OpenShift Container Platform 4.18

Cluster APIs

Reference guide for cluster APIs

Last Updated: 2025-05-15

OpenShift Container Platform 4.18 Cluster APIs

Reference guide for cluster APIs

Legal Notice

Abstract

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

Table of Contents

CHAPTER 1. CLUSTER APIS

1.1. IPADDRESS [IPAM.CLUSTER.X-K8S.IO/V1BETA1]

Description

IPAddress is the Schema for the ipaddress API.

Type

object

1.2. IPADDRESSCLAIM [IPAM.CLUSTER.X-K8S.IO/V1BETA1]

Description

IPAddressClaim is the Schema for the ipaddressclaim API.

Type

object

CHAPTER 2. IPADDRESS [IPAM.CLUSTER.X-K8S.IO/V1BETA1]

Description

IPAddress is the Schema for the ipaddress API.

Type

object

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
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**	IPAddressSpec is the desired state of an IPAddress.

2.1.1. .spec

Description

IPAddressSpec is the desired state of an IPAddress.

Type

object

Required

- **address**
- **claimRef**
- **poolRef**
- **prefix**

Property	Type	Description
address	**string**	Address is the IP address.
claimRef	**object**	ClaimRef is a reference to the claim this IPAddress was created for.
gateway	**string**	Gateway is the network gateway of the network the address is from.
poolRef	**object**	PoolRef is a reference to the pool that this IPAddress was created from.
prefix	**integer**	Prefix is the prefix of the address.

2.1.2. .spec.claimRef

Description

ClaimRef is a reference to the claim this IPAddress was created for.

Type

object

Property	Type	Description

Property	Type	Description
name	**string**	Name of the referent. This field is effectively required, but due to backwards compatibility is allowed to be empty. Instances of this type with an empty value here are almost certainly wrong. TODO: Add other useful fields. apiVersion, kind, uid? More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Drop **kubebuilder:default** when controller-gen doesn't need it https://github.com/kubernetes-sigs/kubebuilder/issues/3896.

2.1.3. .spec.poolRef

Description

PoolRef is a reference to the pool that this IPAddress was created from.

Type

object

Required

- **kind**
- **name**

Property	Type	Description
apiGroup	**string**	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	**string**	Kind is the type of resource being referenced
name	**string**	Name is the name of resource being referenced

2.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/ipam.cluster.x-k8s.io/v1beta1/ipaddresses**
  - **GET**: list objects of kind IPAddress
- **/apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddresses**
  - **DELETE**: delete collection of IPAddress
  - **GET**: list objects of kind IPAddress
  - **POST**: create an IPAddress
- **/apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddresses/{name}**
  - **DELETE**: delete an IPAddress
  - **GET**: read the specified IPAddress
  - **PATCH**: partially update the specified IPAddress
  - **PUT**: replace the specified IPAddress

2.2.1. /apis/ipam.cluster.x-k8s.io/v1beta1/ipaddresses

HTTP method

GET

Description

list objects of kind IPAddress

Table 2.1. HTTP responses

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

2.2.2. /apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddresses

HTTP method

DELETE

Description

delete collection of IPAddress

Table 2.2. HTTP responses

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

HTTP code	Reponse body
401 - Unauthorized	Empty

HTTP method

GET

Description

list objects of kind IPAddress

Table 2.3. HTTP responses

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

HTTP method

POST

Description

create an IPAddress

Table 2.4. 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 2.5. Body parameters

Parameter	Type	Description
body	**IPAddress** schema	

Table 2.6. HTTP responses

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

2.2.3. /apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddresses/{name}

Table 2.7. Global path parameters

Parameter	Type	Description

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

HTTP method

DELETE

Description

delete an IPAddress

Table 2.8. 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 2.9. HTTP responses

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

HTTP method

GET

Description

read the specified IPAddress

Table 2.10. HTTP responses

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

HTTP method

PATCH

Description

partially update the specified IPAddress

Table 2.11. 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 2.12. HTTP responses

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

HTTP method

PUT

Description

replace the specified IPAddress

Table 2.13. 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 2.14. Body parameters

Parameter	Type	Description
body	**IPAddress** schema	

Table 2.15. HTTP responses

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

CHAPTER 3. IPADDRESSCLAIM [IPAM.CLUSTER.X-K8S.IO/V1BETA1]

Description

IPAddressClaim is the Schema for the ipaddressclaim API.

Type

object

3.1. 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**	IPAddressClaimSpec is the desired state of an IPAddressClaim.
status	**object**	IPAddressClaimStatus is the observed status of a IPAddressClaim.

3.1.1. .spec

Description

IPAddressClaimSpec is the desired state of an IPAddressClaim.

Type

object

Required

- **poolRef**

Property	Type	Description
clusterName	**string**	ClusterName is the name of the Cluster this object belongs to.
poolRef	**object**	PoolRef is a reference to the pool from which an IP address should be created.

3.1.2. .spec.poolRef

Description

PoolRef is a reference to the pool from which an IP address should be created.

Type

object

Required

- **kind**
- **name**

Property	Type	Description
apiGroup	**string**	APIGroup is the group for the resource being referenced. If APIGroup is not specified, the specified Kind must be in the core API group. For any other third-party types, APIGroup is required.
kind	**string**	Kind is the type of resource being referenced
name	**string**	Name is the name of resource being referenced

3.1.3. .status

Description

IPAddressClaimStatus is the observed status of a IPAddressClaim.

Type

object

Property	Type	Description
addressRef	**object**	AddressRef is a reference to the address that was created for this claim.
conditions	**array**	Conditions summarises the current state of the IPAddressClaim
conditions[]	**object**	Condition defines an observation of a Cluster API resource operational state.

3.1.4. .status.addressRef

Description

AddressRef is a reference to the address that was created for this claim.

Type

object

Property	Type	Description
name	**string**	Name of the referent. This field is effectively required, but due to backwards compatibility is allowed to be empty. Instances of this type with an empty value here are almost certainly wrong. TODO: Add other useful fields. apiVersion, kind, uid? More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Drop **kubebuilder:default** when controller-gen doesn't need it https://github.com/kubernetes-sigs/kubebuilder/issues/3896.

3.1.5. .status.conditions

Description

Conditions summarises the current state of the IPAddressClaim

Type

array

3.1.6. .status.conditions[]

Description

Condition defines an observation of a Cluster API resource operational state.

Type

object

Required

- **lastTransitionTime**
- **status**
- **type**

Property	Type	Description
lastTransitionTime	**string**	Last time the condition transitioned from one status to another. This should be when the underlying condition changed. If that is not known, then using the time when the API field changed is acceptable.
message	**string**	A human readable message indicating details about the transition. This field may be empty.
reason	**string**	The reason for the condition's last transition in CamelCase. The specific API may choose whether or not this field is considered a guaranteed API. This field may not be empty.
severity	**string**	Severity provides an explicit classification of Reason code, so the users or machines can immediately understand the current situation and act accordingly. The Severity field MUST be set only when Status=False.
status	**string**	Status of the condition, one of True, False, Unknown.

Property	Type	Description
type	**string**	Type of condition in CamelCase or in foo.example.com/CamelCase. Many .condition.type values are consistent across resources like Available, but because arbitrary conditions can be useful (see .node.status.conditions), the ability to deconflict is important.

3.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/ipam.cluster.x-k8s.io/v1beta1/ipaddressclaims**
  - **GET**: list objects of kind IPAddressClaim
- **/apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddressclaims**
  - **DELETE**: delete collection of IPAddressClaim
  - **GET**: list objects of kind IPAddressClaim
  - **POST**: create an IPAddressClaim
- **/apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddressclaims/{name}**
  - **DELETE**: delete an IPAddressClaim
  - **GET**: read the specified IPAddressClaim
  - **PATCH**: partially update the specified IPAddressClaim
  - **PUT**: replace the specified IPAddressClaim
- **/apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddressclaims/{name}/status**
  - **GET**: read status of the specified IPAddressClaim
  - **PATCH**: partially update status of the specified IPAddressClaim
  - **PUT**: replace status of the specified IPAddressClaim

3.2.1. /apis/ipam.cluster.x-k8s.io/v1beta1/ipaddressclaims

HTTP method

GET

Description

list objects of kind IPAddressClaim

Table 3.1. HTTP responses

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

3.2.2. /apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddressclaims

HTTP method

DELETE

Description

delete collection of IPAddressClaim

Table 3.2. HTTP responses

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

HTTP method

GET

Description

list objects of kind IPAddressClaim

Table 3.3. HTTP responses

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

HTTP method

POST

Description

create an IPAddressClaim

Table 3.4. 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 3.5. Body parameters

Parameter	Type	Description
body	**IPAddressClaim** schema	

Table 3.6. HTTP responses

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

3.2.3. /apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddressclaims/{name}

Table 3.7. Global path parameters

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

HTTP method

DELETE

Description

delete an IPAddressClaim

Table 3.8. 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 3.9. HTTP responses

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

HTTP method

GET

Description

read the specified IPAddressClaim

Table 3.10. HTTP responses

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

HTTP method

PATCH

Description

partially update the specified IPAddressClaim

Table 3.11. 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 3.12. HTTP responses

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

HTTP method

PUT

Description

replace the specified IPAddressClaim

Table 3.13. 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 3.14. Body parameters

Parameter	Type	Description
body	**IPAddressClaim** schema	

Table 3.15. HTTP responses

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

3.2.4. /apis/ipam.cluster.x-k8s.io/v1beta1/namespaces/{namespace}/ipaddressclaims/{name}/status

Table 3.16. Global path parameters

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

HTTP method

GET

Description

read status of the specified IPAddressClaim

Table 3.17. HTTP responses

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

HTTP method

PATCH

Description

partially update status of the specified IPAddressClaim

Table 3.18. 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 3.19. HTTP responses

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

HTTP method

PUT

Description

replace status of the specified IPAddressClaim

Table 3.20. 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 3.21. Body parameters

Parameter	Type	Description
body	**IPAddressClaim** schema	

Table 3.22. HTTP responses

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