



[Documentation](#) [Kubernetes Blog](#) [Training](#) [Partners](#) [Community](#) [Case Studies](#) [Versions ▾](#) [English ▾](#)

KubeCon + CloudNativeCon 2024

Join us for three days of incredible opportunities to collaborate, learn and share with the cloud native community.

[Buy your ticket now! 12 - 15 November | Salt Lake City.](#)



KubeCon



CloudNativeCon

North America 2024

[Kubernetes Documentation](#) / [Reference](#) / [Kubernetes API](#)
/ [Authorization Resources](#) / [ClusterRole](#)

ClusterRole

ClusterRole is a cluster level, logical grouping of PolicyRules that can be referenced as a unit by a RoleBinding or ClusterRoleBinding.

```
apiVersion: rbac.authorization.k8s.io/v1
```

```
import "k8s.io/api/rbac/v1"
```

ClusterRole

ClusterRole is a cluster level, logical grouping of PolicyRules that can be referenced as a unit by a RoleBinding or ClusterRoleBinding.

- **apiVersion:** rbac.authorization.k8s.io/v1
- **kind:** ClusterRole
- **metadata** ([ObjectMeta](#))

🔍 Search this site

- ▶ Documentation
- ▶ Getting started
- ▶ Concepts
- ▶ Tasks



Overview

Rules are controller managed and direct changes to Rules will be stomped by the controller.

▶ API

Access
Control

AggregationRule describes how to locate ClusterRoles to aggregate into the ClusterRole

▶ Well-

Known
Labels,
Annotatic
and
Taints

- **aggregationRule.clusterRoleSelectors** ([]LabelSelector)

Atomic: will be replaced during a merge

ClusterRoleSelectors holds a list of selectors which will be used to find ClusterRoles and create the rules. If any of the selectors match, then the ClusterRole's permissions will be added

▾ Kubernetes
API

▶ Workl
Resou

- **rules** ([]PolicyRule)

Atomic: will be replaced during a merge

▶ Servic
Resou

Rules holds all the PolicyRules for this ClusterRole

▶ Config
and

PolicyRule holds information that describes a policy rule, but does not contain information about who the rule applies to or which namespace the rule applies to.

- **rules.apiGroups** ([]string)

Atomic: will be replaced during a merge

APIGroups is the name of the APIGroup that contains the resources. If multiple API groups are specified, any action requested against one of the enumerated resources in any API group will be allowed. "" represents the core API group and "*" represents all API groups.

- **rules.resources** ([]string)



Atomic: will be replaced during a merge

Verbs is a list of Verbs that apply to ALL the ResourceKinds contained in this rule. '*' represents all verbs.

- **rules.resourceNames** ([]string)

Atomic: will be replaced during a merge

ResourceNames is an optional white list of names that the rule applies to. An empty set means that everything is allowed.

- **rules.nonResourceURLs** ([]string)

Atomic: will be replaced during a merge

NonResourceURLs is a set of partial urls that a user should have access to. *s are allowed, but only as the full, final step in the path Since non-resource URLs are not namespaced, this field is only applicable for ClusterRoles referenced from a ClusterRoleBinding. Rules can either apply to API resources (such as "pods" or "secrets") or non-resource URL paths (such as "/api"), but not both.

ClusterRoleList

ClusterRoleList is a collection of ClusterRoles

- **apiVersion:** rbac.authorization.k8s.io/v1
- **kind:** ClusterRoleList
- **metadata** ([ListMeta](#))



Operations

get read the specified ClusterRole

HTTP Request

GET /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}

Parameters

- **name** (*in path*): string, required

name of the ClusterRole

- **pretty** (*in query*): string

[pretty](#)

Response

200 ([ClusterRole](#)): OK

401: Unauthorized

list list or watch objects of kind ClusterRole

HTTP Request

GET /apis/rbac.authorization.k8s.io/v1/clusterroles



- **continue** (*in query*): string
[continue](#)
- **fieldSelector** (*in query*): string
[fieldSelector](#)
- **labelSelector** (*in query*): string
[labelSelector](#)
- **limit** (*in query*): integer
[limit](#)
- **pretty** (*in query*): string
[pretty](#)
- **resourceVersion** (*in query*): string
[resourceVersion](#)
- **resourceVersionMatch** (*in query*): string
[resourceVersionMatch](#)
- **sendInitialEvents** (*in query*): boolean
[sendInitialEvents](#)
- **timeoutSeconds** (*in query*): integer
[timeoutSeconds](#)
- **watch** (*in query*): boolean
[watch](#)



101. Unauthorized

create create a ClusterRole

HTTP Request

POST /apis/rbac.authorization.k8s.io/v1/clusterroles

Parameters

- **body**: [ClusterRole](#), required
- **dryRun** (*in query*): string
[dryRun](#)
- **fieldManager** (*in query*): string
[fieldManager](#)
- **fieldValidation** (*in query*): string
[fieldValidation](#)
- **pretty** (*in query*): string
[pretty](#)

Response

- 200 ([ClusterRole](#)): OK
- 201 ([ClusterRole](#)): Created
- 202 ([ClusterRole](#)): Accepted



HTTP Request

PUT /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}

Parameters

- **name** (*in path*): string, required

name of the ClusterRole

- **body**: [ClusterRole](#), required

- **dryRun** (*in query*): string

[dryRun](#)

- **fieldManager** (*in query*): string

[fieldManager](#)

- **fieldValidation** (*in query*): string

[fieldValidation](#)

- **pretty** (*in query*): string

[pretty](#)

Response

200 ([ClusterRole](#)): OK

201 ([ClusterRole](#)): Created

401: Unauthorized



PATCH /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}

Parameters

- **name** (*in path*): string, required

name of the ClusterRole

- **body**: [Patch](#), required

- **dryRun** (*in query*): string

[dryRun](#)

- **fieldManager** (*in query*): string

[fieldManager](#)

- **fieldValidation** (*in query*): string

[fieldValidation](#)

- **force** (*in query*): boolean

[force](#)

- **pretty** (*in query*): string

[pretty](#)

Response

200 ([ClusterRole](#)): OK

201 ([ClusterRole](#)): Created

401: Unauthorized



DELETE /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}

Parameters

- **name** (*in path*): string, required
name of the ClusterRole
- **body**: [DeleteOptions](#)
- **dryRun** (*in query*): string
[dryRun](#)
- **gracePeriodSeconds** (*in query*): integer
[gracePeriodSeconds](#)
- **pretty** (*in query*): string
[pretty](#)
- **propagationPolicy** (*in query*): string
[propagationPolicy](#)

Response

200 ([Status](#)): OK

202 ([Status](#)): Accepted

401: Unauthorized



DELETE /apis/rbac.authorization.k8s.io/v1/clusterroles

Parameters

- **body**: [DeleteOptions](#)
- **continue** (*in query*): string
[continue](#)
- **dryRun** (*in query*): string
[dryRun](#)
- **fieldSelector** (*in query*): string
[fieldSelector](#)
- **gracePeriodSeconds** (*in query*): integer
[gracePeriodSeconds](#)
- **labelSelector** (*in query*): string
[labelSelector](#)
- **limit** (*in query*): integer
[limit](#)
- **pretty** (*in query*): string
[pretty](#)
- **propagationPolicy** (*in query*): string
[propagationPolicy](#)



[resourceVersionMatch](#)

- **sendInitialEvents** (*in query*): boolean

[sendInitialEvents](#)

- **timeoutSeconds** (*in query*): integer

[timeoutSeconds](#)

Response

200 ([Status](#)): OK

401: Unauthorized

This page is automatically generated.

If you plan to report an issue with this page, mention that the page is auto-generated in your issue description. The fix may need to happen elsewhere in the Kubernetes project.

Feedback

Was this page helpful?

Yes

No



[Documentation](#) [Kubernetes Blog](#) [Training](#) [Partners](#) [Community](#) [Case Studies](#) [Versions ▾](#) [English ▾](#)



© 2024 The Kubernetes Authors |
Documentation Distributed under CC BY 4.0



© 2024 The Linux Foundation ®. All rights reserved. The Linux Foundation has registered trademarks and uses trademarks. For a list of trademarks of The Linux Foundation, please see our [Trademark Usage](#) page

ICP license: 京ICP备17074266号-3