KUBERNETES (ENGLISH)

6. forduló

A kategória támogatója: Nokia

Ismertető a feladatlaphoz

> Kezdj neki minél hamarabb, mert a feladatot a forduló záró időpontjáig lehet beküldeni, nem addig lehet elkezdeni!

Sok sikert!

Role-based access control (RBAC) is a method of regulating access to computer or network resources based on the roles of individual users within your organization. RBAC authorization allowings to dynamically configure policies through the Kubernetes API.

Before you start, please read the following hint:

KDiff3 is a free and open-source diff and merge tool that can be helpful during the solution of coding-related exercises.
Download link: https://sourceforge.net/projects/kdiff3/files/

1. feladat 1 pont

You are able to use kubectl for managing a Kubernetes cluster. You have opened a shell inside a pod with kubectl exec command. Can you use kubectl from this shell?

Válasz

- Yes, by default kubectl works from any pod using the service account associated with the pod
- Yes, by default kubectl works from any pod using the same credentials as for kubectl exec
- No, Kubernetes API can not be accessed from running pods
- No, container images cannot contain kubectl
- Maybe, depends entirely on the pod

2. feladat 2 pont

Your company provides Chicken and Egg custom resources. You want to provide a ClusterRole that allows managing these resources, with the exception of deletion. Which of these would be sufficient?

Válasz

-
```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: chicken-egg-role
rules:
- apiGroups: ['example.com']
  resources: ['Chicken','Egg']
  verbs: [^'delete']
```

-
```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: chicken-egg-role
rules:
- apiGroups: ['example.com']
  resources: ['*']
  verbs: ['*']
```

-
```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: chicken-egg-role
```

```
rules:
- apiGroups: ['chickens.example.com', 'eggs.example.com']
  resources: ['*']
  verbs: ['create','read','update']
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: chicken-egg-role
rules:
- apiGroups: ['example.com']
  resources: ['chickens','eggs']
  verbs: ['create', 'get', 'list', 'patch', 'update', 'watch']
```

```
apiVersion: v1
kind: ClusterRole
metadata:
  name: chicken-egg-role
rules:
- apiGroups: ['rbac.authorization.k8s.io']
  resources: ['chickens','eggs']
  verbs: ['create', 'get', 'list', 'watch']
```

3. feladat 3 pont

You want to set up a service account to read Roles and ClusterRoles. The service account already exists, and the following definition is created:

```
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: default
  name: clusterrole-reader
rules:
- apiGroups: ["rbac.authorization.k8s.io"]
  resources: ["clusterroles"]
  verbs: ["get","read","list"]
---
apiVersion: rbac.authorization.k8s.io/v1
```

```yaml
kind: RoleBinding
metadata:
  name: clusterrole-reader-binding
  namespace: default
subjects:
- kind: ServiceAccount
  name: ithon-test
roleRef:
  kind: Role
  name: clusterrole-reader
  apiGroup: rbac.authorization.k8s.io
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: default
  name: role-reader
rules:
- apiGroups: ["rbac.authorization.k8s.io"]
  resources: ["roles"]
  verbs: ["get","read","list"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: role-reader-binding
  namespace: default
subjects:
- kind: ServiceAccount
  name: ithon-test
roleRef:
  kind: Role
  name: role-reader
  apiGroup: rbac.authorization.k8s.io
```

What outcome do you expect and why?

Válasz

- The definition can be applied, and service account "ithon-test" will be able to read all roles and clusterroles.
- The definition can be applied, and service account "ithon-test" will be able to read roles in the default namespace, and all clusterroles.
- The definition can be applied, and service account "ithon-test" will be able to read all roles in the default namespace, but not the clusterroles.
- The definition can not be applied as it contains an unrecognized verb.
- The definition will be ineffective as RBAC objects should be referenced in the builtin empty API group.

- The definition can be applied, and service account "ithon-test" will be able to read roles in the default namespace, and clusterroles in the default namespace.

Megoldások beküldése