Policy APIs

PodDisruptionBudget [policy/v1]

PodDisruptionBudget [policy/v1]

Description

PodDisruptionBudget is an object to define the max disruption that can be caused to a collection of pods

Type

`object`

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:

Property	Type	Description
		https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds ↗
`metadata`	`ObjectMeta`	ObjectMeta is metadata that all persisted resources must have, which includes all objects users must create.
`spec`	`object`	PodDisruptionBudgetSpec is a description of a PodDisruptionBudget.
`status`	`object`	PodDisruptionBudgetStatus represents information about the status of a PodDisruptionBudget. Status may trail the actual state of a system.

.spec

Description

PodDisruptionBudgetSpec is a description of a PodDisruptionBudget.

Type

`object`

Property	Type	Description
`maxUnavailable`	`integer|string`	IntOrString is a type that can hold an int32 or a string. When used in JSON or YAML marshalling and unmarshalling, it produces or consumes the inner type. This allows you to have, for example, a JSON

Property	Type	Description
		field that can accept a name or number.
`minAvailable`	`integer|string`	IntOrString is a type that can hold an int32 or a string. When used in JSON or YAML marshalling and unmarshalling, it produces or consumes the inner type. This allows you to have, for example, a JSON field that can accept a name or number.
`selector`	`object`	A label selector is a label query over a set of resources. The result of matchLabels and matchExpressions are ANDed. An empty label selector matches all objects. A null label selector matches no objects.
`unhealthyPodEvictionPolicy`	`string`	UnhealthyPodEvictionPolicy defines the criteria for when unhealthy pods should be considered for eviction. Current implementation considers healthy pods, as pods that have status.conditions item with type="Ready",status="True". Valid policies are IfHealthyBudget and AlwaysAllow. If no policy is specified, the default behavior will be used,

Property	Type	Description
		which corresponds to the IfHealthyBudget policy. IfHealthyBudget policy means that running pods (status.phase="Running"), but not yet healthy can be evicted only if the guarded application is not disrupted (status.currentHealthy is at least equal to status.desiredHealthy). Healthy pods will be subject to the PDB for eviction. AlwaysAllow policy means that all running pods (status.phase="Running"), but not yet healthy are considered disrupted and can be evicted regardless of whether the criteria in a PDB is met. This means perspective running pods of a disrupted application might not get a chance to become healthy. Healthy pods will be subject to the PDB for eviction. Additional policies may be added in the future. Clients making eviction decisions should disallow eviction of unhealthy pods if they encounter an unrecognized policy in this field.

.spec.selector

Description

A label selector is a label query over a set of resources. The result of matchLabels and matchExpressions are ANDed. An empty label selector matches all objects. A null label selector matches no objects.

Type

`object`

Property	Type	Description
`matchExpressions`	`array`	matchExpressions is a list of label selector requirements. The requirements are ANDed.
`matchLabels`	`object`	matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

.spec.selector.matchExpressions

Description

matchExpressions is a list of label selector requirements. The requirements are ANDed.

Type

`array`

.spec.selector.matchExpressions[]

Description

A label selector requirement is a selector that contains values, a key, and an operator that relates the key and values.

Type

`object`

Required

`key` `operator`

Property	Type	Description
`key`	`string`	key is the label key that the selector applies to.
`operator`	`string`	operator represents a key's relationship to a set of values. Valid operators are In, NotIn, Exists and DoesNotExist.
`values`	`array`	values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

.spec.selector.matchExpressions[].values

Description

values is an array of string values. If the operator is In or NotIn, the values array must be non-empty. If the operator is Exists or DoesNotExist, the values array must be empty. This array is replaced during a strategic merge patch.

Type

`array`

.spec.selector.matchExpressions[].values[]

Type

`string`

.spec.selector.matchLabels

Description

matchLabels is a map of {key,value} pairs. A single {key,value} in the matchLabels map is equivalent to an element of matchExpressions, whose key field is "key", the operator is "In", and the values array contains only "value". The requirements are ANDed.

Type

`object`

.status

Description

PodDisruptionBudgetStatus represents information about the status of a PodDisruptionBudget. Status may trail the actual state of a system.

Type

`object`

Property	Type	Description
`conditions`	`array`	Conditions contain conditions for PDB. The disruption controller sets the DisruptionAllowed condition. The following are known values for the reason field (additional reasons could be added in the future): - SyncFailed: The controller encountered an error and wasn't able to compute the number of allowed disruptions. Therefore no disruptions are allowed and the status of the condition will be False. • InsufficientPods: The number of pods are either at or below the number required by the PodDisruptionBudget. No disruptions are allowed and the status of the condition will be False. • SufficientPods: There are more pods than required by the PodDisruptionBudget. The

Property	Type	Description
		condition will be True, and the number of allowed disruptions are provided by the disruptionsAllowed property.
`currentHealthy`	`integer`	current number of healthy pods
`desiredHealthy`	`integer`	minimum desired number of healthy pods
`disruptedPods`	`object`	DisruptedPods contains information about pods whose eviction was processed by the API server eviction subresource handler but has not yet been observed by the PodDisruptionBudget controller. A pod will be in this map from the time when the API server processed the eviction request to the time when the pod is seen by PDB controller as having been marked for deletion (or after a timeout). The key in the map is the name of the pod and the value is the time when the API server processed the eviction request. If the deletion didn't occur and a pod is still there it will be removed from the list automatically by PodDisruptionBudget controller after some time. If everything goes smooth this map should be empty for the most of the time. Large number of entries in the map may indicate problems with pod deletions.
`disruptionsAllowed`	`integer`	Number of pod disruptions that are currently allowed.

Property	Type	Description
`expectedPods`	`integer`	total number of pods counted by this disruption budget
`observedGeneration`	`integer`	Most recent generation observed when updating this PDB status. DisruptionsAllowed and other status information is valid only if observedGeneration equals to PDB's object generation.

.status.conditions

Description

Conditions contain conditions for PDB. The disruption controller sets the DisruptionAllowed condition. The following are known values for the reason field (additional reasons could be added in the future): - SyncFailed: The controller encountered an error and wasn't able to compute the number of allowed disruptions. Therefore no disruptions are allowed and the status of the condition will be False. - InsufficientPods: The number of pods are either at or below the number required by the PodDisruptionBudget. No disruptions are allowed and the status of the condition will be False. - SufficientPods: There are more pods than required by the PodDisruptionBudget. The condition will be True, and the number of allowed disruptions are provided by the disruptionsAllowed property.

Type

`array`

.status.conditions[]

Description

Condition contains details for one aspect of the current state of this API Resource.

Type

`object`

Required

`type` `status` `lastTransitionTime` `reason` `message`

Property	Type	Description
`lastTransitionTime`	`string`	Time is a wrapper around time.Time which supports correct marshaling to YAML and JSON. Wrappers are provided for many of the factory methods that the time package offers.
`message`	`string`	message is a human readable message indicating details about the transition. This may be an empty string.
`observedGeneration`	`integer`	observedGeneration represents the .metadata.generation that the condition was set based upon. For instance, if .metadata.generation is currently 12, but the .status.conditions[x].observedGeneration is 9, the condition is out of date with respect to the current state of the instance.
`reason`	`string`	reason contains a programmatic identifier indicating the reason for the condition's last transition. Producers of specific condition types may define expected values and meanings for this field, and whether the values are considered a guaranteed API. The value should be a CamelCase string. This field may not be empty.

Property	Type	Description
`status`	`string`	status of the condition, one of True, False, Unknown.
`type`	`string`	type of condition in CamelCase or in foo.example.com/CamelCase.

.status.disruptedPods

Description

DisruptedPods contains information about pods whose eviction was processed by the API server eviction subresource handler but has not yet been observed by the PodDisruptionBudget controller. A pod will be in this map from the time when the API server processed the eviction request to the time when the pod is seen by PDB controller as having been marked for deletion (or after a timeout). The key in the map is the name of the pod and the value is the time when the API server processed the eviction request. If the deletion didn't occur and a pod is still there it will be removed from the list automatically by PodDisruptionBudget controller after some time. If everything goes smooth this map should be empty for the most of the time. Large number of entries in the map may indicate problems with pod deletions.

Type

`object`

API Endpoints

The following API endpoints are available:

- `/kubernetes/{cluster}/apis/policy/v1/namespaces/{namespace}/poddisruptionbudgets`
  - `DELETE`: delete collection of PodDisruptionBudget

- `GET` : list objects of kind PodDisruptionBudget
- `POST` : create a new PodDisruptionBudget

- `/kubernetes/{cluster}/apis/policy/v1/namespaces/{namespace}/poddisruptionbudgets/{name}`
  - `DELETE` : delete the specified PodDisruptionBudget
  - `GET` : read the specified PodDisruptionBudget
  - `PATCH` : partially update the specified PodDisruptionBudget
  - `PUT` : replace the specified PodDisruptionBudget
- `/kubernetes/{cluster}/apis/policy/v1/namespaces/{namespace}/poddisruptionbudgets/{name}/status`
  - `GET` : read status of the specified PodDisruptionBudget
  - `PATCH` : partially update status of the specified PodDisruptionBudget
  - `PUT` : replace status of the specified PodDisruptionBudget

/kubernetes/{cluster}/apis/policy/v1/namespaces/{namespace}/poddisruptionbudgets

HTTP method

`DELETE`

Description

delete collection of PodDisruptionBudget

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

list objects of kind PodDisruptionBudget

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudgetList` schema
401 - Unauthorized	Empty

HTTP method

`POST`

Description

create a new PodDisruptionBudget

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

Parameter	Type	Description
		fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.

Body parameters

Parameter	Type	Description
`body`	`PodDisruptionBudget` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudget` schema
201 - Created	`PodDisruptionBudget` schema
202 - Accepted	`PodDisruptionBudget` schema
401 - Unauthorized	Empty

/kubernetes/{cluster}/apis/policy/v1/namespaces/{namespace}/poddisruptionbudgets/{name}

HTTP method

`DELETE`

Description

delete the specified PodDisruptionBudget

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will

Parameter	Type	Description
		result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed

HTTP responses

HTTP code	Response body
200 - OK	`Status` schema
202 - Accepted	`Status` schema
401 - Unauthorized	Empty

HTTP method

`GET`

Description

read the specified PodDisruptionBudget

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudget` schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update the specified PodDisruptionBudget

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.

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudget` schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace the specified PodDisruptionBudget

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.

Body parameters

Parameter	Type	Description
`body`	`PodDisruptionBudget` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudget` schema

HTTP code	Response body
201 - Created	`PodDisruptionBudget` schema
401 - Unauthorized	Empty

/kubernetes/{cluster}/apis/policy/v1/namespaces/{namespace}/poddisruptionbudgets/{name}/status

HTTP method

`GET`

Description

read status of the specified PodDisruptionBudget

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudget` schema
401 - Unauthorized	Empty

HTTP method

`PATCH`

Description

partially update status of the specified PodDisruptionBudget

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.

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudget` schema
401 - Unauthorized	Empty

HTTP method

`PUT`

Description

replace status of the specified PodDisruptionBudget

Query parameters

Parameter	Type	Description
`dryRun`	`string`	When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun

Parameter	Type	Description
		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.

Body parameters

Parameter	Type	Description
`body`	`PodDisruptionBudget` schema	`application/json` formatted

HTTP responses

HTTP code	Response body
200 - OK	`PodDisruptionBudget` schema
201 - Created	`PodDisruptionBudget` schema
401 - Unauthorized	Empty