command to run on the managed cluster you want to import.
- * Copy that import command.
- * Log into the managed cluster (spoke) using oc and run the copied command.
- * Back on the hub, wait until the cluster status becomes Ready / Managed .

Why these steps matter:

- * Import registers the managed cluster, installs the klusterlet/agent components, and enables policy/app placement management.

NEW QUESTION: 18

Create a Governance Policy to ensure a namespace exists on selected clusters

Answer:

See the solution below in Explanation.

Explanation:

Policies in ACM require:

- * a Policy resource
- * a Placement (which clusters to target)
- * a PlacementBinding (bind policy # placement)
- * Create the Policy (in team-dev) enforcing a namespace audit-logs:

cat < < 'EOF' | oc apply -f -

```
apiVersion: policy.open-cluster-management.io/v1
kind: Policy
metadata:
name: policy-ensure-audit-namespace
namespace: team-dev
spec:
remediationAction: enforce
disabled: false
policy-templates:
- objectDefinition:
apiVersion: policy.open-cluster-management.io/v1
kind: ConfigurationPolicy
metadata:
name: ensure-audit-namespace
spec:
remediationAction: enforce
severity: low
object-templates:
- complianceType: musthave
objectDefinition:
apiVersion: v1
kind: Namespace
metadata:
name: audit-logs
EOF
```

* Create PlacementBinding to bind it to dev-clusters placement:

```
cat < < 'EOF' | oc apply -f -
apiVersion: policy.open-cluster-management.io/v1
kind: PlacementBinding
metadata:
name: bind-policy-ensure-audit-namespace
namespace: team-dev
placementRef:
apiGroup: cluster.open-cluster-management.io
kind: Placement
name: dev-clusters
subjects:
- apiGroup: policy.open-cluster-management.io
kind: Policy
name: policy-ensure-audit-namespace
EOF
```

* Verify compliance:

```
oc get policy -n team-dev
```

```
oc describe policy policy-ensure-audit-namespace -n team-dev
```

Why this matters:

This is the core "ACM governance" exam pattern: define desired state and enforce across clusters.

NEW QUESTION: 19

Troubleshoot a policy that shows "Pending" (no placement decisions)

Answer:

See the solution below in Explanation.

Explanation:

* Confirm the policy exists in the namespace:

```
oc get policy -n team-dev
```

* Confirm Placement exists and selects clusters:

```
oc get placement -n team-dev
```

```
oc get placementdecision -n team-dev
```

* Confirm PlacementBinding points to correct placement + policy:

```
oc get placementbinding -n team-dev -o yaml
```

* Fix common mistakes:

* wrong namespace

* placement name mismatch

* missing ClusterSetBinding in the namespace Why this matters: Policy framework requires placement + binding; missing/incorrect targeting is the #1 real-world issue.

Valid EX432 Dumps shared by EduDump.com for Helping Passing EX432 Exam!

EduDump.com now offer the **newest EX432 exam dumps**, the EduDump.com EX432 exam

questions have been updated and **answers have been corrected** get the **newest**

EduDump.com EX432 dumps with Test Engine here:

<https://www.edudump.com/exams/RedHat/EX432/premium/> (47 Q&As Dumps, **35%OFF**)

Special Discount Code: freecram)