OpenShift Container Platform 4.18

Role APIs

Reference guide for role APIs

Last Updated: 2025-05-15

OpenShift Container Platform 4.18 Role APIs

Reference guide for role APIs

Legal Notice

Abstract

This document describes the OpenShift Container Platform role API objects and their detailed specifications.

Table of Contents

CHAPTER 1. ROLE APIS

1.1. CLUSTERROLEBINDING [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

ClusterRoleBinding references a ClusterRole, but not contain it. It can reference any ClusterRole in the same namespace or in the global namespace. It adds who information via (Users and Groups) OR Subjects and namespace information by which namespace it exists in. ClusterRoleBindings in a given namespace only have effect in that namespace (excepting the master namespace which has power in all namespaces).
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

1.2. CLUSTERROLE [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

ClusterRole is a logical grouping of PolicyRules that can be referenced as a unit by ClusterRoleBindings.
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

1.3. ROLEBINDINGRESTRICTION [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

RoleBindingRestriction is an object that can be matched against a subject (user, group, or service account) to determine whether rolebindings on that subject are allowed in the namespace to which the RoleBindingRestriction belongs. If any one of those RoleBindingRestriction objects matches a subject, rolebindings on that subject in the namespace are allowed.
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

1.4. ROLEBINDING [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

RoleBinding references a Role, but not contain it. It can reference any Role in the same namespace or in the global namespace. It adds who information via (Users and Groups) OR Subjects and namespace information by which namespace it exists in. RoleBindings in a given namespace only have effect in that namespace (excepting the master namespace which has power in all namespaces).
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

1.5. ROLE [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

Role is a logical grouping of PolicyRules that can be referenced as a unit by RoleBindings.
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

CHAPTER 2. CLUSTERROLEBINDING [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

ClusterRoleBinding references a ClusterRole, but not contain it. It can reference any ClusterRole in the same namespace or in the global namespace. It adds who information via (Users and Groups) OR Subjects and namespace information by which namespace it exists in. ClusterRoleBindings in a given namespace only have effect in that namespace (excepting the master namespace which has power in all namespaces).
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

Required

- **subjects**
- **roleRef**

2.1. SPECIFICATION

Property	Type	Description
apiVersion	**string**	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
groupNames	**array (string)**	GroupNames holds all the groups directly bound to the role. This field should only be specified when supporting legacy clients and servers. See Subjects for further details.

Property	Type	Description
kind	**string**	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds
metadata	**ObjectMeta_v2**	metadata is the standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
roleRef	**ObjectReference**	RoleRef can only reference the current namespace and the global namespace. If the ClusterRoleRef cannot be resolved, the Authorizer must return an error. Since Policy is a singleton, this is sufficient knowledge to locate a role.
subjects	**array (ObjectReference)**	Subjects hold object references to authorize with this rule. This field is ignored if UserNames or GroupNames are specified to support legacy clients and servers. Thus newer clients that do not need to support backwards compatibility should send only fully qualified Subjects and should omit the UserNames and GroupNames fields. Clients that need to support backwards compatibility can use this field to build the UserNames and GroupNames.

Property	Type	Description
userNames	**array (string)**	UserNames holds all the usernames directly bound to the role. This field should only be specified when supporting legacy clients and servers. See Subjects for further details.

2.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/authorization.openshift.io/v1/clusterrolebindings**
  - **GET**: list objects of kind ClusterRoleBinding
  - **POST**: create a ClusterRoleBinding
- **/apis/authorization.openshift.io/v1/clusterrolebindings/{name}**
  - **DELETE**: delete a ClusterRoleBinding
  - **GET**: read the specified ClusterRoleBinding
  - **PATCH**: partially update the specified ClusterRoleBinding
  - **PUT**: replace the specified ClusterRoleBinding

2.2.1. /apis/authorization.openshift.io/v1/clusterrolebindings

HTTP method

GET

Description

list objects of kind ClusterRoleBinding

Table 2.1. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRoleBindingList** schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a ClusterRoleBinding

Table 2.2. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 2.3. Body parameters

Parameter	Type	Description
body	**ClusterRoleBinding** schema	

Table 2.4. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRoleBinding** schema
201 - Created	**ClusterRoleBinding** schema
202 - Accepted	**ClusterRoleBinding** schema
401 - Unauthorized	Empty

2.2.2. /apis/authorization.openshift.io/v1/clusterrolebindings/{name}

Table 2.5. Global path parameters

Parameter	Type	Description
name	**string**	name of the ClusterRoleBinding

HTTP method

DELETE

Description

delete a ClusterRoleBinding

Table 2.6. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Table 2.7. HTTP responses

HTTP code	Reponse body
200 - OK	**Status_v3** schema
202 - Accepted	**Status_v3** schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified ClusterRoleBinding

Table 2.8. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRoleBinding** schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified ClusterRoleBinding

Table 2.9. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 2.10. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRoleBinding** schema
201 - Created	**ClusterRoleBinding** schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified ClusterRoleBinding

Table 2.11. Query parameters

Parameter	Type	Description

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 2.12. Body parameters

Parameter	Type	Description
body	**ClusterRoleBinding** schema	

Table 2.13. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRoleBinding** schema
201 - Created	**ClusterRoleBinding** schema
401 - Unauthorized	Empty

CHAPTER 3. CLUSTERROLE [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

ClusterRole is a logical grouping of PolicyRules that can be referenced as a unit by ClusterRoleBindings.
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

Required

- **rules**

3.1. SPECIFICATION

Property	Type	Description
aggregationRule	AggregationRule_v2	AggregationRule is an optional field that describes how to build the Rules for this ClusterRole. If AggregationRule is set, then the Rules are controller managed and direct changes to Rules will be stomped by the controller.
apiVersion	**string**	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	**string**	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds

Property	Type	Description
metadata	**ObjectMeta_v2**	metadata is the standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
rules	**array**	Rules holds all the PolicyRules for this ClusterRole
rules[]	**object**	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.

3.1.1. .rules

Description

Rules holds all the PolicyRules for this ClusterRole

Type

array

3.1.2. .rules[]

Description

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.

Type

object

Required

- **verbs**
- **resources**

Property	Type	Description

Property	Type	Description
apiGroups	**array (string)**	APIGroups is the name of the APIGroup that contains the resources. If this field is empty, then both kubernetes and origin API groups are assumed. That means that if an action is requested against one of the enumerated resources in either the kubernetes or the origin API group, the request will be allowed
attributeRestrictions	**RawExtension**	AttributeRestrictions will vary depending on what the Authorizer/AuthorizationAttributeBuilder pair supports. If the Authorizer does not recognize how to handle the AttributeRestrictions, the Authorizer should report an error.
nonResourceURLs	**array (string)**	NonResourceURLsSlice 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 This name is intentionally different than the internal type so that the DefaultConvert works nicely and because the ordering may be different.
resourceNames	**array (string)**	ResourceNames is an optional white list of names that the rule applies to. An empty set means that everything is allowed.
resources	**array (string)**	Resources is a list of resources this rule applies to. ResourceAll represents all resources.
verbs	**array (string)**	Verbs is a list of Verbs that apply to ALL the ResourceKinds and AttributeRestrictions contained in this rule. VerbAll represents all kinds.

3.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/authorization.openshift.io/v1/clusterroles**
  - **GET**: list objects of kind ClusterRole
  - **POST**: create a ClusterRole
- **/apis/authorization.openshift.io/v1/clusterroles/{name}**
  - **DELETE**: delete a ClusterRole
  - **GET**: read the specified ClusterRole
  - **PATCH**: partially update the specified ClusterRole
  - **PUT**: replace the specified ClusterRole

3.2.1. /apis/authorization.openshift.io/v1/clusterroles

HTTP method

GET

Description

list objects of kind ClusterRole

Table 3.1. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRoleList** schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a ClusterRole

Table 3.2. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Parameter	Type	Description
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 3.3. Body parameters

Parameter	Type	Description
body	**ClusterRole** schema	

Table 3.4. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRole** schema
201 - Created	**ClusterRole** schema
202 - Accepted	**ClusterRole** schema
401 - Unauthorized	Empty

3.2.2. /apis/authorization.openshift.io/v1/clusterroles/{name}

Table 3.5. Global path parameters

Parameter	Type	Description
name	**string**	name of the ClusterRole

HTTP method

DELETE

Description

delete a ClusterRole

Table 3.6. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Table 3.7. HTTP responses

HTTP code	Reponse body
200 - OK	**Status_v3** schema
202 - Accepted	**Status_v3** schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified ClusterRole

Table 3.8. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRole** schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified ClusterRole

Table 3.9. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 3.10. HTTP responses

HTTP code	Reponse body
200 - OK	**ClusterRole** schema
201 - Created	**ClusterRole** schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified ClusterRole

Table 3.11. Query parameters

Parameter	Type	Description

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 3.12. Body parameters

Parameter	Type	Description
body	ClusterRole schema	

Table 3.13. HTTP responses

HTTP code	Reponse body
200 - OK	ClusterRole schema
201 - Created	ClusterRole schema
401 - Unauthorized	Empty

CHAPTER 4. ROLEBINDINGRESTRICTION [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

RoleBindingRestriction is an object that can be matched against a subject (user, group, or service account) to determine whether rolebindings on that subject are allowed in the namespace to which the RoleBindingRestriction belongs. If any one of those RoleBindingRestriction objects matches a subject, rolebindings on that subject in the namespace are allowed.
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

4.1. SPECIFICATION

Property	Type	Description
apiVersion	**string**	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	**string**	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds
metadata	**ObjectMeta**	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
spec	**object**	Spec defines the matcher.

4.1.1. .spec

Description

Spec defines the matcher.

Type

object

Property	Type	Description
grouprestriction	``	GroupRestriction matches against group subjects.
serviceaccountrestriction	``	ServiceAccountRestriction matches against service-account subjects.
userrestriction	``	UserRestriction matches against user subjects.

4.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/authorization.openshift.io/v1/rolebindingrestrictions**
  - **GET**: list objects of kind RoleBindingRestriction
- **/apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindingrestrictions**
  - **DELETE**: delete collection of RoleBindingRestriction
  - **GET**: list objects of kind RoleBindingRestriction
  - **POST**: create a RoleBindingRestriction
- **/apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindingrestrictions/{name}**
  - **DELETE**: delete a RoleBindingRestriction
  - **GET**: read the specified RoleBindingRestriction
  - **PATCH**: partially update the specified RoleBindingRestriction
  - **PUT**: replace the specified RoleBindingRestriction

4.2.1. /apis/authorization.openshift.io/v1/rolebindingrestrictions

HTTP method

GET

Description

list objects of kind RoleBindingRestriction

Table 4.1. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingRestrictionList** schema
401 - Unauthorized	Empty

4.2.2. /apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindingrestr

HTTP method

DELETE

Description

delete collection of RoleBindingRestriction

Table 4.2. HTTP responses

HTTP code	Reponse body
200 - OK	**Status** schema
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind RoleBindingRestriction

Table 4.3. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingRestrictionList** schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a RoleBindingRestriction

Table 4.4. Query parameters

Parameter	Type	Description

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 4.5. Body parameters

Parameter	Type	Description
body	**RoleBindingRestriction** schema	

Table 4.6. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingRestriction** schema
201 - Created	**RoleBindingRestriction** schema
202 - Accepted	**RoleBindingRestriction** schema
401 - Unauthorized	Empty

4.2.3. /apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindingrestr

Table 4.7. Global path parameters

Parameter	Type	Description
name	**string**	name of the RoleBindingRestriction

HTTP method

DELETE

Description

delete a RoleBindingRestriction

Table 4.8. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Table 4.9. HTTP responses

HTTP code	Reponse body
200 - OK	**Status** schema
202 - Accepted	**Status** schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified RoleBindingRestriction

Table 4.10. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingRestriction** schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified RoleBindingRestriction

Table 4.11. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 4.12. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingRestriction** schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified RoleBindingRestriction

Table 4.13. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 4.14. Body parameters

Parameter	Type	Description
body	**RoleBindingRestriction** schema	

Table 4.15. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingRestriction** schema
201 - Created	**RoleBindingRestriction** schema
401 - Unauthorized	Empty

CHAPTER 5. ROLEBINDING [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

RoleBinding references a Role, but not contain it. It can reference any Role in the same namespace or in the global namespace. It adds who information via (Users and Groups) OR Subjects and namespace information by which namespace it exists in. RoleBindings in a given namespace only have effect in that namespace (excepting the master namespace which has power in all namespaces).
Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

Required

- **subjects**
- **roleRef**

5.1. SPECIFICATION

Property	Type	Description
apiVersion	**string**	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
groupNames	**array (string)**	GroupNames holds all the groups directly bound to the role. This field should only be specified when supporting legacy clients and servers. See Subjects for further details.

Property	Type	Description
kind	**string**	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds
metadata	**ObjectMeta_v2**	metadata is the standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
roleRef	**ObjectReference**	RoleRef can only reference the current namespace and the global namespace. If the RoleRef cannot be resolved, the Authorizer must return an error. Since Policy is a singleton, this is sufficient knowledge to locate a role.
subjects	**array (ObjectReference)**	Subjects hold object references to authorize with this rule. This field is ignored if UserNames or GroupNames are specified to support legacy clients and servers. Thus newer clients that do not need to support backwards compatibility should send only fully qualified Subjects and should omit the UserNames and GroupNames fields. Clients that need to support backwards compatibility can use this field to build the UserNames and GroupNames.
userNames	**array (string)**	UserNames holds all the usernames directly bound to the role. This field should only be specified when supporting legacy clients and servers. See Subjects for further details.

5.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/authorization.openshift.io/v1/rolebindings**
  - **GET**: list objects of kind RoleBinding
- **/apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindings**
  - **GET**: list objects of kind RoleBinding
  - **POST**: create a RoleBinding
- **/apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindings/{name}**
  - **DELETE**: delete a RoleBinding
  - **GET**: read the specified RoleBinding
  - **PATCH**: partially update the specified RoleBinding
  - **PUT**: replace the specified RoleBinding

5.2.1. /apis/authorization.openshift.io/v1/rolebindings

HTTP method

GET

Description

list objects of kind RoleBinding

Table 5.1. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingList** schema
401 - Unauthorized	Empty

5.2.2. /apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindings

HTTP method

GET

Description

list objects of kind RoleBinding

Table 5.2. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBindingList** schema

HTTP code	Reponse body
401 - Unauthorized	Empty

HTTP method

POST

Description

create a RoleBinding

Table 5.3. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 5.4. Body parameters

Parameter	Type	Description
body	**RoleBinding** schema	

Table 5.5. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBinding** schema
201 - Created	**RoleBinding** schema
202 - Accepted	**RoleBinding** schema
401 - Unauthorized	Empty

5.2.3. /apis/authorization.openshift.io/v1/namespaces/{namespace}/rolebindings/{na

Table 5.6. Global path parameters

Parameter	Type	Description
name	**string**	name of the RoleBinding

HTTP method

DELETE

Description

delete a RoleBinding

Table 5.7. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Table 5.8. HTTP responses

HTTP code	Reponse body
200 - OK	**Status_v3** schema
202 - Accepted	**Status_v3** schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified RoleBinding

Table 5.9. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBinding** schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified RoleBinding

Table 5.10. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 5.11. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBinding** schema

HTTP code	Reponse body
201 - Created	**RoleBinding** schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified RoleBinding

Table 5.12. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 5.13. Body parameters

Parameter	Type	Description
body	**RoleBinding** schema	

Table 5.14. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleBinding** schema
201 - Created	**RoleBinding** schema
401 - Unauthorized	Empty

CHAPTER 6. ROLE [AUTHORIZATION.OPENSHIFT.IO/V1]

Description

Role is a logical grouping of PolicyRules that can be referenced as a unit by RoleBindings. Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

Type

object

Required

- **rules**

6.1. SPECIFICATION

Property	Type	Description
apiVersion	**string**	APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources
kind	**string**	Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds
metadata	**ObjectMeta_v2**	metadata is the standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata
rules	**array**	Rules holds all the PolicyRules for this Role

Property	Type	Description
rules[]	**object**	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.

6.1.1. .rules

Description

Rules holds all the PolicyRules for this Role

Type

array

6.1.2. .rules[]

Description

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.

Type

object

Required

- **verbs**
- **resources**

Property	Type	Description
apiGroups	**array (string)**	APIGroups is the name of the APIGroup that contains the resources. If this field is empty, then both kubernetes and origin API groups are assumed. That means that if an action is requested against one of the enumerated resources in either the kubernetes or the origin API group, the request will be allowed

Property	Type	Description
attributeRestrictions	RawExtension	AttributeRestrictions will vary depending on what the Authorizer/AuthorizationAttributeBuilder pair supports. If the Authorizer does not recognize how to handle the AttributeRestrictions, the Authorizer should report an error.
nonResourceURLs	**array (string)**	NonResourceURLsSlice 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 This name is intentionally different than the internal type so that the DefaultConvert works nicely and because the ordering may be different.
resourceNames	**array (string)**	ResourceNames is an optional white list of names that the rule applies to. An empty set means that everything is allowed.
resources	**array (string)**	Resources is a list of resources this rule applies to. ResourceAll represents all resources.
verbs	**array (string)**	Verbs is a list of Verbs that apply to ALL the ResourceKinds and AttributeRestrictions contained in this rule. VerbAll represents all kinds.

6.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/authorization.openshift.io/v1/roles**
  - **GET**: list objects of kind Role
- **/apis/authorization.openshift.io/v1/namespaces/{namespace}/roles**
  - **GET**: list objects of kind Role
  - **POST**: create a Role

- **/apis/authorization.openshift.io/v1/namespaces/{namespace}/roles/{name}**
  - **DELETE**: delete a Role
  - **GET**: read the specified Role
  - **PATCH**: partially update the specified Role
  - **PUT**: replace the specified Role

6.2.1. /apis/authorization.openshift.io/v1/roles

HTTP method

GET

Description

list objects of kind Role

Table 6.1. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleList** schema
401 - Unauthorized	Empty

6.2.2. /apis/authorization.openshift.io/v1/namespaces/{namespace}/roles

HTTP method

GET

Description

list objects of kind Role

Table 6.2. HTTP responses

HTTP code	Reponse body
200 - OK	**RoleList** schema
401 - Unauthorized	Empty

HTTP method

POST

Description

create a Role

Table 6.3. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 6.4. Body parameters

Parameter	Type	Description
body	**Role** schema	

Table 6.5. HTTP responses

HTTP code	Reponse body
200 - OK	**Role** schema
201 - Created	**Role** schema
202 - Accepted	**Role** schema
401 - Unauthorized	Empty

6.2.3. /apis/authorization.openshift.io/v1/namespaces/{namespace}/roles/{name}

Table 6.6. Global path parameters

Parameter	Type	Description
name	**string**	name of the Role

HTTP method

DELETE

Description

delete a Role

Table 6.7. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

Table 6.8. HTTP responses

HTTP code	Reponse body
200 - OK	**Status_v3** schema
202 - Accepted	**Status_v3** schema
401 - Unauthorized	Empty

HTTP method

GET

Description

read the specified Role

Table 6.9. HTTP responses

HTTP code	Reponse body
200 - OK	**Role** schema
401 - Unauthorized	Empty

HTTP method

PATCH

Description

partially update the specified Role

Table 6.10. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 6.11. HTTP responses

HTTP code	Reponse body
200 - OK	**Role** schema
201 - Created	**Role** schema
401 - Unauthorized	Empty

HTTP method

PUT

Description

replace the specified Role

Table 6.12. Query parameters

Parameter	Type	Description
dryRun	**string**	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed
fieldValidation	**string**	fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Table 6.13. Body parameters

Parameter	Type	Description
body	Role schema	

Table 6.14. HTTP responses

HTTP code	Reponse body
200 - OK	Role schema
201 - Created	Role schema
401 - Unauthorized	Empty