[Kubecronic] Руководство администратора

Оглавление

Для установки ПО необходимы:

- доступ уровня Администратор к работающим кластерам kubernetes
- доступ к экземляру redis для хранения блокировок

Установка и настройка этих систем выходит за рамки данной инструкции

Для установки kubecronic в кластера kubernetes необходимо отредактировать следующие манифесты (знаком # будут помечены пункты, требующие ввода данных. Для каждого кластера kubernetes данные могут отличаться)

00-crd.yaml. Устанавливается без изменений

```
---
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  annotations:
    controller-gen.kubebuilder.io/version: v0.14.0
  name: cronjobs.kubecronic.vseinstrumenti.ru
spec:
  group: kubecronic.vseinstrumenti.ru
  names:
    kind: CronJob
    listKind: CronJobList
    plural: cronjobs
    singular: cronjob
  scope: Namespaced
  versions:
  - additionalPrinterColumns:
    - jsonPath: .status.active
      name: Active
      type: string
    - jsonPath: .status.datacenter
      name: Datacenter
      type: string
    name: v1
    schema:
      openAPIV3Schema:
```

```
      properties:
        apiVersion:
          description: |-
            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
          type: string
        kind:
          description: |-
            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
          type: string
        metadata:
          type: object
        spec:
          properties:
            concurrencyPolicy:
              description: |-
                Specifies how to treat concurrent executions of a Job.
                Valid values are:


                - "Allow" (default): allows CronJobs to run concurrently;
                - "Forbid": forbids concurrent runs, skipping next run if previous run hasn't finished yet;
                - "Replace": cancels currently running job and replaces it with a new one
              type: string
            failedJobsHistoryLimit:
              format: int32
              minimum: 0
              type: integer
            jobTemplate:
              properties:
                metadata:
                  properties:
                    annotations:
                      additionalProperties:
                        type: string
                      type: object
                    finalizers:
                      items:
                        type: string
                      type: array
                    labels:
                      additionalProperties:
                        type: string
                      type: object
                    name:
                      type: string
                    namespace:
                      type: string
```

```
      type: object
    spec:
      properties:
        backoffLimit:
          format: int32
          type: integer
        completions:
          format: int32
          type: integer
        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.
          properties:
            matchExpressions:
              description: matchExpressions is a list of label selector
                requirements. The requirements are ANDed.
              items:
                description: |-
                  A label selector requirement is a selector that contains values, a key, and an operator
that
                  relates the key and values.
                properties:
                  key:
                    description: key is the label key that the selector
                      applies to.
                    type: string
                  operator:
                    description: |-
                      operator represents a key's relationship to a set of values.
                      Valid operators are In, NotIn, Exists and DoesNotExist.
                    type: string
                  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.
                    items:
                      type: string
                    type: array
                required:
                - key
                - operator
                type: object
              type: array
            matchLabels:
              additionalProperties:
                type: string
              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
```

```
                type: object
                x-kubernetes-map-type: atomic
              template:
                description: PodTemplateSpec describes the data a pod should
                  have when created from a template
                properties:
                  metadata:
                    description: |-
                      Standard object's metadata.
                      More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-
conventions.md#metadata
                    properties:
                      annotations:
                        additionalProperties:
                          type: string
                        type: object
                      finalizers:
                        items:
                          type: string
                        type: array
                      labels:
                        additionalProperties:
                          type: string
                        type: object
                      name:
                        type: string
                      namespace:
                        type: string
                    type: object
                  spec:
                    description: |-
                      Specification of the desired behavior of the pod.
                      More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-
conventions.md#spec-and-status
                    properties:
                      activeDeadlineSeconds:
                        description: |-
                          Optional duration in seconds the pod may be active on the node relative to
                          StartTime before the system will actively try to mark it failed and kill associated
containers.
                          Value must be a positive integer.
                        format: int64
                        type: integer
                      affinity:
                        description: If specified, the pod's scheduling constraints
                        properties:
                          nodeAffinity:
                            description: Describes node affinity scheduling
                              rules for the pod.
                            properties:
                              preferredDuringSchedulingIgnoredDuringExecution:
                                description: |-
                                  The scheduler will prefer to schedule pods to nodes that satisfy
                                  the affinity expressions specified by this field, but it may choose
                                  a node that violates one or more of the expressions. The node that is
                                  most preferred is the one with the greatest sum of weights, i.e.
```

```
                  for each node that meets all of the scheduling requirements (resource
                  request, requiredDuringScheduling affinity expressions, etc.),
                  compute a sum by iterating through the elements of this field and adding
                  "weight" to the sum if the node matches the corresponding matchExpressions;
the
                  node(s) with the highest sum are the most preferred.
                items:
                  description: |-
                    An empty preferred scheduling term matches all objects with implicit weight 0
                    (i.e. it's a no-op). A null preferred scheduling term matches no objects (i.e. is
also a no-op).
                  properties:
                    preference:
                      description: A node selector term, associated
                        with the corresponding weight.
                      properties:
                        matchExpressions:
                          description: A list of node selector
                            requirements by node's labels.
                          items:
                            description: |-
                              A node selector requirement is a selector that contains values, a key,
and an operator
                              that relates the key and values.
                            properties:
                              key:
                                description: The label key
                                  that the selector applies
                                  to.
                                type: string
                              operator:
                                description: |-
                                  Represents a key's relationship to a set of values.
                                  Valid operators are In, NotIn, Exists, DoesNotExist. Gt, and Lt.
                                type: string
                              values:
                                description: |-
                                  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. If the operator is Gt or Lt, the values
                                  array must have a single element, which will be interpreted as an
integer.
                                  This array is replaced during a strategic merge patch.
                                items:
                                  type: string
                                type: array
                            required:
                            - key
                            - operator
                            type: object
                          type: array
                        matchFields:
                          description: A list of node selector
                            requirements by node's fields.
                          items:
```

```
                    description: |-
                      A node selector requirement is a selector that contains values, a key,
and an operator
                      that relates the key and values.
                    properties:
                      key:
                        description: The label key
                          that the selector applies
                          to.
                        type: string
                      operator:
                        description: |-
                          Represents a key's relationship to a set of values.
                          Valid operators are In, NotIn, Exists, DoesNotExist. Gt, and Lt.
                        type: string
                      values:
                        description: |-
                          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. If the operator is Gt or Lt, the values
                          array must have a single element, which will be interpreted as an
integer.
                          This array is replaced during a strategic merge patch.
                        items:
                          type: string
                        type: array
                    required:
                    - key
                    - operator
                    type: object
                  type: array
                type: object
                x-kubernetes-map-type: atomic
              weight:
                description: Weight associated with
                  matching the corresponding nodeSelectorTerm,
                  in the range 1-100.
                format: int32
                type: integer
            required:
            - preference
            - weight
            type: object
          type: array
        requiredDuringSchedulingIgnoredDuringExecution:
          description: |-
            If the affinity requirements specified by this field are not met at
            scheduling time, the pod will not be scheduled onto the node.
            If the affinity requirements specified by this field cease to be met
            at some point during pod execution (e.g. due to an update), the system
            may or may not try to eventually evict the pod from its node.
          properties:
            nodeSelectorTerms:
              description: Required. A list of node
                selector terms. The terms are ORed.
```

```
                        items:
                          description: |-
                            A null or empty node selector term matches no objects. The requirements
of
                            them are ANDed.
                            The TopologySelectorTerm type implements a subset of the
NodeSelectorTerm.
                          properties:
                            matchExpressions:
                              description: A list of node selector
                                requirements by node's labels.
                              items:
                                description: |-
                                  A node selector requirement is a selector that contains values, a key,
and an operator
                                  that relates the key and values.
                                properties:
                                  key:
                                    description: The label key
                                      that the selector applies
                                      to.
                                    type: string
                                  operator:
                                    description: |-
                                      Represents a key's relationship to a set of values.
                                      Valid operators are In, NotIn, Exists, DoesNotExist. Gt, and Lt.
                                    type: string
                                  values:
                                    description: |-
                                      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. If the operator is Gt or Lt, the values
                                      array must have a single element, which will be interpreted as an
integer.
                                      This array is replaced during a strategic merge patch.
                                    items:
                                      type: string
                                    type: array
                                required:
                                - key
                                - operator
                                type: object
                              type: array
                            matchFields:
                              description: A list of node selector
                                requirements by node's fields.
                              items:
                                description: |-
                                  A node selector requirement is a selector that contains values, a key,
and an operator
                                  that relates the key and values.
                                properties:
                                  key:
                                    description: The label key
                                      that the selector applies
```

```
                                    to.
                                  type: string
                                operator:
                                  description: |-
                                    Represents a key's relationship to a set of values.
                                    Valid operators are In, NotIn, Exists, DoesNotExist. Gt, and Lt.
                                  type: string
                                values:
                                  description: |-
                                    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. If the operator is Gt or Lt, the values
                                    array must have a single element, which will be interpreted as an
integer.
                                    This array is replaced during a strategic merge patch.
                                  items:
                                    type: string
                                  type: array
                              required:
                              - key
                              - operator
                              type: object
                            type: array
                          type: object
                          x-kubernetes-map-type: atomic
                        type: array
                      required:
                      - nodeSelectorTerms
                      type: object
                      x-kubernetes-map-type: atomic
                  type: object
                podAffinity:
                  description: Describes pod affinity scheduling
                    rules (e.g. co-locate this pod in the same node,
                    zone, etc. as some other pod(s)).
                  properties:
                    preferredDuringSchedulingIgnoredDuringExecution:
                      description: |-
                        The scheduler will prefer to schedule pods to nodes that satisfy
                        the affinity expressions specified by this field, but it may choose
                        a node that violates one or more of the expressions. The node that is
                        most preferred is the one with the greatest sum of weights, i.e.
                        for each node that meets all of the scheduling requirements (resource
                        request, requiredDuringScheduling affinity expressions, etc.),
                        compute a sum by iterating through the elements of this field and adding
                        "weight" to the sum if the node has pods which matches the corresponding
podAffinityTerm; the
                        node(s) with the highest sum are the most preferred.
                      items:
                        description: The weights of all of the matched
                          WeightedPodAffinityTerm fields are added
                          per-node to find the most preferred node(s)
                        properties:
                          podAffinityTerm:
                            description: Required. A pod affinity
```

```
                term, associated with the corresponding
                weight.
              properties:
                labelSelector:
                  description: A label query over
                    a set of resources, in this case
                    pods.
                  properties:
                    matchExpressions:
                      description: matchExpressions
                        is a list of label selector
                        requirements. The requirements
                        are ANDed.
                      items:
                        description: |-
                          A label selector requirement is a selector that contains values, a key,
and an operator that
                          relates the key and values.
                        properties:
                          key:
                            description: key is the
                              label key that the selector
                              applies to.
                            type: string
                          operator:
                            description: |-
                              operator represents a key's relationship to a set of values.
                              Valid operators are In, NotIn, Exists and DoesNotExist.
                            type: string
                          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.
                            items:
                              type: string
                            type: array
                        required:
                        - key
                        - operator
                        type: object
                      type: array
                    matchLabels:
                      additionalProperties:
                        type: string
                      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
```

```
                type: object
                x-kubernetes-map-type: atomic
              namespaceSelector:
                description: |-
                  A label query over the set of namespaces that the term applies to.
                  The term is applied to the union of the namespaces selected by this field
                  and the ones listed in the namespaces field.
                  null selector and null or empty namespaces list means "this pod's
namespace".
                  An empty selector ({}) matches all namespaces.
                properties:
                  matchExpressions:
                    description: matchExpressions
                      is a list of label selector
                      requirements. The requirements
                      are ANDed.
                    items:
                      description: |-
                        A label selector requirement is a selector that contains values, a key,
and an operator that
                        relates the key and values.
                      properties:
                        key:
                          description: key is the
                            label key that the selector
                            applies to.
                          type: string
                        operator:
                          description: |-
                            operator represents a key's relationship to a set of values.
                            Valid operators are In, NotIn, Exists and DoesNotExist.
                          type: string
                        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.
                          items:
                            type: string
                          type: array
                      required:
                      - key
                      - operator
                      type: object
                    type: array
                  matchLabels:
                    additionalProperties:
                      type: string
                    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
                                      type: object
                                      x-kubernetes-map-type: atomic
                                    namespaces:
                                      description: |-
                                        namespaces specifies a static list of namespace names that the term
applies to.
                                        The term is applied to the union of the namespaces listed in this field
                                        and the ones selected by namespaceSelector.
                                        null or empty namespaces list and null namespaceSelector means "this
pod's namespace".
                                      items:
                                        type: string
                                      type: array
                                    topologyKey:
                                      description: |-
                                        This pod should be co-located (affinity) or not co-located (anti-affinity)
with the pods matching
                                        the labelSelector in the specified namespaces, where co-located is
defined as running on a node
                                        whose value of the label with key topologyKey matches that of any node
on which any of the
                                        selected pods is running.
                                        Empty topologyKey is not allowed.
                                      type: string
                                  required:
                                  - topologyKey
                                  type: object
                                weight:
                                  description: |-
                                    weight associated with matching the corresponding podAffinityTerm,
                                    in the range 1-100.
                                  format: int32
                                  type: integer
                              required:
                              - podAffinityTerm
                              - weight
                              type: object
                            type: array
                          requiredDuringSchedulingIgnoredDuringExecution:
                            description: |-
                              If the affinity requirements specified by this field are not met at
                              scheduling time, the pod will not be scheduled onto the node.
                              If the affinity requirements specified by this field cease to be met
                              at some point during pod execution (e.g. due to a pod label update), the
                              system may or may not try to eventually evict the pod from its node.
                              When there are multiple elements, the lists of nodes corresponding to each
                              podAffinityTerm are intersected, i.e. all terms must be satisfied.
                            items:
                              description: |-
                                Defines a set of pods (namely those matching the labelSelector
                                relative to the given namespace(s)) that this pod should be
                                co-located (affinity) or not co-located (anti-affinity) with,
                                where co-located is defined as running on a node whose value of
```

```
                the label with key <topologyKey> matches that of any node on which
                a pod of the set of pods is running
              properties:
                labelSelector:
                  description: A label query over a set
                    of resources, in this case pods.
                  properties:
                    matchExpressions:
                      description: matchExpressions is
                        a list of label selector requirements.
                        The requirements are ANDed.
                      items:
                        description: |-
                          A label selector requirement is a selector that contains values, a key,
and an operator that
                          relates the key and values.
                        properties:
                          key:
                            description: key is the label
                              key that the selector applies
                              to.
                            type: string
                          operator:
                            description: |-
                              operator represents a key's relationship to a set of values.
                              Valid operators are In, NotIn, Exists and DoesNotExist.
                            type: string
                          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.
                            items:
                              type: string
                            type: array
                        required:
                        - key
                        - operator
                        type: object
                      type: array
                    matchLabels:
                      additionalProperties:
                        type: string
                      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
                  type: object
                  x-kubernetes-map-type: atomic
```

```
                    namespaceSelector:
                      description: |-
                        A label query over the set of namespaces that the term applies to.
                        The term is applied to the union of the namespaces selected by this field
                        and the ones listed in the namespaces field.
                        null selector and null or empty namespaces list means "this pod's
namespace".
                        An empty selector ({}) matches all namespaces.
                      properties:
                        matchExpressions:
                          description: matchExpressions is
                            a list of label selector requirements.
                            The requirements are ANDed.
                          items:
                            description: |-
                              A label selector requirement is a selector that contains values, a key,
and an operator that
                              relates the key and values.
                            properties:
                              key:
                                description: key is the label
                                  key that the selector applies
                                  to.
                                type: string
                              operator:
                                description: |-
                                  operator represents a key's relationship to a set of values.
                                  Valid operators are In, NotIn, Exists and DoesNotExist.
                                type: string
                              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.
                                items:
                                  type: string
                                type: array
                            required:
                            - key
                            - operator
                            type: object
                          type: array
                        matchLabels:
                          additionalProperties:
                            type: string
                          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
```

```
                type: object
                x-kubernetes-map-type: atomic
              namespaces:
                description: |-
                  namespaces specifies a static list of namespace names that the term
applies to.
                  The term is applied to the union of the namespaces listed in this field
                  and the ones selected by namespaceSelector.
                  null or empty namespaces list and null namespaceSelector means "this
pod's namespace".
                items:
                  type: string
                type: array
              topologyKey:
                description: |-
                  This pod should be co-located (affinity) or not co-located (anti-affinity) with
the pods matching
                  the labelSelector in the specified namespaces, where co-located is defined
as running on a node
                  whose value of the label with key topologyKey matches that of any node on
which any of the
                  selected pods is running.
                  Empty topologyKey is not allowed.
                type: string
            required:
            - topologyKey
            type: object
          type: array
        type: object
      podAntiAffinity:
        description: Describes pod anti-affinity scheduling
          rules (e.g. avoid putting this pod in the same
          node, zone, etc. as some other pod(s)).
        properties:
          preferredDuringSchedulingIgnoredDuringExecution:
            description: |-
              The scheduler will prefer to schedule pods to nodes that satisfy
              the anti-affinity expressions specified by this field, but it may choose
              a node that violates one or more of the expressions. The node that is
              most preferred is the one with the greatest sum of weights, i.e.
              for each node that meets all of the scheduling requirements (resource
              request, requiredDuringScheduling anti-affinity expressions, etc.),
              compute a sum by iterating through the elements of this field and adding
              "weight" to the sum if the node has pods which matches the corresponding
podAffinityTerm; the
              node(s) with the highest sum are the most preferred.
            items:
              description: The weights of all of the matched
                WeightedPodAffinityTerm fields are added
                per-node to find the most preferred node(s)
              properties:
                podAffinityTerm:
                  description: Required. A pod affinity
                    term, associated with the corresponding
                    weight.
                  properties:
```

```
                    labelSelector:
                      description: A label query over
                        a set of resources, in this case
                        pods.
                      properties:
                        matchExpressions:
                          description: matchExpressions
                            is a list of label selector
                            requirements. The requirements
                            are ANDed.
                          items:
                            description: |-
                              A label selector requirement is a selector that contains values, a key,
and an operator that

                              relates the key and values.
                            properties:
                              key:
                                description: key is the
                                  label key that the selector
                                  applies to.
                                type: string
                              operator:
                                description: |-
                                  operator represents a key's relationship to a set of values.
                                  Valid operators are In, NotIn, Exists and DoesNotExist.
                                type: string
                              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.
                                items:
                                  type: string
                                type: array
                            required:
                            - key
                            - operator
                            type: object
                          type: array
                        matchLabels:
                          additionalProperties:
                            type: string
                          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
                      type: object
                      x-kubernetes-map-type: atomic
                    namespaceSelector:
```

```
                    description: |-
                      A label query over the set of namespaces that the term applies to.
                      The term is applied to the union of the namespaces selected by this field
                      and the ones listed in the namespaces field.
                      null selector and null or empty namespaces list means "this pod's
namespace".
                      An empty selector ({}) matches all namespaces.
                    properties:
                      matchExpressions:
                        description: matchExpressions
                          is a list of label selector
                          requirements. The requirements
                          are ANDed.
                        items:
                          description: |-
                            A label selector requirement is a selector that contains values, a key,
and an operator that
                            relates the key and values.
                          properties:
                            key:
                              description: key is the
                                label key that the selector
                                applies to.
                              type: string
                            operator:
                              description: |-
                                operator represents a key's relationship to a set of values.
                                Valid operators are In, NotIn, Exists and DoesNotExist.
                              type: string
                            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.
                              items:
                                type: string
                              type: array
                          required:
                          - key
                          - operator
                          type: object
                        type: array
                      matchLabels:
                        additionalProperties:
                          type: string
                        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
```

```
                                    type: object
                                    x-kubernetes-map-type: atomic
                                  namespaces:
                                    description: |-
                                      namespaces specifies a static list of namespace names that the term
applies to.
                                      The term is applied to the union of the namespaces listed in this field
                                      and the ones selected by namespaceSelector.
                                      null or empty namespaces list and null namespaceSelector means "this
pod's namespace".
                                    items:
                                      type: string
                                    type: array
                                  topologyKey:
                                    description: |-
                                      This pod should be co-located (affinity) or not co-located (anti-affinity)
with the pods matching
                                      the labelSelector in the specified namespaces, where co-located is
defined as running on a node
                                      whose value of the label with key topologyKey matches that of any node
on which any of the
                                      selected pods is running.
                                      Empty topologyKey is not allowed.
                                    type: string
                                  required:
                                  - topologyKey
                                  type: object
                                weight:
                                  description: |-
                                    weight associated with matching the corresponding podAffinityTerm,
                                    in the range 1-100.
                                  format: int32
                                  type: integer
                              required:
                              - podAffinityTerm
                              - weight
                              type: object
                            type: array
                          requiredDuringSchedulingIgnoredDuringExecution:
                            description: |-
                              If the anti-affinity requirements specified by this field are not met at
                              scheduling time, the pod will not be scheduled onto the node.
                              If the anti-affinity requirements specified by this field cease to be met
                              at some point during pod execution (e.g. due to a pod label update), the
                              system may or may not try to eventually evict the pod from its node.
                              When there are multiple elements, the lists of nodes corresponding to each
                              podAffinityTerm are intersected, i.e. all terms must be satisfied.
                            items:
                              description: |-
                                Defines a set of pods (namely those matching the labelSelector
                                relative to the given namespace(s)) that this pod should be
                                co-located (affinity) or not co-located (anti-affinity) with,
                                where co-located is defined as running on a node whose value of
                                the label with key <topologyKey> matches that of any node on which
                                a pod of the set of pods is running
                              properties:
```

```
                    labelSelector:
                      description: A label query over a set
                        of resources, in this case pods.
                      properties:
                        matchExpressions:
                          description: matchExpressions is
                            a list of label selector requirements.
                            The requirements are ANDed.
                          items:
                            description: |-
                              A label selector requirement is a selector that contains values, a key,
and an operator that
                              relates the key and values.
                            properties:
                              key:
                                description: key is the label
                                  key that the selector applies
                                  to.
                                type: string
                              operator:
                                description: |-
                                  operator represents a key's relationship to a set of values.
                                  Valid operators are In, NotIn, Exists and DoesNotExist.
                                type: string
                              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.
                                items:
                                  type: string
                                type: array
                            required:
                            - key
                            - operator
                            type: object
                          type: array
                        matchLabels:
                          additionalProperties:
                            type: string
                          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
                      type: object
                      x-kubernetes-map-type: atomic
                    namespaceSelector:
                      description: |-
                        A label query over the set of namespaces that the term applies to.
```

```
                The term is applied to the union of the namespaces selected by this field
                and the ones listed in the namespaces field.
                null selector and null or empty namespaces list means "this pod's
namespace".
                An empty selector ({}) matches all namespaces.
              properties:
                matchExpressions:
                  description: matchExpressions is
                    a list of label selector requirements.
                    The requirements are ANDed.
                  items:
                    description: |-
                      A label selector requirement is a selector that contains values, a key,
and an operator that
                      relates the key and values.
                    properties:
                      key:
                        description: key is the label
                          key that the selector applies
                          to.
                        type: string
                      operator:
                        description: |-
                          operator represents a key's relationship to a set of values.
                          Valid operators are In, NotIn, Exists and DoesNotExist.
                        type: string
                      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.
                        items:
                          type: string
                        type: array
                    required:
                    - key
                    - operator
                    type: object
                  type: array
                matchLabels:
                  additionalProperties:
                    type: string
                  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
              type: object
              x-kubernetes-map-type: atomic
            namespaces:
```

```
                                    description: |-
                                      namespaces specifies a static list of namespace names that the term
applies to.
                                      The term is applied to the union of the namespaces listed in this field
                                      and the ones selected by namespaceSelector.
                                      null or empty namespaces list and null namespaceSelector means "this
pod's namespace".
                                    items:
                                      type: string
                                    type: array
                                  topologyKey:
                                    description: |-
                                      This pod should be co-located (affinity) or not co-located (anti-affinity) with
the pods matching
                                      the labelSelector in the specified namespaces, where co-located is defined
as running on a node
                                      whose value of the label with key topologyKey matches that of any node on
which any of the
                                      selected pods is running.
                                      Empty topologyKey is not allowed.
                                    type: string
                                required:
                                - topologyKey
                                type: object
                              type: array
                            type: object
                          type: object
                        automountServiceAccountToken:
                          description: AutomountServiceAccountToken indicates
                            whether a service account token should be automatically
                            mounted.
                          type: boolean
                        containers:
                          description: |-
                            List of containers belonging to the pod.
                            Containers cannot currently be added or removed.
                            There must be at least one container in a Pod.
                            Cannot be updated.
                          items:
                            description: A single application container that
                              you want to run within a pod.
                            properties:
                              args:
                                description: |-
                                  Arguments to the entrypoint.
                                  The container image's CMD is used if this is not provided.
                                  Variable references $(VAR_NAME) are expanded using the container's
environment. If a variable
                                  cannot be resolved, the reference in the input string will be unchanged. Double
$$ are reduced
                                  to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
"$$(VAR_NAME)" will
                                  produce the string literal "$(VAR_NAME)". Escaped references will never be
expanded, regardless
                                  of whether the variable exists or not. Cannot be updated.
```

```
                More info: https://kubernetes.io/docs/tasks/inject-data-application/define-
command-argument-container/#running-a-command-in-a-shell
              items:
                type: string
              type: array
            command:
              description: |-
                Entrypoint array. Not executed within a shell.
                The container image's ENTRYPOINT is used if this is not provided.
                Variable references $(VAR_NAME) are expanded using the container's
environment. If a variable
                cannot be resolved, the reference in the input string will be unchanged. Double
$$ are reduced
                to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
"$$(VAR_NAME)" will
                produce the string literal "$(VAR_NAME)". Escaped references will never be
expanded, regardless
                of whether the variable exists or not. Cannot be updated.
                More info: https://kubernetes.io/docs/tasks/inject-data-application/define-
command-argument-container/#running-a-command-in-a-shell
              items:
                type: string
              type: array
            env:
              description: |-
                List of environment variables to set in the container.
                Cannot be updated.
              items:
                description: EnvVar represents an environment
                  variable present in a Container.
                properties:
                  name:
                    description: Name of the environment variable.
                      Must be a C_IDENTIFIER.
                    type: string
                  value:
                    description: |-
                      Variable references $(VAR_NAME) are expanded
                      using the previously defined environment variables in the container and
                      any service environment variables. If a variable cannot be resolved,
                      the reference in the input string will be unchanged. Double $$ are reduced
                      to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
                      "$$(VAR_NAME)" will produce the string literal "$(VAR_NAME)".
                      Escaped references will never be expanded, regardless of whether the
variable
                      exists or not.
                      Defaults to "".
                    type: string
                  valueFrom:
                    description: Source for the environment
                      variable's value. Cannot be used if
                      value is not empty.
                    properties:
                      configMapKeyRef:
                        description: Selects a key of a ConfigMap.
                        properties:
```

```
                                    key:
                                      description: The key to select.
                                      type: string
                                    name:
                                      description: |-
                                        Name of the referent.
                                        More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                        TODO: Add other useful fields. apiVersion, kind, uid?
                                      type: string
                                    optional:
                                      description: Specify whether the
                                        ConfigMap or its key must be
                                        defined
                                      type: boolean
                                  required:
                                  - key
                                  type: object
                                  x-kubernetes-map-type: atomic
                                fieldRef:
                                  description: |-
                                    Selects a field of the pod: supports metadata.name,
metadata.namespace, `metadata.labels['<KEY>']`, `metadata.annotations['<KEY>']`,
                                    spec.nodeName, spec.serviceAccountName, status.hostIP, status.podIP,
status.podIPs.
                                  properties:
                                    apiVersion:
                                      description: Version of the schema
                                        the FieldPath is written in
                                        terms of, defaults to "v1".
                                      type: string
                                    fieldPath:
                                      description: Path of the field
                                        to select in the specified API
                                        version.
                                      type: string
                                  required:
                                  - fieldPath
                                  type: object
                                  x-kubernetes-map-type: atomic
                                resourceFieldRef:
                                  description: |-
                                    Selects a resource of the container: only resources limits and requests
                                    (limits.cpu, limits.memory, limits.ephemeral-storage, requests.cpu,
requests.memory and requests.ephemeral-storage) are currently supported.
                                  properties:
                                    containerName:
                                      description: 'Container name:
                                        required for volumes, optional
                                        for env vars'
                                      type: string
                                    divisor:
                                      anyOf:
                                      - type: integer
                                      - type: string
                                      description: Specifies the output
```

```
                                    format of the exposed resources,
                                    defaults to "1"
                                  pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                  x-kubernetes-int-or-string: true
                                resource:
                                  description: 'Required: resource
                                    to select'
                                  type: string
                              required:
                              - resource
                              type: object
                              x-kubernetes-map-type: atomic
                            secretKeyRef:
                              description: Selects a key of a secret
                                in the pod's namespace
                              properties:
                                key:
                                  description: The key of the secret
                                    to select from.  Must be a valid
                                    secret key.
                                  type: string
                                name:
                                  description: |-
                                    Name of the referent.
                                    More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                    TODO: Add other useful fields. apiVersion, kind, uid?
                                  type: string
                                optional:
                                  description: Specify whether the
                                    Secret or its key must be defined
                                  type: boolean
                              required:
                              - key
                              type: object
                              x-kubernetes-map-type: atomic
                          type: object
                        required:
                        - name
                        type: object
                      type: array
                    envFrom:
                      description: |-
                        List of sources to populate environment variables in the container.
                        The keys defined within a source must be a C_IDENTIFIER. All invalid keys
                        will be reported as an event when the container is starting. When a key exists in
multiple
                        sources, the value associated with the last source will take precedence.
                        Values defined by an Env with a duplicate key will take precedence.
                        Cannot be updated.
                      items:
                        description: EnvFromSource represents the
                          source of a set of ConfigMaps
                        properties:
                          configMapRef:
```

```
                        description: The ConfigMap to select from
                        properties:
                          name:
                            description: |-
                              Name of the referent.
                              More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                              TODO: Add other useful fields. apiVersion, kind, uid?
                            type: string
                          optional:
                            description: Specify whether the ConfigMap
                              must be defined
                            type: boolean
                        type: object
                        x-kubernetes-map-type: atomic
                      prefix:
                        description: An optional identifier to
                          prepend to each key in the ConfigMap.
                          Must be a C_IDENTIFIER.
                        type: string
                      secretRef:
                        description: The Secret to select from
                        properties:
                          name:
                            description: |-
                              Name of the referent.
                              More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                              TODO: Add other useful fields. apiVersion, kind, uid?
                            type: string
                          optional:
                            description: Specify whether the Secret
                              must be defined
                            type: boolean
                        type: object
                        x-kubernetes-map-type: atomic
                    type: object
                  type: array
                image:
                  description: |-
                    Container image name.
                    More info: https://kubernetes.io/docs/concepts/containers/images
                    This field is optional to allow higher level config management to default or
override
                    container images in workload controllers like Deployments and StatefulSets.
                  type: string
                imagePullPolicy:
                  description: |-
                    Image pull policy.
                    One of Always, Never, IfNotPresent.
                    Defaults to Always if :latest tag is specified, or IfNotPresent otherwise.
                    Cannot be updated.
                    More info: https://kubernetes.io/docs/concepts/containers/images#updating-
images
                  type: string
                lifecycle:
```

```
                        description: |-
                          Actions that the management system should take in response to container
lifecycle events.
                          Cannot be updated.
                        properties:
                          postStart:
                            description: |-
                              PostStart is called immediately after a container is created. If the handler fails,
                              the container is terminated and restarted according to its restart policy.
                              Other management of the container blocks until the hook completes.
                              More info: https://kubernetes.io/docs/concepts/containers/container-lifecycle-
hooks/#container-hooks
                            properties:
                              exec:
                                description: Exec specifies the action
                                  to take.
                                properties:
                                  command:
                                    description: |-
                                      Command is the command line to execute inside the container, the
working directory for the
                                      command  is root ('/') in the container's filesystem. The command is
simply exec'd, it is
                                      not run inside a shell, so traditional shell instructions ('|', etc) won't work.
To use
                                      a shell, you need to explicitly call out to that shell.
                                      Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                                    items:
                                      type: string
                                    type: array
                                type: object
                              httpGet:
                                description: HTTPGet specifies the http
                                  request to perform.
                                properties:
                                  host:
                                    description: |-
                                      Host name to connect to, defaults to the pod IP. You probably want to
set
                                      "Host" in httpHeaders instead.
                                    type: string
                                  httpHeaders:
                                    description: Custom headers to set
                                      in the request. HTTP allows repeated
                                      headers.
                                    items:
                                      description: HTTPHeader describes
                                        a custom header to be used in
                                        HTTP probes
                                      properties:
                                        name:
                                          description: The header field
                                            name
                                          type: string
                                        value:
                                          description: The header field
```

```
                value
              type: string
            required:
            - name
            - value
            type: object
          type: array
        path:
          description: Path to access on the
            HTTP server.
          type: string
        port:
          anyOf:
          - type: integer
          - type: string
          description: |-
            Name or number of the port to access on the container.
            Number must be in the range 1 to 65535.
            Name must be an IANA_SVC_NAME.
          x-kubernetes-int-or-string: true
        scheme:
          description: |-
            Scheme to use for connecting to the host.
            Defaults to HTTP.
          type: string
      required:
      - port
      type: object
    tcpSocket:
      description: |-
        Deprecated. TCPSocket is NOT supported as a LifecycleHandler and kept
        for the backward compatibility. There are no validation of this field and
        lifecycle hooks will fail in runtime when tcp handler is specified.
      properties:
        host:
          description: 'Optional: Host name
            to connect to, defaults to the
            pod IP.'
          type: string
        port:
          anyOf:
          - type: integer
          - type: string
          description: |-
            Number or name of the port to access on the container.
            Number must be in the range 1 to 65535.
            Name must be an IANA_SVC_NAME.
          x-kubernetes-int-or-string: true
      required:
      - port
      type: object
  type: object
preStop:
  description: |-
    PreStop is called immediately before a container is terminated due to an
    API request or management event such as liveness/startup probe failure,
```

```
                        preemption, resource contention, etc. The handler is not called if the
                        container crashes or exits. The Pod's termination grace period countdown
begins before the
                        PreStop hook is executed. Regardless of the outcome of the handler, the
                        container will eventually terminate within the Pod's termination grace
                        period (unless delayed by finalizers). Other management of the container
blocks until the hook completes
                        or until the termination grace period is reached.
                        More info: https://kubernetes.io/docs/concepts/containers/container-lifecycle-
hooks/#container-hooks
                      properties:
                        exec:
                          description: Exec specifies the action
                            to take.
                          properties:
                            command:
                              description: |-
                                Command is the command line to execute inside the container, the
working directory for the
                                command  is root ('/') in the container's filesystem. The command is
simply exec'd, it is
                                not run inside a shell, so traditional shell instructions ('|', etc) won't work.
To use
                                a shell, you need to explicitly call out to that shell.
                                Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                              items:
                                type: string
                              type: array
                          type: object
                        httpGet:
                          description: HTTPGet specifies the http
                            request to perform.
                          properties:
                            host:
                              description: |-
                                Host name to connect to, defaults to the pod IP. You probably want to
set
                                "Host" in httpHeaders instead.
                              type: string
                            httpHeaders:
                              description: Custom headers to set
                                in the request. HTTP allows repeated
                                headers.
                              items:
                                description: HTTPHeader describes
                                  a custom header to be used in
                                  HTTP probes
                                properties:
                                  name:
                                    description: The header field
                                      name
                                    type: string
                                  value:
                                    description: The header field
                                      value
                                    type: string
```

```
            required:
            - name
            - value
            type: object
          type: array
        path:
          description: Path to access on the
            HTTP server.
          type: string
        port:
          anyOf:
          - type: integer
          - type: string
          description: |-
            Name or number of the port to access on the container.
            Number must be in the range 1 to 65535.
            Name must be an IANA_SVC_NAME.
          x-kubernetes-int-or-string: true
        scheme:
          description: |-
            Scheme to use for connecting to the host.
            Defaults to HTTP.
          type: string
      required:
      - port
      type: object
    tcpSocket:
      description: |-
        Deprecated. TCPSocket is NOT supported as a LifecycleHandler and kept
        for the backward compatibility. There are no validation of this field and
        lifecycle hooks will fail in runtime when tcp handler is specified.
      properties:
        host:
          description: 'Optional: Host name
            to connect to, defaults to the
            pod IP.'
          type: string
        port:
          anyOf:
          - type: integer
          - type: string
          description: |-
            Number or name of the port to access on the container.
            Number must be in the range 1 to 65535.
            Name must be an IANA_SVC_NAME.
          x-kubernetes-int-or-string: true
      required:
      - port
      type: object
  type: object
type: object
livenessProbe:
  description: |-
    Periodic probe of container liveness.
    Container will be restarted if the probe fails.
    Cannot be updated.
```

```
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
              properties:
                exec:
                  description: Exec specifies the action to
                    take.
                  properties:
                    command:
                      description: |-
                        Command is the command line to execute inside the container, the working
directory for the
                        command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                        not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                        a shell, you need to explicitly call out to that shell.
                        Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                      items:
                        type: string
                      type: array
                  type: object
                failureThreshold:
                  description: |-
                    Minimum consecutive failures for the probe to be considered failed after
having succeeded.
                    Defaults to 3. Minimum value is 1.
                  format: int32
                  type: integer
                grpc:
                  description: |-
                    GRPC specifies an action involving a GRPC port.
                    This is a beta field and requires enabling GRPCContainerProbe feature gate.
                  properties:
                    port:
                      description: Port number of the gRPC
                        service. Number must be in the range
                        1 to 65535.
                      format: int32
                      type: integer
                    service:
                      description: |-
                        Service is the name of the service to place in the gRPC
HealthCheckRequest
                        (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


                        If this is not specified, the default behavior is defined by gRPC.
                      type: string
                  required:
                  - port
                  type: object
                httpGet:
                  description: HTTPGet specifies the http
                    request to perform.
                  properties:
                    host:
```

```
                description: |-
                  Host name to connect to, defaults to the pod IP. You probably want to set
                  "Host" in httpHeaders instead.
                type: string
              httpHeaders:
                description: Custom headers to set in
                  the request. HTTP allows repeated
                  headers.
                items:
                  description: HTTPHeader describes
                    a custom header to be used in HTTP
                    probes
                  properties:
                    name:
                      description: The header field
                        name
                      type: string
                    value:
                      description: The header field
                        value
                      type: string
                  required:
                  - name
                  - value
                  type: object
                type: array
              path:
                description: Path to access on the HTTP
                  server.
                type: string
              port:
                anyOf:
                - type: integer
                - type: string
                description: |-
                  Name or number of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
              scheme:
                description: |-
                  Scheme to use for connecting to the host.
                  Defaults to HTTP.
                type: string
            required:
            - port
            type: object
          initialDelaySeconds:
            description: |-
              Number of seconds after the container has started before liveness probes are
initiated.
              More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
            format: int32
            type: integer
          periodSeconds:
```

```
      description: |-
        How often (in seconds) to perform the probe.
        Default to 10 seconds. Minimum value is 1.
      format: int32
      type: integer
    successThreshold:
      description: |-
        Minimum consecutive successes for the probe to be considered successful
after having failed.
        Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
      format: int32
      type: integer
    tcpSocket:
      description: TCPSocket specifies an action
        involving a TCP port.
      properties:
        host:
          description: 'Optional: Host name to
            connect to, defaults to the pod IP.'
          type: string
        port:
          anyOf:
          - type: integer
          - type: string
          description: |-
            Number or name of the port to access on the container.
            Number must be in the range 1 to 65535.
            Name must be an IANA_SVC_NAME.
          x-kubernetes-int-or-string: true
      required:
      - port
      type: object
    terminationGracePeriodSeconds:
      description: |-
        Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
        The grace period is the duration in seconds after the processes running in the
pod are sent
        a termination signal and the time when the processes are forcibly halted with
a kill signal.
        Set this value longer than the expected cleanup time for your process.
        If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
        value overrides the value provided by the pod spec.
        Value must be non-negative integer. The value zero indicates stop
immediately via
        the kill signal (no opportunity to shut down).
        This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
        Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
      format: int64
      type: integer
    timeoutSeconds:
      description: |-
        Number of seconds after which the probe times out.
        Defaults to 1 second. Minimum value is 1.
```

```
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                format: int32
                type: integer
            type: object
          name:
            description: |-
              Name of the container specified as a DNS_LABEL.
              Each container in a pod must have a unique name (DNS_LABEL).
              Cannot be updated.
            type: string
          ports:
            description: |-
              List of ports to expose from the container. Exposing a port here gives
              the system additional information about the network connections a
              container uses, but is primarily informational. Not specifying a port here
              DOES NOT prevent that port from being exposed. Any port which is
              listening on the default "0.0.0.0" address inside a container will be
              accessible from the network.
              Cannot be updated.
            items:
              description: ContainerPort represents a network
                port in a single container.
              properties:
                containerPort:
                  description: |-
                    Number of port to expose on the pod's IP address.
                    This must be a valid port number, 0 < x < 65536.
                  format: int32
                  type: integer
                hostIP:
                  description: What host IP to bind the
                    external port to.
                  type: string
                hostPort:
                  description: |-
                    Number of port to expose on the host.
                    If specified, this must be a valid port number, 0 < x < 65536.
                    If HostNetwork is specified, this must match ContainerPort.
                    Most containers do not need this.
                  format: int32
                  type: integer
                name:
                  description: |-
                    If specified, this must be an IANA_SVC_NAME and unique within the pod.
Each
                    named port in a pod must have a unique name. Name for the port that can
be
                    referred to by services.
                  type: string
                protocol:
                  default: TCP
                  description: |-
                    Protocol for port. Must be UDP, TCP, or SCTP.
                    Defaults to "TCP".
                  type: string
```

```
                required:
                - containerPort
                type: object
              type: array
              x-kubernetes-list-map-keys:
              - containerPort
              - protocol
              x-kubernetes-list-type: map
            readinessProbe:
              description: |-
                Periodic probe of container service readiness.
                Container will be removed from service endpoints if the probe fails.
                Cannot be updated.
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
              properties:
                exec:
                  description: Exec specifies the action to
                    take.
                  properties:
                    command:
                      description: |-
                        Command is the command line to execute inside the container, the working
directory for the
                        command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                        not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                        a shell, you need to explicitly call out to that shell.
                        Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                      items:
                        type: string
                      type: array
                  type: object
                failureThreshold:
                  description: |-
                    Minimum consecutive failures for the probe to be considered failed after
having succeeded.
                    Defaults to 3. Minimum value is 1.
                  format: int32
                  type: integer
                grpc:
                  description: |-
                    GRPC specifies an action involving a GRPC port.
                    This is a beta field and requires enabling GRPCContainerProbe feature gate.
                  properties:
                    port:
                      description: Port number of the gRPC
                        service. Number must be in the range
                        1 to 65535.
                      format: int32
                      type: integer
                    service:
                      description: |-
                        Service is the name of the service to place in the gRPC
HealthCheckRequest
```

```
        (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


        If this is not specified, the default behavior is defined by gRPC.
      type: string
    required:
    - port
    type: object
  httpGet:
    description: HTTPGet specifies the http
      request to perform.
    properties:
      host:
        description: |-
          Host name to connect to, defaults to the pod IP. You probably want to set
          "Host" in httpHeaders instead.
        type: string
      httpHeaders:
        description: Custom headers to set in
          the request. HTTP allows repeated
          headers.
        items:
          description: HTTPHeader describes
            a custom header to be used in HTTP
            probes
          properties:
            name:
              description: The header field
                name
              type: string
            value:
              description: The header field
                value
              type: string
          required:
          - name
          - value
          type: object
        type: array
      path:
        description: Path to access on the HTTP
          server.
        type: string
      port:
        anyOf:
        - type: integer
        - type: string
        description: |-
          Name or number of the port to access on the container.
          Number must be in the range 1 to 65535.
          Name must be an IANA_SVC_NAME.
        x-kubernetes-int-or-string: true
      scheme:
        description: |-
          Scheme to use for connecting to the host.
          Defaults to HTTP.
```

```
                type: string
              required:
              - port
              type: object
            initialDelaySeconds:
              description: |-
                Number of seconds after the container has started before liveness probes are
initiated.
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
              format: int32
              type: integer
            periodSeconds:
              description: |-
                How often (in seconds) to perform the probe.
                Default to 10 seconds. Minimum value is 1.
              format: int32
              type: integer
            successThreshold:
              description: |-
                Minimum consecutive successes for the probe to be considered successful
after having failed.
                Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
              format: int32
              type: integer
            tcpSocket:
              description: TCPSocket specifies an action
                involving a TCP port.
              properties:
                host:
                  description: 'Optional: Host name to
                    connect to, defaults to the pod IP.'
                  type: string
                port:
                  anyOf:
                  - type: integer
                  - type: string
                  description: |-
                    Number or name of the port to access on the container.
                    Number must be in the range 1 to 65535.
                    Name must be an IANA_SVC_NAME.
                  x-kubernetes-int-or-string: true
              required:
              - port
              type: object
            terminationGracePeriodSeconds:
              description: |-
                Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
                The grace period is the duration in seconds after the processes running in the
pod are sent
                a termination signal and the time when the processes are forcibly halted with
a kill signal.
                Set this value longer than the expected cleanup time for your process.
                If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
```

```
                                value overrides the value provided by the pod spec.
                                Value must be non-negative integer. The value zero indicates stop
immediately via
                                the kill signal (no opportunity to shut down).
                                This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
                                Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
                              format: int64
                              type: integer
                            timeoutSeconds:
                              description: |-
                                Number of seconds after which the probe times out.
                                Defaults to 1 second. Minimum value is 1.
                                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                              format: int32
                              type: integer
                          type: object
                        resources:
                          description: |-
                            Compute Resources required by this container.
                            Cannot be updated.
                            More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-
containers/
                          properties:
                            limits:
                              additionalProperties:
                                anyOf:
                                - type: integer
                                - type: string
                                pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                x-kubernetes-int-or-string: true
                              description: |-
                                Limits describes the maximum amount of compute resources allowed.
                                More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                              type: object
                            requests:
                              additionalProperties:
                                anyOf:
                                - type: integer
                                - type: string
                                pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                x-kubernetes-int-or-string: true
                              description: |-
                                Requests describes the minimum amount of compute resources required.
                                If Requests is omitted for a container, it defaults to Limits if that is explicitly
specified,
                                otherwise to an implementation-defined value.
                                More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                              type: object
                          type: object
                        securityContext:
```

```
                    description: |-
                      SecurityContext defines the security options the container should be run with.
                      If set, the fields of SecurityContext override the equivalent fields of
PodSecurityContext.
                      More info: https://kubernetes.io/docs/tasks/configure-pod-container/security-
context/
                    properties:
                      allowPrivilegeEscalation:
                        description: |-
                          AllowPrivilegeEscalation controls whether a process can gain more
                          privileges than its parent process. This bool directly controls if
                          the no_new_privs flag will be set on the container process.
                          AllowPrivilegeEscalation is true always when the container is:
                          1) run as Privileged
                          2) has CAP_SYS_ADMIN
                          Note that this field cannot be set when spec.os.name is windows.
                        type: boolean
                      capabilities:
                        description: |-
                          The capabilities to add/drop when running containers.
                          Defaults to the default set of capabilities granted by the container runtime.
                          Note that this field cannot be set when spec.os.name is windows.
                        properties:
                          add:
                            description: Added capabilities
                            items:
                              description: Capability represent
                                POSIX capabilities type
                              type: string
                            type: array
                          drop:
                            description: Removed capabilities
                            items:
                              description: Capability represent
                                POSIX capabilities type
                              type: string
                            type: array
                        type: object
                      privileged:
                        description: |-
                          Run container in privileged mode.
                          Processes in privileged containers are essentially equivalent to root on the
host.
                          Defaults to false.
                          Note that this field cannot be set when spec.os.name is windows.
                        type: boolean
                      procMount:
                        description: |-
                          procMount denotes the type of proc mount to use for the containers.
                          The default is DefaultProcMount which uses the container runtime defaults for
                          readonly paths and masked paths.
                          This requires the ProcMountType feature flag to be enabled.
                          Note that this field cannot be set when spec.os.name is windows.
                        type: string
                      readOnlyRootFilesystem:
                        description: |-
```

```
      Whether this container has a read-only root filesystem.
      Default is false.
      Note that this field cannot be set when spec.os.name is windows.
    type: boolean
  runAsGroup:
    description: |-
      The GID to run the entrypoint of the container process.
      Uses runtime default if unset.
      May also be set in PodSecurityContext.  If set in both SecurityContext and
      PodSecurityContext, the value specified in SecurityContext takes precedence.
      Note that this field cannot be set when spec.os.name is windows.
    format: int64
    type: integer
  runAsNonRoot:
    description: |-
      Indicates that the container must run as a non-root user.
      If true, the Kubelet will validate the image at runtime to ensure that it
      does not run as UID 0 (root) and fail to start the container if it does.
      If unset or false, no such validation will be performed.
      May also be set in PodSecurityContext.  If set in both SecurityContext and
      PodSecurityContext, the value specified in SecurityContext takes precedence.
    type: boolean
  runAsUser:
    description: |-
      The UID to run the entrypoint of the container process.
      Defaults to user specified in image metadata if unspecified.
      May also be set in PodSecurityContext.  If set in both SecurityContext and
      PodSecurityContext, the value specified in SecurityContext takes precedence.
      Note that this field cannot be set when spec.os.name is windows.
    format: int64
    type: integer
  seLinuxOptions:
    description: |-
      The SELinux context to be applied to the container.
      If unspecified, the container runtime will allocate a random SELinux context
for each
      container.  May also be set in PodSecurityContext.  If set in both
SecurityContext and
      PodSecurityContext, the value specified in SecurityContext takes precedence.
      Note that this field cannot be set when spec.os.name is windows.
    properties:
      level:
        description: Level is SELinux level
          label that applies to the container.
        type: string
      role:
        description: Role is a SELinux role
          label that applies to the container.
        type: string
      type:
        description: Type is a SELinux type
          label that applies to the container.
        type: string
      user:
        description: User is a SELinux user
          label that applies to the container.
```

```
                type: string
              type: object
            seccompProfile:
              description: |-
                The seccomp options to use by this container. If seccomp options are
                provided at both the pod & container level, the container options
                override the pod options.
                Note that this field cannot be set when spec.os.name is windows.
              properties:
                localhostProfile:
                  description: |-
                    localhostProfile indicates a profile defined in a file on the node should be
used.
                    The profile must be preconfigured on the node to work.
                    Must be a descending path, relative to the kubelet's configured seccomp
profile location.
                    Must only be set if type is "Localhost".
                  type: string
                type:
                  description: |-
                    type indicates which kind of seccomp profile will be applied.
                    Valid options are:


                    Localhost - a profile defined in a file on the node should be used.
                    RuntimeDefault - the container runtime default profile should be used.
                    Unconfined - no profile should be applied.
                  type: string
              required:
              - type
              type: object
            windowsOptions:
              description: |-
                The Windows specific settings applied to all containers.
                If unspecified, the options from the PodSecurityContext will be used.
                If set in both SecurityContext and PodSecurityContext, the value specified in
SecurityContext takes precedence.
                Note that this field cannot be set when spec.os.name is linux.
              properties:
                gmsaCredentialSpec:
                  description: |-
                    GMSACredentialSpec is where the GMSA admission webhook
                    (https://github.com/kubernetes-sigs/windows-gmsa) inlines the contents of
the
                    GMSA credential spec named by the GMSACredentialSpecName field.
                  type: string
                gmsaCredentialSpecName:
                  description: GMSACredentialSpecName
                    is the name of the GMSA credential
                    spec to use.
                  type: string
                hostProcess:
                  description: |-
                    HostProcess determines if a container should be run as a 'Host Process'
container.
```

```
                        This field is alpha-level and will only be honored by components that
enable the
                        WindowsHostProcessContainers feature flag. Setting this field without the
feature
                        flag will result in errors when validating the Pod. All of a Pod's containers
must
                        have the same effective HostProcess value (it is not allowed to have a mix
of HostProcess
                        containers and non-HostProcess containers).  In addition, if HostProcess is
true
                        then HostNetwork must also be set to true.
                      type: boolean
                    runAsUserName:
                      description: |-
                        The UserName in Windows to run the entrypoint of the container process.
                        Defaults to the user specified in image metadata if unspecified.
                        May also be set in PodSecurityContext. If set in both SecurityContext and
                        PodSecurityContext, the value specified in SecurityContext takes
precedence.
                      type: string
                  type: object
                type: object
              startupProbe:
                description: |-
                  StartupProbe indicates that the Pod has successfully initialized.
                  If specified, no other probes are executed until this completes successfully.
                  If this probe fails, the Pod will be restarted, just as if the livenessProbe failed.
                  This can be used to provide different probe parameters at the beginning of a
Pod's lifecycle,
                  when it might take a long time to load data or warm a cache, than during steady-
state operation.
                  This cannot be updated.
                  More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                properties:
                  exec:
                    description: Exec specifies the action to
                      take.
                    properties:
                      command:
                        description: |-
                          Command is the command line to execute inside the container, the working
directory for the
                          command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                          not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                          a shell, you need to explicitly call out to that shell.
                          Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                        items:
                          type: string
                        type: array
                    type: object
                  failureThreshold:
                    description: |-
```

```
      Minimum consecutive failures for the probe to be considered failed after
having succeeded.
      Defaults to 3. Minimum value is 1.
    format: int32
    type: integer
  grpc:
    description: |-
      GRPC specifies an action involving a GRPC port.
      This is a beta field and requires enabling GRPCContainerProbe feature gate.
    properties:
      port:
        description: Port number of the gRPC
          service. Number must be in the range
          1 to 65535.
        format: int32
        type: integer
      service:
        description: |-
          Service is the name of the service to place in the gRPC
HealthCheckRequest
          (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


          If this is not specified, the default behavior is defined by gRPC.
        type: string
    required:
    - port
    type: object
  httpGet:
    description: HTTPGet specifies the http
      request to perform.
    properties:
      host:
        description: |-
          Host name to connect to, defaults to the pod IP. You probably want to set
          "Host" in httpHeaders instead.
        type: string
      httpHeaders:
        description: Custom headers to set in
          the request. HTTP allows repeated
          headers.
        items:
          description: HTTPHeader describes
            a custom header to be used in HTTP
            probes
          properties:
            name:
              description: The header field
                name
              type: string
            value:
              description: The header field
                value
              type: string
          required:
          - name
```

```
                - value
                type: object
              type: array
            path:
              description: Path to access on the HTTP
                server.
              type: string
            port:
              anyOf:
              - type: integer
              - type: string
              description: |-
                Name or number of the port to access on the container.
                Number must be in the range 1 to 65535.
                Name must be an IANA_SVC_NAME.
              x-kubernetes-int-or-string: true
            scheme:
              description: |-
                Scheme to use for connecting to the host.
                Defaults to HTTP.
              type: string
          required:
          - port
          type: object
        initialDelaySeconds:
          description: |-
            Number of seconds after the container has started before liveness probes are
initiated.
            More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
          format: int32
          type: integer
        periodSeconds:
          description: |-
            How often (in seconds) to perform the probe.
            Default to 10 seconds. Minimum value is 1.
          format: int32
          type: integer
        successThreshold:
          description: |-
            Minimum consecutive successes for the probe to be considered successful
after having failed.
            Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
          format: int32
          type: integer
        tcpSocket:
          description: TCPSocket specifies an action
            involving a TCP port.
          properties:
            host:
              description: 'Optional: Host name to
                connect to, defaults to the pod IP.'
              type: string
            port:
              anyOf:
              - type: integer
```

```
                - type: string
                description: |-
                  Number or name of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
              required:
              - port
              type: object
            terminationGracePeriodSeconds:
              description: |-
                Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
                The grace period is the duration in seconds after the processes running in the
pod are sent
                a termination signal and the time when the processes are forcibly halted with
a kill signal.
                Set this value longer than the expected cleanup time for your process.
                If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
                value overrides the value provided by the pod spec.
                Value must be non-negative integer. The value zero indicates stop
immediately via
                the kill signal (no opportunity to shut down).
                This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
                Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
              format: int64
              type: integer
            timeoutSeconds:
              description: |-
                Number of seconds after which the probe times out.
                Defaults to 1 second. Minimum value is 1.
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
              format: int32
              type: integer
          type: object
        stdin:
          description: |-
            Whether this container should allocate a buffer for stdin in the container runtime.
If this
            is not set, reads from stdin in the container will always result in EOF.
            Default is false.
          type: boolean
        stdinOnce:
          description: |-
            Whether the container runtime should close the stdin channel after it has been
opened by
            a single attach. When stdin is true the stdin stream will remain open across
multiple attach
            sessions. If stdinOnce is set to true, stdin is opened on container start, is empty
until the
            first client attaches to stdin, and then remains open and accepts data until the
client disconnects,
```

```
                at which time stdin is closed and remains closed until the container is restarted.
If this
                flag is false, a container processes that reads from stdin will never receive an
EOF.
                Default is false
              type: boolean
            terminationMessagePath:
              description: |-
                Optional: Path at which the file to which the container's termination message
                will be written is mounted into the container's filesystem.
                Message written is intended to be brief final status, such as an assertion failure
message.
                Will be truncated by the node if greater than 4096 bytes. The total message
length across
                all containers will be limited to 12kb.
                Defaults to /dev/termination-log.
                Cannot be updated.
              type: string
            terminationMessagePolicy:
              description: |-
                Indicate how the termination message should be populated. File will use the
contents of
                terminationMessagePath to populate the container status message on both
success and failure.
                FallbackToLogsOnError will use the last chunk of container log output if the
termination
                message file is empty and the container exited with an error.
                The log output is limited to 2048 bytes or 80 lines, whichever is smaller.
                Defaults to File.
                Cannot be updated.
              type: string
            tty:
              description: |-
                Whether this container should allocate a TTY for itself, also requires 'stdin' to be
true.
                Default is false.
              type: boolean
            volumeDevices:
              description: volumeDevices is the list of block
                devices to be used by the container.
              items:
                description: volumeDevice describes a mapping
                  of a raw block device within a container.
                properties:
                  devicePath:
                    description: devicePath is the path inside
                      of the container that the device will
                      be mapped to.
                    type: string
                  name:
                    description: name must match the name
                      of a persistentVolumeClaim in the pod
                    type: string
                required:
                - devicePath
                - name
```

```
                type: object
              type: array
            volumeMounts:
              description: |-
                Pod volumes to mount into the container's filesystem.
                Cannot be updated.
              items:
                description: VolumeMount describes a mounting
                  of a Volume within a container.
                properties:
                  mountPath:
                    description: |-
                      Path within the container at which the volume should be mounted.  Must
                      not contain ':'.
                    type: string
                  mountPropagation:
                    description: |-
                      mountPropagation determines how mounts are propagated from the host
                      to container and the other way around.
                      When not set, MountPropagationNone is used.
                      This field is beta in 1.10.
                    type: string
                  name:
                    description: This must match the Name
                      of a Volume.
                    type: string
                  readOnly:
                    description: |-
                      Mounted read-only if true, read-write otherwise (false or unspecified).
                      Defaults to false.
                    type: boolean
                  subPath:
                    description: |-
                      Path within the volume from which the container's volume should be
mounted.
                      Defaults to "" (volume's root).
                    type: string
                  subPathExpr:
                    description: |-
                      Expanded path within the volume from which the container's volume should
be mounted.
                      Behaves similarly to SubPath but environment variable references
$(VAR_NAME) are expanded using the container's environment.
                      Defaults to "" (volume's root).
                      SubPathExpr and SubPath are mutually exclusive.
                    type: string
                required:
                - mountPath
                - name
                type: object
              type: array
            workingDir:
              description: |-
                Container's working directory.
                If not specified, the container runtime's default will be used, which
                might be configured in the container image.
```

```
                Cannot be updated.
              type: string
            required:
            - name
            type: object
          type: array
        dnsConfig:
          description: |-
            Specifies the DNS parameters of a pod.
            Parameters specified here will be merged to the generated DNS
            configuration based on DNSPolicy.
          properties:
            nameservers:
              description: |-
                A list of DNS name server IP addresses.
                This will be appended to the base nameservers generated from DNSPolicy.
                Duplicated nameservers will be removed.
              items:
                type: string
              type: array
            options:
              description: |-
                A list of DNS resolver options.
                This will be merged with the base options generated from DNSPolicy.
                Duplicated entries will be removed. Resolution options given in Options
                will override those that appear in the base DNSPolicy.
              items:
                description: PodDNSConfigOption defines DNS
                  resolver options of a pod.
                properties:
                  name:
                    description: Required.
                    type: string
                  value:
                    type: string
                type: object
              type: array
            searches:
              description: |-
                A list of DNS search domains for host-name lookup.
                This will be appended to the base search paths generated from DNSPolicy.
                Duplicated search paths will be removed.
              items:
                type: string
              type: array
          type: object
        dnsPolicy:
          description: |-
            Set DNS policy for the pod.
            Defaults to "ClusterFirst".
            Valid values are 'ClusterFirstWithHostNet', 'ClusterFirst', 'Default' or 'None'.
            DNS parameters given in DNSConfig will be merged with the policy selected with
DNSPolicy.
            To have DNS options set along with hostNetwork, you have to specify DNS policy
            explicitly to 'ClusterFirstWithHostNet'.
          type: string
```

```
                    enableServiceLinks:
                      description: |-
                        EnableServiceLinks indicates whether information about services should be injected
into pod's
                        environment variables, matching the syntax of Docker links.
                        Optional: Defaults to true.
                      type: boolean
                    ephemeralContainers:
                      description: |-
                        List of ephemeral containers run in this pod. Ephemeral containers may be run in an
existing
                        pod to perform user-initiated actions such as debugging. This list cannot be
specified when
                        creating a pod, and it cannot be modified by updating the pod spec. In order to add
an
                        ephemeral container to an existing pod, use the pod's ephemeralcontainers
subresource.
                        This field is beta-level and available on clusters that haven't disabled the
EphemeralContainers feature gate.
                      items:
                        description: |-
                          An EphemeralContainer is a temporary container that you may add to an existing
Pod for
                          user-initiated activities such as debugging. Ephemeral containers have no
resource or
                          scheduling guarantees, and they will not be restarted when they exit or when a
Pod is
                          removed or restarted. The kubelet may evict a Pod if an ephemeral container
causes the
                          Pod to exceed its resource allocation.


                          To add an ephemeral container, use the ephemeralcontainers subresource of an
existing
                          Pod. Ephemeral containers may not be removed or restarted.


                          This is a beta feature available on clusters that haven't disabled the
EphemeralContainers feature gate.
                        properties:
                          args:
                            description: |-
                              Arguments to the entrypoint.
                              The image's CMD is used if this is not provided.
                              Variable references $(VAR_NAME) are expanded using the container's
environment. If a variable
                              cannot be resolved, the reference in the input string will be unchanged. Double
$$ are reduced
                              to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
"$$(VAR_NAME)" will
                              produce the string literal "$(VAR_NAME)". Escaped references will never be
expanded, regardless
                              of whether the variable exists or not. Cannot be updated.
                              More info: https://kubernetes.io/docs/tasks/inject-data-application/define-
command-argument-container/#running-a-command-in-a-shell
                            items:
```

```
                type: string
              type: array
            command:
              description: |-
                Entrypoint array. Not executed within a shell.
                The image's ENTRYPOINT is used if this is not provided.
                Variable references $(VAR_NAME) are expanded using the container's
environment. If a variable
                cannot be resolved, the reference in the input string will be unchanged. Double
$$ are reduced
                to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
"$$(VAR_NAME)" will
                produce the string literal "$(VAR_NAME)". Escaped references will never be
expanded, regardless
                of whether the variable exists or not. Cannot be updated.
                More info: https://kubernetes.io/docs/tasks/inject-data-application/define-
command-argument-container/#running-a-command-in-a-shell
              items:
                type: string
              type: array
            env:
              description: |-
                List of environment variables to set in the container.
                Cannot be updated.
              items:
                description: EnvVar represents an environment
                  variable present in a Container.
                properties:
                  name:
                    description: Name of the environment variable.
                      Must be a C_IDENTIFIER.
                    type: string
                  value:
                    description: |-
                      Variable references $(VAR_NAME) are expanded
                      using the previously defined environment variables in the container and
                      any service environment variables. If a variable cannot be resolved,
                      the reference in the input string will be unchanged. Double $$ are reduced
                      to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
                      "$$(VAR_NAME)" will produce the string literal "$(VAR_NAME)".
                      Escaped references will never be expanded, regardless of whether the
variable
                      exists or not.
                      Defaults to "".
                    type: string
                  valueFrom:
                    description: Source for the environment
                      variable's value. Cannot be used if
                      value is not empty.
                    properties:
                      configMapKeyRef:
                        description: Selects a key of a ConfigMap.
                        properties:
                          key:
                            description: The key to select.
                            type: string
```

```
                                    name:
                                      description: |-
                                        Name of the referent.
                                        More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                        TODO: Add other useful fields. apiVersion, kind, uid?
                                      type: string
                                    optional:
                                      description: Specify whether the
                                        ConfigMap or its key must be
                                        defined
                                      type: boolean
                                  required:
                                  - key
                                  type: object
                                  x-kubernetes-map-type: atomic
                                fieldRef:
                                  description: |-
                                    Selects a field of the pod: supports metadata.name,
metadata.namespace, `metadata.labels['<KEY>']`, `metadata.annotations['<KEY>']`,
                                    spec.nodeName, spec.serviceAccountName, status.hostIP, status.podIP,
status.podIPs.
                                  properties:
                                    apiVersion:
                                      description: Version of the schema
                                        the FieldPath is written in
                                        terms of, defaults to "v1".
                                      type: string
                                    fieldPath:
                                      description: Path of the field
                                        to select in the specified API
                                        version.
                                      type: string
                                  required:
                                  - fieldPath
                                  type: object
                                  x-kubernetes-map-type: atomic
                                resourceFieldRef:
                                  description: |-
                                    Selects a resource of the container: only resources limits and requests
                                    (limits.cpu, limits.memory, limits.ephemeral-storage, requests.cpu,
requests.memory and requests.ephemeral-storage) are currently supported.
                                  properties:
                                    containerName:
                                      description: 'Container name:
                                        required for volumes, optional
                                        for env vars'
                                      type: string
                                    divisor:
                                      anyOf:
                                      - type: integer
                                      - type: string
                                      description: Specifies the output
                                        format of the exposed resources,
                                        defaults to "1"
```

```
                                    pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                    x-kubernetes-int-or-string: true
                                  resource:
                                    description: 'Required: resource
                                      to select'
                                    type: string
                                  required:
                                  - resource
                                  type: object
                                  x-kubernetes-map-type: atomic
                                secretKeyRef:
                                  description: Selects a key of a secret
                                    in the pod's namespace
                                  properties:
                                    key:
                                      description: The key of the secret
                                        to select from.  Must be a valid
                                        secret key.
                                      type: string
                                    name:
                                      description: |-
                                        Name of the referent.
                                        More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                        TODO: Add other useful fields. apiVersion, kind, uid?
                                      type: string
                                    optional:
                                      description: Specify whether the
                                        Secret or its key must be defined
                                      type: boolean
                                  required:
                                  - key
                                  type: object
                                  x-kubernetes-map-type: atomic
                              type: object
                            required:
                            - name
                            type: object
                          type: array
                        envFrom:
                          description: |-
                            List of sources to populate environment variables in the container.
                            The keys defined within a source must be a C_IDENTIFIER. All invalid keys
                            will be reported as an event when the container is starting. When a key exists in
multiple
                            sources, the value associated with the last source will take precedence.
                            Values defined by an Env with a duplicate key will take precedence.
                            Cannot be updated.
                          items:
                            description: EnvFromSource represents the
                              source of a set of ConfigMaps
                            properties:
                              configMapRef:
                                description: The ConfigMap to select from
                                properties:
```

```
                                name:
                                  description: |-
                                    Name of the referent.
                                    More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                    TODO: Add other useful fields. apiVersion, kind, uid?
                                  type: string
                                optional:
                                  description: Specify whether the ConfigMap
                                    must be defined
                                  type: boolean
                              type: object
                              x-kubernetes-map-type: atomic
                            prefix:
                              description: An optional identifier to
                                prepend to each key in the ConfigMap.
                                Must be a C_IDENTIFIER.
                              type: string
                            secretRef:
                              description: The Secret to select from
                              properties:
                                name:
                                  description: |-
                                    Name of the referent.
                                    More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                    TODO: Add other useful fields. apiVersion, kind, uid?
                                  type: string
                                optional:
                                  description: Specify whether the Secret
                                    must be defined
                                  type: boolean
                              type: object
                              x-kubernetes-map-type: atomic
                          type: object
                        type: array
                      image:
                        description: |-
                          Container image name.
                          More info: https://kubernetes.io/docs/concepts/containers/images
                        type: string
                      imagePullPolicy:
                        description: |-
                          Image pull policy.
                          One of Always, Never, IfNotPresent.
                          Defaults to Always if :latest tag is specified, or IfNotPresent otherwise.
                          Cannot be updated.
                          More info: https://kubernetes.io/docs/concepts/containers/images#updating-
images
                        type: string
                      lifecycle:
                        description: Lifecycle is not allowed for ephemeral
                          containers.
                        properties:
                          postStart:
                            description: |-
```

```
                PostStart is called immediately after a container is created. If the handler fails,
                the container is terminated and restarted according to its restart policy.
                Other management of the container blocks until the hook completes.
                More info: https://kubernetes.io/docs/concepts/containers/container-lifecycle-
hooks/#container-hooks
              properties:
                exec:
                  description: Exec specifies the action
                    to take.
                  properties:
                    command:
                      description: |-
                        Command is the command line to execute inside the container, the
working directory for the
                        command  is root ('/') in the container's filesystem. The command is
simply exec'd, it is
                        not run inside a shell, so traditional shell instructions ('|', etc) won't work.
To use
                        a shell, you need to explicitly call out to that shell.
                        Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                      items:
                        type: string
                      type: array
                  type: object
                httpGet:
                  description: HTTPGet specifies the http
                    request to perform.
                  properties:
                    host:
                      description: |-
                        Host name to connect to, defaults to the pod IP. You probably want to
set
                        "Host" in httpHeaders instead.
                      type: string
                    httpHeaders:
                      description: Custom headers to set
                        in the request. HTTP allows repeated
                        headers.
                      items:
                        description: HTTPHeader describes
                          a custom header to be used in
                          HTTP probes
                        properties:
                          name:
                            description: The header field
                              name
                            type: string
                          value:
                            description: The header field
                              value
                            type: string
                        required:
                        - name
                        - value
                        type: object
                      type: array
```

```
                path:
                  description: Path to access on the
                    HTTP server.
                  type: string
                port:
                  anyOf:
                  - type: integer
                  - type: string
                  description: |-
                    Name or number of the port to access on the container.
                    Number must be in the range 1 to 65535.
                    Name must be an IANA_SVC_NAME.
                  x-kubernetes-int-or-string: true
                scheme:
                  description: |-
                    Scheme to use for connecting to the host.
                    Defaults to HTTP.
                  type: string
              required:
              - port
              type: object
            tcpSocket:
              description: |-
                Deprecated. TCPSocket is NOT supported as a LifecycleHandler and kept
                for the backward compatibility. There are no validation of this field and
                lifecycle hooks will fail in runtime when tcp handler is specified.
              properties:
                host:
                  description: 'Optional: Host name
                    to connect to, defaults to the
                    pod IP.'
                  type: string
                port:
                  anyOf:
                  - type: integer
                  - type: string
                  description: |-
                    Number or name of the port to access on the container.
                    Number must be in the range 1 to 65535.
                    Name must be an IANA_SVC_NAME.
                  x-kubernetes-int-or-string: true
              required:
              - port
              type: object
          type: object
        preStop:
          description: |-
            PreStop is called immediately before a container is terminated due to an
            API request or management event such as liveness/startup probe failure,
            preemption, resource contention, etc. The handler is not called if the
            container crashes or exits. The Pod's termination grace period countdown
begins before the
            PreStop hook is executed. Regardless of the outcome of the handler, the
            container will eventually terminate within the Pod's termination grace
            period (unless delayed by finalizers). Other management of the container
blocks until the hook completes
```

```
                or until the termination grace period is reached.
                More info: https://kubernetes.io/docs/concepts/containers/container-lifecycle-
hooks/#container-hooks
              properties:
                exec:
                  description: Exec specifies the action
                    to take.
                  properties:
                    command:
                      description: |-
                        Command is the command line to execute inside the container, the
working directory for the
                        command  is root ('/') in the container's filesystem. The command is
simply exec'd, it is
                        not run inside a shell, so traditional shell instructions ('|', etc) won't work.
To use
                        a shell, you need to explicitly call out to that shell.
                        Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                      items:
                        type: string
                      type: array
                  type: object
                httpGet:
                  description: HTTPGet specifies the http
                    request to perform.
                  properties:
                    host:
                      description: |-
                        Host name to connect to, defaults to the pod IP. You probably want to
set
                        "Host" in httpHeaders instead.
                      type: string
                    httpHeaders:
                      description: Custom headers to set
                        in the request. HTTP allows repeated
                        headers.
                      items:
                        description: HTTPHeader describes
                          a custom header to be used in
                          HTTP probes
                        properties:
                          name:
                            description: The header field
                              name
                            type: string
                          value:
                            description: The header field
                              value
                            type: string
                        required:
                        - name
                        - value
                        type: object
                      type: array
                    path:
                      description: Path to access on the
```

```
                HTTP server.
              type: string
            port:
              anyOf:
              - type: integer
              - type: string
              description: |-
                Name or number of the port to access on the container.
                Number must be in the range 1 to 65535.
                Name must be an IANA_SVC_NAME.
              x-kubernetes-int-or-string: true
            scheme:
              description: |-
                Scheme to use for connecting to the host.
                Defaults to HTTP.
              type: string
          required:
          - port
          type: object
        tcpSocket:
          description: |-
            Deprecated. TCPSocket is NOT supported as a LifecycleHandler and kept
            for the backward compatibility. There are no validation of this field and
            lifecycle hooks will fail in runtime when tcp handler is specified.
          properties:
            host:
              description: 'Optional: Host name
                to connect to, defaults to the
                pod IP.'
              type: string
            port:
              anyOf:
              - type: integer
              - type: string
              description: |-
                Number or name of the port to access on the container.
                Number must be in the range 1 to 65535.
                Name must be an IANA_SVC_NAME.
              x-kubernetes-int-or-string: true
          required:
          - port
          type: object
      type: object
    type: object
  livenessProbe:
    description: Probes are not allowed for ephemeral
      containers.
    properties:
      exec:
        description: Exec specifies the action to
          take.
        properties:
          command:
            description: |-
              Command is the command line to execute inside the container, the working
directory for the
```

```
                command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                a shell, you need to explicitly call out to that shell.
                Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
              items:
                type: string
              type: array
            type: object
          failureThreshold:
            description: |-
              Minimum consecutive failures for the probe to be considered failed after
having succeeded.
              Defaults to 3. Minimum value is 1.
            format: int32
            type: integer
          grpc:
            description: |-
              GRPC specifies an action involving a GRPC port.
              This is a beta field and requires enabling GRPCContainerProbe feature gate.
            properties:
              port:
                description: Port number of the gRPC
                  service. Number must be in the range
                  1 to 65535.
                format: int32
                type: integer
              service:
                description: |-
                  Service is the name of the service to place in the gRPC
HealthCheckRequest
                  (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


                  If this is not specified, the default behavior is defined by gRPC.
                type: string
            required:
            - port
            type: object
          httpGet:
            description: HTTPGet specifies the http
              request to perform.
            properties:
              host:
                description: |-
                  Host name to connect to, defaults to the pod IP. You probably want to set
                  "Host" in httpHeaders instead.
                type: string
              httpHeaders:
                description: Custom headers to set in
                  the request. HTTP allows repeated
                  headers.
                items:
                  description: HTTPHeader describes
                    a custom header to be used in HTTP
```

```
                probes
              properties:
                name:
                  description: The header field
                    name
                  type: string
                value:
                  description: The header field
                    value
                  type: string
              required:
              - name
              - value
              type: object
            type: array
          path:
            description: Path to access on the HTTP
              server.
            type: string
          port:
            anyOf:
            - type: integer
            - type: string
            description: |-
              Name or number of the port to access on the container.
              Number must be in the range 1 to 65535.
              Name must be an IANA_SVC_NAME.
            x-kubernetes-int-or-string: true
          scheme:
            description: |-
              Scheme to use for connecting to the host.
              Defaults to HTTP.
            type: string
        required:
        - port
        type: object
      initialDelaySeconds:
        description: |-
          Number of seconds after the container has started before liveness probes are
initiated.
          More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
        format: int32
        type: integer
      periodSeconds:
        description: |-
          How often (in seconds) to perform the probe.
          Default to 10 seconds. Minimum value is 1.
        format: int32
        type: integer
      successThreshold:
        description: |-
          Minimum consecutive successes for the probe to be considered successful
after having failed.
          Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
        format: int32
```

```
                type: integer
              tcpSocket:
                description: TCPSocket specifies an action
                  involving a TCP port.
                properties:
                  host:
                    description: 'Optional: Host name to
                      connect to, defaults to the pod IP.'
                    type: string
                  port:
                    anyOf:
                    - type: integer
                    - type: string
                    description: |-
                      Number or name of the port to access on the container.
                      Number must be in the range 1 to 65535.
                      Name must be an IANA_SVC_NAME.
                    x-kubernetes-int-or-string: true
                required:
                - port
                type: object
              terminationGracePeriodSeconds:
                description: |-
                  Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
                  The grace period is the duration in seconds after the processes running in the
pod are sent
                  a termination signal and the time when the processes are forcibly halted with
a kill signal.
                  Set this value longer than the expected cleanup time for your process.
                  If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
                  value overrides the value provided by the pod spec.
                  Value must be non-negative integer. The value zero indicates stop
immediately via
                  the kill signal (no opportunity to shut down).
                  This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
                  Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
                format: int64
                type: integer
              timeoutSeconds:
                description: |-
                  Number of seconds after which the probe times out.
                  Defaults to 1 second. Minimum value is 1.
                  More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                format: int32
                type: integer
            type: object
          name:
            description: |-
              Name of the ephemeral container specified as a DNS_LABEL.
              This name must be unique among all containers, init containers and ephemeral
containers.
            type: string
```

```
                    ports:
                      description: Ports are not allowed for ephemeral
                        containers.
                      items:
                        description: ContainerPort represents a network
                          port in a single container.
                        properties:
                          containerPort:
                            description: |-
                              Number of port to expose on the pod's IP address.
                              This must be a valid port number, 0 < x < 65536.
                            format: int32
                            type: integer
                          hostIP:
                            description: What host IP to bind the
                              external port to.
                            type: string
                          hostPort:
                            description: |-
                              Number of port to expose on the host.
                              If specified, this must be a valid port number, 0 < x < 65536.
                              If HostNetwork is specified, this must match ContainerPort.
                              Most containers do not need this.
                            format: int32
                            type: integer
                          name:
                            description: |-
                              If specified, this must be an IANA_SVC_NAME and unique within the pod.
Each
                              named port in a pod must have a unique name. Name for the port that can
be
                              referred to by services.
                            type: string
                          protocol:
                            default: TCP
                            description: |-
                              Protocol for port. Must be UDP, TCP, or SCTP.
                              Defaults to "TCP".
                            type: string
                        required:
                        - containerPort
                        type: object
                      type: array
                      x-kubernetes-list-map-keys:
                      - containerPort
                      - protocol
                      x-kubernetes-list-type: map
                    readinessProbe:
                      description: Probes are not allowed for ephemeral
                        containers.
                      properties:
                        exec:
                          description: Exec specifies the action to
                            take.
                          properties:
                            command:
```

```
                description: |-
                  Command is the command line to execute inside the container, the working
directory for the
                  command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                  not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                  a shell, you need to explicitly call out to that shell.
                  Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                items:
                  type: string
                type: array
              type: object
            failureThreshold:
              description: |-
                Minimum consecutive failures for the probe to be considered failed after
having succeeded.
                Defaults to 3. Minimum value is 1.
              format: int32
              type: integer
            grpc:
              description: |-
                GRPC specifies an action involving a GRPC port.
                This is a beta field and requires enabling GRPCContainerProbe feature gate.
              properties:
                port:
                  description: Port number of the gRPC
                    service. Number must be in the range
                    1 to 65535.
                  format: int32
                  type: integer
                service:
                  description: |-
                    Service is the name of the service to place in the gRPC
HealthCheckRequest
                    (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


                    If this is not specified, the default behavior is defined by gRPC.
                  type: string
              required:
              - port
              type: object
            httpGet:
              description: HTTPGet specifies the http
                request to perform.
              properties:
                host:
                  description: |-
                    Host name to connect to, defaults to the pod IP. You probably want to set
                    "Host" in httpHeaders instead.
                  type: string
                httpHeaders:
                  description: Custom headers to set in
                    the request. HTTP allows repeated
                    headers.
```

```
                items:
                  description: HTTPHeader describes
                    a custom header to be used in HTTP
                    probes
                  properties:
                    name:
                      description: The header field
                        name
                      type: string
                    value:
                      description: The header field
                        value
                      type: string
                  required:
                  - name
                  - value
                  type: object
                type: array
              path:
                description: Path to access on the HTTP
                  server.
                type: string
              port:
                anyOf:
                - type: integer
                - type: string
                description: |-
                  Name or number of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
              scheme:
                description: |-
                  Scheme to use for connecting to the host.
                  Defaults to HTTP.
                type: string
            required:
            - port
            type: object
          initialDelaySeconds:
            description: |-
              Number of seconds after the container has started before liveness probes are
initiated.
              More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
            format: int32
            type: integer
          periodSeconds:
            description: |-
              How often (in seconds) to perform the probe.
              Default to 10 seconds. Minimum value is 1.
            format: int32
            type: integer
          successThreshold:
            description: |-
```

```
                Minimum consecutive successes for the probe to be considered successful
after having failed.
                Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
              format: int32
              type: integer
            tcpSocket:
              description: TCPSocket specifies an action
                involving a TCP port.
              properties:
                host:
                  description: 'Optional: Host name to
                    connect to, defaults to the pod IP.'
                  type: string
                port:
                  anyOf:
                  - type: integer
                  - type: string
                  description: |-
                    Number or name of the port to access on the container.
                    Number must be in the range 1 to 65535.
                    Name must be an IANA_SVC_NAME.
                  x-kubernetes-int-or-string: true
              required:
              - port
              type: object
            terminationGracePeriodSeconds:
              description: |-
                Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
                The grace period is the duration in seconds after the processes running in the
pod are sent
                a termination signal and the time when the processes are forcibly halted with
a kill signal.
                Set this value longer than the expected cleanup time for your process.
                If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
                value overrides the value provided by the pod spec.
                Value must be non-negative integer. The value zero indicates stop
immediately via
                the kill signal (no opportunity to shut down).
                This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
                Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
              format: int64
              type: integer
            timeoutSeconds:
              description: |-
                Number of seconds after which the probe times out.
                Defaults to 1 second. Minimum value is 1.
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
              format: int32
              type: integer
          type: object
        resources:
          description: |-
```

```
                  Resources are not allowed for ephemeral containers. Ephemeral containers use
spare resources
                  already allocated to the pod.
                properties:
                  limits:
                    additionalProperties:
                      anyOf:
                      - type: integer
                      - type: string
                      pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                      x-kubernetes-int-or-string: true
                    description: |-
                      Limits describes the maximum amount of compute resources allowed.
                      More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                    type: object
                  requests:
                    additionalProperties:
                      anyOf:
                      - type: integer
                      - type: string
                      pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                      x-kubernetes-int-or-string: true
                    description: |-
                      Requests describes the minimum amount of compute resources required.
                      If Requests is omitted for a container, it defaults to Limits if that is explicitly
specified,
                      otherwise to an implementation-defined value.
                      More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                    type: object
                type: object
              securityContext:
                description: |-
                  Optional: SecurityContext defines the security options the ephemeral container
should be run with.
                  If set, the fields of SecurityContext override the equivalent fields of
PodSecurityContext.
                properties:
                  allowPrivilegeEscalation:
                    description: |-
                      AllowPrivilegeEscalation controls whether a process can gain more
                      privileges than its parent process. This bool directly controls if
                      the no_new_privs flag will be set on the container process.
                      AllowPrivilegeEscalation is true always when the container is:
                      1) run as Privileged
                      2) has CAP_SYS_ADMIN
                      Note that this field cannot be set when spec.os.name is windows.
                    type: boolean
                  capabilities:
                    description: |-
                      The capabilities to add/drop when running containers.
                      Defaults to the default set of capabilities granted by the container runtime.
                      Note that this field cannot be set when spec.os.name is windows.
```

```
      properties:
        add:
          description: Added capabilities
          items:
            description: Capability represent
              POSIX capabilities type
            type: string
          type: array
        drop:
          description: Removed capabilities
          items:
            description: Capability represent
              POSIX capabilities type
            type: string
          type: array
      type: object
    privileged:
      description: |-
        Run container in privileged mode.
        Processes in privileged containers are essentially equivalent to root on the
host.
        Defaults to false.
        Note that this field cannot be set when spec.os.name is windows.
      type: boolean
    procMount:
      description: |-
        procMount denotes the type of proc mount to use for the containers.
        The default is DefaultProcMount which uses the container runtime defaults for
        readonly paths and masked paths.
        This requires the ProcMountType feature flag to be enabled.
        Note that this field cannot be set when spec.os.name is windows.
      type: string
    readOnlyRootFilesystem:
      description: |-
        Whether this container has a read-only root filesystem.
        Default is false.
        Note that this field cannot be set when spec.os.name is windows.
      type: boolean
    runAsGroup:
      description: |-
        The GID to run the entrypoint of the container process.
        Uses runtime default if unset.
        May also be set in PodSecurityContext.  If set in both SecurityContext and
        PodSecurityContext, the value specified in SecurityContext takes precedence.
        Note that this field cannot be set when spec.os.name is windows.
      format: int64
      type: integer
    runAsNonRoot:
      description: |-
        Indicates that the container must run as a non-root user.
        If true, the Kubelet will validate the image at runtime to ensure that it
        does not run as UID 0 (root) and fail to start the container if it does.
        If unset or false, no such validation will be performed.
        May also be set in PodSecurityContext.  If set in both SecurityContext and
        PodSecurityContext, the value specified in SecurityContext takes precedence.
      type: boolean
```

```
                    runAsUser:
                      description: |-
                        The UID to run the entrypoint of the container process.
                        Defaults to user specified in image metadata if unspecified.
                        May also be set in PodSecurityContext.  If set in both SecurityContext and
                        PodSecurityContext, the value specified in SecurityContext takes precedence.
                        Note that this field cannot be set when spec.os.name is windows.
                      format: int64
                      type: integer
                    seLinuxOptions:
                      description: |-
                        The SELinux context to be applied to the container.
                        If unspecified, the container runtime will allocate a random SELinux context
for each
                        container.  May also be set in PodSecurityContext.  If set in both
SecurityContext and
                        PodSecurityContext, the value specified in SecurityContext takes precedence.
                        Note that this field cannot be set when spec.os.name is windows.
                      properties:
                        level:
                          description: Level is SELinux level
                            label that applies to the container.
                          type: string
                        role:
                          description: Role is a SELinux role
                            label that applies to the container.
                          type: string
                        type:
                          description: Type is a SELinux type
                            label that applies to the container.
                          type: string
                        user:
                          description: User is a SELinux user
                            label that applies to the container.
                          type: string
                      type: object
                    seccompProfile:
                      description: |-
                        The seccomp options to use by this container. If seccomp options are
                        provided at both the pod & container level, the container options
                        override the pod options.
                        Note that this field cannot be set when spec.os.name is windows.
                      properties:
                        localhostProfile:
                          description: |-
                            localhostProfile indicates a profile defined in a file on the node should be
used.
                            The profile must be preconfigured on the node to work.
                            Must be a descending path, relative to the kubelet's configured seccomp
profile location.
                            Must only be set if type is "Localhost".
                          type: string
                        type:
                          description: |-
                            type indicates which kind of seccomp profile will be applied.
                            Valid options are:
```

```
                                    Localhost - a profile defined in a file on the node should be used.
                                    RuntimeDefault - the container runtime default profile should be used.
                                    Unconfined - no profile should be applied.
                                  type: string
                                required:
                                - type
                                type: object
                              windowsOptions:
                                description: |-
                                  The Windows specific settings applied to all containers.
                                  If unspecified, the options from the PodSecurityContext will be used.
                                  If set in both SecurityContext and PodSecurityContext, the value specified in
SecurityContext takes precedence.
                                  Note that this field cannot be set when spec.os.name is linux.
                                properties:
                                  gmsaCredentialSpec:
                                    description: |-
                                      GMSACredentialSpec is where the GMSA admission webhook
                                      (https://github.com/kubernetes-sigs/windows-gmsa) inlines the contents of
the
                                      GMSA credential spec named by the GMSACredentialSpecName field.
                                    type: string
                                  gmsaCredentialSpecName:
                                    description: GMSACredentialSpecName
                                      is the name of the GMSA credential
                                      spec to use.
                                    type: string
                                  hostProcess:
                                    description: |-
                                      HostProcess determines if a container should be run as a 'Host Process'
container.
                                      This field is alpha-level and will only be honored by components that
enable the
                                      WindowsHostProcessContainers feature flag. Setting this field without the
feature
                                      flag will result in errors when validating the Pod. All of a Pod's containers
must
                                      have the same effective HostProcess value (it is not allowed to have a mix
of HostProcess
                                      containers and non-HostProcess containers).  In addition, if HostProcess is
true
                                      then HostNetwork must also be set to true.
                                    type: boolean
                                  runAsUserName:
                                    description: |-
                                      The UserName in Windows to run the entrypoint of the container process.
                                      Defaults to the user specified in image metadata if unspecified.
                                      May also be set in PodSecurityContext. If set in both SecurityContext and
                                      PodSecurityContext, the value specified in SecurityContext takes
precedence.
                                    type: string
                                type: object
                            type: object
                          startupProbe:
```

```
                    description: Probes are not allowed for ephemeral
                      containers.
                    properties:
                      exec:
                        description: Exec specifies the action to
                          take.
                        properties:
                          command:
                            description: |-
                              Command is the command line to execute inside the container, the working
directory for the
                              command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                              not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                              a shell, you need to explicitly call out to that shell.
                              Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                            items:
                              type: string
                            type: array
                        type: object
                      failureThreshold:
                        description: |-
                          Minimum consecutive failures for the probe to be considered failed after
having succeeded.
                          Defaults to 3. Minimum value is 1.
                        format: int32
                        type: integer
                      grpc:
                        description: |-
                          GRPC specifies an action involving a GRPC port.
                          This is a beta field and requires enabling GRPCContainerProbe feature gate.
                        properties:
                          port:
                            description: Port number of the gRPC
                              service. Number must be in the range
                              1 to 65535.
                            format: int32
                            type: integer
                          service:
                            description: |-
                              Service is the name of the service to place in the gRPC
HealthCheckRequest
                              (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


                              If this is not specified, the default behavior is defined by gRPC.
                            type: string
                        required:
                        - port
                        type: object
                      httpGet:
                        description: HTTPGet specifies the http
                          request to perform.
                        properties:
                          host:
```

```
                description: |-
                  Host name to connect to, defaults to the pod IP. You probably want to set
                  "Host" in httpHeaders instead.
                type: string
              httpHeaders:
                description: Custom headers to set in
                  the request. HTTP allows repeated
                  headers.
                items:
                  description: HTTPHeader describes
                    a custom header to be used in HTTP
                    probes
                  properties:
                    name:
                      description: The header field
                        name
                      type: string
                    value:
                      description: The header field
                        value
                      type: string
                  required:
                  - name
                  - value
                  type: object
                type: array
              path:
                description: Path to access on the HTTP
                  server.
                type: string
              port:
                anyOf:
                - type: integer
                - type: string
                description: |-
                  Name or number of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
              scheme:
                description: |-
                  Scheme to use for connecting to the host.
                  Defaults to HTTP.
                type: string
            required:
            - port
            type: object
          initialDelaySeconds:
            description: |-
              Number of seconds after the container has started before liveness probes are
initiated.
              More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
            format: int32
            type: integer
          periodSeconds:
```

```
      description: |-
        How often (in seconds) to perform the probe.
        Default to 10 seconds. Minimum value is 1.
      format: int32
      type: integer
    successThreshold:
      description: |-
        Minimum consecutive successes for the probe to be considered successful
after having failed.
        Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
      format: int32
      type: integer
    tcpSocket:
      description: TCPSocket specifies an action
        involving a TCP port.
      properties:
        host:
          description: 'Optional: Host name to
            connect to, defaults to the pod IP.'
          type: string
        port:
          anyOf:
          - type: integer
          - type: string
          description: |-
            Number or name of the port to access on the container.
            Number must be in the range 1 to 65535.
            Name must be an IANA_SVC_NAME.
          x-kubernetes-int-or-string: true
      required:
      - port
      type: object
    terminationGracePeriodSeconds:
      description: |-
        Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
        The grace period is the duration in seconds after the processes running in the
pod are sent
        a termination signal and the time when the processes are forcibly halted with
a kill signal.
        Set this value longer than the expected cleanup time for your process.
        If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
        value overrides the value provided by the pod spec.
        Value must be non-negative integer. The value zero indicates stop
immediately via
        the kill signal (no opportunity to shut down).
        This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
        Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
      format: int64
      type: integer
    timeoutSeconds:
      description: |-
        Number of seconds after which the probe times out.
        Defaults to 1 second. Minimum value is 1.
```

```
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                format: int32
                type: integer
            type: object
          stdin:
            description: |-
              Whether this container should allocate a buffer for stdin in the container runtime.
If this
              is not set, reads from stdin in the container will always result in EOF.
              Default is false.
            type: boolean
          stdinOnce:
            description: |-
              Whether the container runtime should close the stdin channel after it has been
opened by
              a single attach. When stdin is true the stdin stream will remain open across
multiple attach
              sessions. If stdinOnce is set to true, stdin is opened on container start, is empty
until the
              first client attaches to stdin, and then remains open and accepts data until the
client disconnects,
              at which time stdin is closed and remains closed until the container is restarted.
If this
              flag is false, a container processes that reads from stdin will never receive an
EOF.
              Default is false
            type: boolean
          targetContainerName:
            description: |-
              If set, the name of the container from PodSpec that this ephemeral container
targets.
              The ephemeral container will be run in the namespaces (IPC, PID, etc) of this
container.
              If not set then the ephemeral container uses the namespaces configured in the
Pod spec.


              The container runtime must implement support for this feature. If the runtime
does not
              support namespace targeting then the result of setting this field is undefined.
            type: string
          terminationMessagePath:
            description: |-
              Optional: Path at which the file to which the container's termination message
              will be written is mounted into the container's filesystem.
              Message written is intended to be brief final status, such as an assertion failure
message.
              Will be truncated by the node if greater than 4096 bytes. The total message
length across
              all containers will be limited to 12kb.
              Defaults to /dev/termination-log.
              Cannot be updated.
            type: string
          terminationMessagePolicy:
            description: |-
```

```
                  Indicate how the termination message should be populated. File will use the
contents of
                  terminationMessagePath to populate the container status message on both
success and failure.
                  FallbackToLogsOnError will use the last chunk of container log output if the
termination
                  message file is empty and the container exited with an error.
                  The log output is limited to 2048 bytes or 80 lines, whichever is smaller.
                  Defaults to File.
                  Cannot be updated.
                type: string
              tty:
                description: |-
                  Whether this container should allocate a TTY for itself, also requires 'stdin' to be
true.
                  Default is false.
                type: boolean
              volumeDevices:
                description: volumeDevices is the list of block
                  devices to be used by the container.
                items:
                  description: volumeDevice describes a mapping
                    of a raw block device within a container.
                  properties:
                    devicePath:
                      description: devicePath is the path inside
                        of the container that the device will
                        be mapped to.
                      type: string
                    name:
                      description: name must match the name
                        of a persistentVolumeClaim in the pod
                      type: string
                  required:
                  - devicePath
                  - name
                  type: object
                type: array
              volumeMounts:
                description: |-
                  Pod volumes to mount into the container's filesystem. Subpath mounts are not
allowed for ephemeral containers.
                  Cannot be updated.
                items:
                  description: VolumeMount describes a mounting
                    of a Volume within a container.
                  properties:
                    mountPath:
                      description: |-
                        Path within the container at which the volume should be mounted.  Must
                        not contain ':'.
                      type: string
                    mountPropagation:
                      description: |-
                        mountPropagation determines how mounts are propagated from the host
                        to container and the other way around.
```

```
                When not set, MountPropagationNone is used.
                This field is beta in 1.10.
              type: string
            name:
              description: This must match the Name
                of a Volume.
              type: string
            readOnly:
              description: |-
                Mounted read-only if true, read-write otherwise (false or unspecified).
                Defaults to false.
              type: boolean
            subPath:
              description: |-
                Path within the volume from which the container's volume should be
mounted.
                Defaults to "" (volume's root).
              type: string
            subPathExpr:
              description: |-
                Expanded path within the volume from which the container's volume should
be mounted.
                Behaves similarly to SubPath but environment variable references
$(VAR_NAME) are expanded using the container's environment.
                Defaults to "" (volume's root).
                SubPathExpr and SubPath are mutually exclusive.
              type: string
          required:
          - mountPath
          - name
          type: object
        type: array
      workingDir:
        description: |-
          Container's working directory.
          If not specified, the container runtime's default will be used, which
          might be configured in the container image.
          Cannot be updated.
        type: string
    required:
    - name
    type: object
  type: array
hostAliases:
  description: |-
    HostAliases is an optional list of hosts and IPs that will be injected into the pod's
hosts
    file if specified. This is only valid for non-hostNetwork pods.
  items:
    description: |-
      HostAlias holds the mapping between IP and hostnames that will be injected as an
entry in the
      pod's hosts file.
    properties:
      hostnames:
        description: Hostnames for the above IP address.
```

```
                items:
                  type: string
                type: array
              ip:
                description: IP address of the host file entry.
                type: string
            type: object
          type: array
        hostIPC:
          description: |-
            Use the host's ipc namespace.
            Optional: Default to false.
          type: boolean
        hostNetwork:
          description: |-
            Host networking requested for this pod. Use the host's network namespace.
            If this option is set, the ports that will be used must be specified.
            Default to false.
          type: boolean
        hostPID:
          description: |-
            Use the host's pid namespace.
            Optional: Default to false.
          type: boolean
        hostname:
          description: |-
            Specifies the hostname of the Pod
            If not specified, the pod's hostname will be set to a system-defined value.
          type: string
        imagePullSecrets:
          description: |-
            ImagePullSecrets is an optional list of references to secrets in the same namespace
to use for pulling any of the images used by this PodSpec.
            If specified, these secrets will be passed to individual puller implementations for
them to use.
            More info: https://kubernetes.io/docs/concepts/containers/images#specifying-
imagepullsecrets-on-a-pod
          items:
            description: |-
              LocalObjectReference contains enough information to let you locate the
              referenced object inside the same namespace.
            properties:
              name:
                description: |-
                  Name of the referent.
                  More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                  TODO: Add other useful fields. apiVersion, kind, uid?
                type: string
            type: object
            x-kubernetes-map-type: atomic
          type: array
        initContainers:
          description: |-
            List of initialization containers belonging to the pod.
            Init containers are executed in order prior to containers being started. If any
```

```
                    init container fails, the pod is considered to have failed and is handled according
                    to its restartPolicy. The name for an init container or normal container must be
                    unique among all containers.
                    Init containers may not have Lifecycle actions, Readiness probes, Liveness probes,
or Startup probes.
                    The resourceRequirements of an init container are taken into account during
scheduling
                    by finding the highest request/limit for each resource type, and then using the max
of
                    of that value or the sum of the normal containers. Limits are applied to init
containers
                    in a similar fashion.
                    Init containers cannot currently be added or removed.
                    Cannot be updated.
                    More info: https://kubernetes.io/docs/concepts/workloads/pods/init-containers/
                  items:
                    description: A single application container that
                      you want to run within a pod.
                    properties:
                      args:
                        description: |-
                          Arguments to the entrypoint.
                          The container image's CMD is used if this is not provided.
                          Variable references $(VAR_NAME) are expanded using the container's
environment. If a variable
                          cannot be resolved, the reference in the input string will be unchanged. Double
$$ are reduced
                          to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
"$$(VAR_NAME)" will
                          produce the string literal "$(VAR_NAME)". Escaped references will never be
expanded, regardless
                          of whether the variable exists or not. Cannot be updated.
                          More info: https://kubernetes.io/docs/tasks/inject-data-application/define-
command-argument-container/#running-a-command-in-a-shell
                        items:
                          type: string
                        type: array
                      command:
                        description: |-
                          Entrypoint array. Not executed within a shell.
                          The container image's ENTRYPOINT is used if this is not provided.
                          Variable references $(VAR_NAME) are expanded using the container's
environment. If a variable
                          cannot be resolved, the reference in the input string will be unchanged. Double
$$ are reduced
                          to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
"$$(VAR_NAME)" will
                          produce the string literal "$(VAR_NAME)". Escaped references will never be
expanded, regardless
                          of whether the variable exists or not. Cannot be updated.
                          More info: https://kubernetes.io/docs/tasks/inject-data-application/define-
command-argument-container/#running-a-command-in-a-shell
                        items:
                          type: string
                        type: array
                      env:
```

```
                        description: |-
                          List of environment variables to set in the container.
                          Cannot be updated.
                        items:
                          description: EnvVar represents an environment
                            variable present in a Container.
                          properties:
                            name:
                              description: Name of the environment variable.
                                Must be a C_IDENTIFIER.
                              type: string
                            value:
                              description: |-
                                Variable references $(VAR_NAME) are expanded
                                using the previously defined environment variables in the container and
                                any service environment variables. If a variable cannot be resolved,
                                the reference in the input string will be unchanged. Double $$ are reduced
                                to a single $, which allows for escaping the $(VAR_NAME) syntax: i.e.
                                "$$(VAR_NAME)" will produce the string literal "$(VAR_NAME)".
                                Escaped references will never be expanded, regardless of whether the
variable
                                exists or not.
                                Defaults to "".
                              type: string
                            valueFrom:
                              description: Source for the environment
                                variable's value. Cannot be used if
                                value is not empty.
                              properties:
                                configMapKeyRef:
                                  description: Selects a key of a ConfigMap.
                                  properties:
                                    key:
                                      description: The key to select.
                                      type: string
                                    name:
                                      description: |-
                                        Name of the referent.
                                        More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                        TODO: Add other useful fields. apiVersion, kind, uid?
                                      type: string
                                    optional:
                                      description: Specify whether the
                                        ConfigMap or its key must be
                                        defined
                                      type: boolean
                                  required:
                                  - key
                                  type: object
                                  x-kubernetes-map-type: atomic
                                fieldRef:
                                  description: |-
                                    Selects a field of the pod: supports metadata.name,
metadata.namespace, `metadata.labels['<KEY>']`, `metadata.annotations['<KEY>']`,
```

```
                                    spec.nodeName, spec.serviceAccountName, status.hostIP, status.podIP,
status.podIPs.
                                  properties:
                                    apiVersion:
                                      description: Version of the schema
                                        the FieldPath is written in
                                        terms of, defaults to "v1".
                                      type: string
                                    fieldPath:
                                      description: Path of the field
                                        to select in the specified API
                                        version.
                                      type: string
                                  required:
                                  - fieldPath
                                  type: object
                                  x-kubernetes-map-type: atomic
                                resourceFieldRef:
                                  description: |-
                                    Selects a resource of the container: only resources limits and requests
                                    (limits.cpu, limits.memory, limits.ephemeral-storage, requests.cpu,
requests.memory and requests.ephemeral-storage) are currently supported.
                                  properties:
                                    containerName:
                                      description: 'Container name:
                                        required for volumes, optional
                                        for env vars'
                                      type: string
                                    divisor:
                                      anyOf:
                                      - type: integer
                                      - type: string
                                      description: Specifies the output
                                        format of the exposed resources,
                                        defaults to "1"
                                      pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                      x-kubernetes-int-or-string: true
                                    resource:
                                      description: 'Required: resource
                                        to select'
                                      type: string
                                  required:
                                  - resource
                                  type: object
                                  x-kubernetes-map-type: atomic
                                secretKeyRef:
                                  description: Selects a key of a secret
                                    in the pod's namespace
                                  properties:
                                    key:
                                      description: The key of the secret
                                        to select from.  Must be a valid
                                        secret key.
                                      type: string
                                    name:
```

```
                                    description: |-
                                      Name of the referent.
                                      More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                      TODO: Add other useful fields. apiVersion, kind, uid?
                                    type: string
                                  optional:
                                    description: Specify whether the
                                      Secret or its key must be defined
                                    type: boolean
                                required:
                                - key
                                type: object
                                x-kubernetes-map-type: atomic
                              type: object
                            required:
                            - name
                            type: object
                          type: array
                        envFrom:
                          description: |-
                            List of sources to populate environment variables in the container.
                            The keys defined within a source must be a C_IDENTIFIER. All invalid keys
                            will be reported as an event when the container is starting. When a key exists in
multiple
                            sources, the value associated with the last source will take precedence.
                            Values defined by an Env with a duplicate key will take precedence.
                            Cannot be updated.
                          items:
                            description: EnvFromSource represents the
                              source of a set of ConfigMaps
                            properties:
                              configMapRef:
                                description: The ConfigMap to select from
                                properties:
                                  name:
                                    description: |-
                                      Name of the referent.
                                      More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                      TODO: Add other useful fields. apiVersion, kind, uid?
                                    type: string
                                  optional:
                                    description: Specify whether the ConfigMap
                                      must be defined
                                    type: boolean
                                type: object
                                x-kubernetes-map-type: atomic
                              prefix:
                                description: An optional identifier to
                                  prepend to each key in the ConfigMap.
                                  Must be a C_IDENTIFIER.
                                type: string
                              secretRef:
                                description: The Secret to select from
                                properties:
```

```
                                    name:
                                      description: |-
                                        Name of the referent.
                                        More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                        TODO: Add other useful fields. apiVersion, kind, uid?
                                      type: string
                                    optional:
                                      description: Specify whether the Secret
                                        must be defined
                                      type: boolean
                                  type: object
                                  x-kubernetes-map-type: atomic
                              type: object
                            type: array
                          image:
                            description: |-
                              Container image name.
                              More info: https://kubernetes.io/docs/concepts/containers/images
                              This field is optional to allow higher level config management to default or
override
                              container images in workload controllers like Deployments and StatefulSets.
                            type: string
                          imagePullPolicy:
                            description: |-
                              Image pull policy.
                              One of Always, Never, IfNotPresent.
                              Defaults to Always if :latest tag is specified, or IfNotPresent otherwise.
                              Cannot be updated.
                              More info: https://kubernetes.io/docs/concepts/containers/images#updating-
images
                            type: string
                          lifecycle:
                            description: |-
                              Actions that the management system should take in response to container
lifecycle events.
                              Cannot be updated.
                            properties:
                              postStart:
                                description: |-
                                  PostStart is called immediately after a container is created. If the handler fails,
                                  the container is terminated and restarted according to its restart policy.
                                  Other management of the container blocks until the hook completes.
                                  More info: https://kubernetes.io/docs/concepts/containers/container-lifecycle-
hooks/#container-hooks
                                properties:
                                  exec:
                                    description: Exec specifies the action
                                      to take.
                                    properties:
                                      command:
                                        description: |-
                                          Command is the command line to execute inside the container, the
working directory for the
                                          command  is root ('/') in the container's filesystem. The command is
simply exec'd, it is
```

```
                not run inside a shell, so traditional shell instructions ('|', etc) won't work.
To use
                a shell, you need to explicitly call out to that shell.
                Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
              items:
                type: string
              type: array
            type: object
          httpGet:
            description: HTTPGet specifies the http
              request to perform.
            properties:
              host:
                description: |-
                  Host name to connect to, defaults to the pod IP. You probably want to
set
                  "Host" in httpHeaders instead.
                type: string
              httpHeaders:
                description: Custom headers to set
                  in the request. HTTP allows repeated
                  headers.
                items:
                  description: HTTPHeader describes
                    a custom header to be used in
                    HTTP probes
                  properties:
                    name:
                      description: The header field
                        name
                      type: string
                    value:
                      description: The header field
                        value
                      type: string
                  required:
                  - name
                  - value
                  type: object
                type: array
              path:
                description: Path to access on the
                  HTTP server.
                type: string
              port:
                anyOf:
                - type: integer
                - type: string
                description: |-
                  Name or number of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
              scheme:
                description: |-
                  Scheme to use for connecting to the host.
```

```
                Defaults to HTTP.
              type: string
            required:
            - port
            type: object
          tcpSocket:
            description: |-
              Deprecated. TCPSocket is NOT supported as a LifecycleHandler and kept
              for the backward compatibility. There are no validation of this field and
              lifecycle hooks will fail in runtime when tcp handler is specified.
            properties:
              host:
                description: 'Optional: Host name
                  to connect to, defaults to the
                  pod IP.'
                type: string
              port:
                anyOf:
                - type: integer
                - type: string
                description: |-
                  Number or name of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
            required:
            - port
            type: object
        type: object
      preStop:
        description: |-
          PreStop is called immediately before a container is terminated due to an
          API request or management event such as liveness/startup probe failure,
          preemption, resource contention, etc. The handler is not called if the
          container crashes or exits. The Pod's termination grace period countdown
begins before the
          PreStop hook is executed. Regardless of the outcome of the handler, the
          container will eventually terminate within the Pod's termination grace
          period (unless delayed by finalizers). Other management of the container
blocks until the hook completes
          or until the termination grace period is reached.
          More info: https://kubernetes.io/docs/concepts/containers/container-lifecycle-
hooks/#container-hooks
        properties:
          exec:
            description: Exec specifies the action
              to take.
            properties:
              command:
                description: |-
                  Command is the command line to execute inside the container, the
working directory for the
                  command  is root ('/') in the container's filesystem. The command is
simply exec'd, it is
                  not run inside a shell, so traditional shell instructions ('|', etc) won't work.
To use
```

```
                a shell, you need to explicitly call out to that shell.
                Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
              items:
                type: string
              type: array
            type: object
          httpGet:
            description: HTTPGet specifies the http
              request to perform.
            properties:
              host:
                description: |-
                  Host name to connect to, defaults to the pod IP. You probably want to
set
                  "Host" in httpHeaders instead.
                type: string
              httpHeaders:
                description: Custom headers to set
                  in the request. HTTP allows repeated
                  headers.
                items:
                  description: HTTPHeader describes
                    a custom header to be used in
                    HTTP probes
                  properties:
                    name:
                      description: The header field
                        name
                      type: string
                    value:
                      description: The header field
                        value
                      type: string
                  required:
                  - name
                  - value
                  type: object
                type: array
              path:
                description: Path to access on the
                  HTTP server.
                type: string
              port:
                anyOf:
                - type: integer
                - type: string
                description: |-
                  Name or number of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
              scheme:
                description: |-
                  Scheme to use for connecting to the host.
                  Defaults to HTTP.
                type: string
```

```
                required:
                - port
                type: object
              tcpSocket:
                description: |-
                  Deprecated. TCPSocket is NOT supported as a LifecycleHandler and kept
                  for the backward compatibility. There are no validation of this field and
                  lifecycle hooks will fail in runtime when tcp handler is specified.
                properties:
                  host:
                    description: 'Optional: Host name
                      to connect to, defaults to the
                      pod IP.'
                    type: string
                  port:
                    anyOf:
                    - type: integer
                    - type: string
                    description: |-
                      Number or name of the port to access on the container.
                      Number must be in the range 1 to 65535.
                      Name must be an IANA_SVC_NAME.
                    x-kubernetes-int-or-string: true
                required:
                - port
                type: object
            type: object
          type: object
        livenessProbe:
          description: |-
            Periodic probe of container liveness.
            Container will be restarted if the probe fails.
            Cannot be updated.
            More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
          properties:
            exec:
              description: Exec specifies the action to
                take.
              properties:
                command:
                  description: |-
                    Command is the command line to execute inside the container, the working
directory for the
                    command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                    not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                    a shell, you need to explicitly call out to that shell.
                    Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                  items:
                    type: string
                  type: array
              type: object
            failureThreshold:
              description: |-
```

```
      Minimum consecutive failures for the probe to be considered failed after
having succeeded.
      Defaults to 3. Minimum value is 1.
    format: int32
    type: integer
  grpc:
    description: |-
      GRPC specifies an action involving a GRPC port.
      This is a beta field and requires enabling GRPCContainerProbe feature gate.
    properties:
      port:
        description: Port number of the gRPC
          service. Number must be in the range
          1 to 65535.
        format: int32
        type: integer
      service:
        description: |-
          Service is the name of the service to place in the gRPC
HealthCheckRequest
          (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


          If this is not specified, the default behavior is defined by gRPC.
        type: string
    required:
    - port
    type: object
  httpGet:
    description: HTTPGet specifies the http
      request to perform.
    properties:
      host:
        description: |-
          Host name to connect to, defaults to the pod IP. You probably want to set
          "Host" in httpHeaders instead.
        type: string
      httpHeaders:
        description: Custom headers to set in
          the request. HTTP allows repeated
          headers.
        items:
          description: HTTPHeader describes
            a custom header to be used in HTTP
            probes
          properties:
            name:
              description: The header field
                name
              type: string
            value:
              description: The header field
                value
              type: string
          required:
          - name
```

```
                                    - value
                                    type: object
                                  type: array
                                path:
                                  description: Path to access on the HTTP
                                    server.
                                  type: string
                                port:
                                  anyOf:
                                  - type: integer
                                  - type: string
                                  description: |-
                                    Name or number of the port to access on the container.
                                    Number must be in the range 1 to 65535.
                                    Name must be an IANA_SVC_NAME.
                                  x-kubernetes-int-or-string: true
                                scheme:
                                  description: |-
                                    Scheme to use for connecting to the host.
                                    Defaults to HTTP.
                                  type: string
                              required:
                              - port
                              type: object
                            initialDelaySeconds:
                              description: |-
                                Number of seconds after the container has started before liveness probes are
initiated.
                                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                              format: int32
                              type: integer
                            periodSeconds:
                              description: |-
                                How often (in seconds) to perform the probe.
                                Default to 10 seconds. Minimum value is 1.
                              format: int32
                              type: integer
                            successThreshold:
                              description: |-
                                Minimum consecutive successes for the probe to be considered successful
after having failed.
                                Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
                              format: int32
                              type: integer
                            tcpSocket:
                              description: TCPSocket specifies an action
                                involving a TCP port.
                              properties:
                                host:
                                  description: 'Optional: Host name to
                                    connect to, defaults to the pod IP.'
                                  type: string
                                port:
                                  anyOf:
                                  - type: integer
```

```
                - type: string
                description: |-
                  Number or name of the port to access on the container.
                  Number must be in the range 1 to 65535.
                  Name must be an IANA_SVC_NAME.
                x-kubernetes-int-or-string: true
              required:
              - port
              type: object
            terminationGracePeriodSeconds:
              description: |-
                Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
                The grace period is the duration in seconds after the processes running in the
pod are sent
                a termination signal and the time when the processes are forcibly halted with
a kill signal.
                Set this value longer than the expected cleanup time for your process.
                If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
                value overrides the value provided by the pod spec.
                Value must be non-negative integer. The value zero indicates stop
immediately via
                the kill signal (no opportunity to shut down).
                This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
                Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
              format: int64
              type: integer
            timeoutSeconds:
              description: |-
                Number of seconds after which the probe times out.
                Defaults to 1 second. Minimum value is 1.
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
              format: int32
              type: integer
          type: object
        name:
          description: |-
            Name of the container specified as a DNS_LABEL.
            Each container in a pod must have a unique name (DNS_LABEL).
            Cannot be updated.
          type: string
        ports:
          description: |-
            List of ports to expose from the container. Exposing a port here gives
            the system additional information about the network connections a
            container uses, but is primarily informational. Not specifying a port here
            DOES NOT prevent that port from being exposed. Any port which is
            listening on the default "0.0.0.0" address inside a container will be
            accessible from the network.
            Cannot be updated.
          items:
            description: ContainerPort represents a network
              port in a single container.
```

```
                properties:
                  containerPort:
                    description: |-
                      Number of port to expose on the pod's IP address.
                      This must be a valid port number, 0 < x < 65536.
                    format: int32
                    type: integer
                  hostIP:
                    description: What host IP to bind the
                      external port to.
                    type: string
                  hostPort:
                    description: |-
                      Number of port to expose on the host.
                      If specified, this must be a valid port number, 0 < x < 65536.
                      If HostNetwork is specified, this must match ContainerPort.
                      Most containers do not need this.
                    format: int32
                    type: integer
                  name:
                    description: |-
                      If specified, this must be an IANA_SVC_NAME and unique within the pod.
Each
                      named port in a pod must have a unique name. Name for the port that can
be
                      referred to by services.
                    type: string
                  protocol:
                    default: TCP
                    description: |-
                      Protocol for port. Must be UDP, TCP, or SCTP.
                      Defaults to "TCP".
                    type: string
                required:
                - containerPort
                type: object
              type: array
              x-kubernetes-list-map-keys:
              - containerPort
              - protocol
              x-kubernetes-list-type: map
            readinessProbe:
              description: |-
                Periodic probe of container service readiness.
                Container will be removed from service endpoints if the probe fails.
                Cannot be updated.
                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
              properties:
                exec:
                  description: Exec specifies the action to
                    take.
                  properties:
                    command:
                      description: |-
```

```
                  Command is the command line to execute inside the container, the working
directory for the
                  command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                  not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                  a shell, you need to explicitly call out to that shell.
                  Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                items:
                  type: string
                type: array
              type: object
            failureThreshold:
              description: |-
                Minimum consecutive failures for the probe to be considered failed after
having succeeded.
                Defaults to 3. Minimum value is 1.
              format: int32
              type: integer
            grpc:
              description: |-
                GRPC specifies an action involving a GRPC port.
                This is a beta field and requires enabling GRPCContainerProbe feature gate.
              properties:
                port:
                  description: Port number of the gRPC
                    service. Number must be in the range
                    1 to 65535.
                  format: int32
                  type: integer
                service:
                  description: |-
                    Service is the name of the service to place in the gRPC
HealthCheckRequest
                    (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


                    If this is not specified, the default behavior is defined by gRPC.
                  type: string
              required:
              - port
              type: object
            httpGet:
              description: HTTPGet specifies the http
                request to perform.
              properties:
                host:
                  description: |-
                    Host name to connect to, defaults to the pod IP. You probably want to set
                    "Host" in httpHeaders instead.
                  type: string
                httpHeaders:
                  description: Custom headers to set in
                    the request. HTTP allows repeated
                    headers.
                  items:
```

```
                                    description: HTTPHeader describes
                                      a custom header to be used in HTTP
                                      probes
                                    properties:
                                      name:
                                        description: The header field
                                          name
                                        type: string
                                      value:
                                        description: The header field
                                          value
                                        type: string
                                    required:
                                    - name
                                    - value
                                    type: object
                                  type: array
                                path:
                                  description: Path to access on the HTTP
                                    server.
                                  type: string
                                port:
                                  anyOf:
                                  - type: integer
                                  - type: string
                                  description: |-
                                    Name or number of the port to access on the container.
                                    Number must be in the range 1 to 65535.
                                    Name must be an IANA_SVC_NAME.
                                  x-kubernetes-int-or-string: true
                                scheme:
                                  description: |-
                                    Scheme to use for connecting to the host.
                                    Defaults to HTTP.
                                  type: string
                              required:
                              - port
                              type: object
                            initialDelaySeconds:
                              description: |-
                                Number of seconds after the container has started before liveness probes are
initiated.
                                More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                              format: int32
                              type: integer
                            periodSeconds:
                              description: |-
                                How often (in seconds) to perform the probe.
                                Default to 10 seconds. Minimum value is 1.
                              format: int32
                              type: integer
                            successThreshold:
                              description: |-
                                Minimum consecutive successes for the probe to be considered successful
after having failed.
```

```
                    Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
                  format: int32
                  type: integer
                tcpSocket:
                  description: TCPSocket specifies an action
                    involving a TCP port.
                  properties:
                    host:
                      description: 'Optional: Host name to
                        connect to, defaults to the pod IP.'
                      type: string
                    port:
                      anyOf:
                      - type: integer
                      - type: string
                      description: |-
                        Number or name of the port to access on the container.
                        Number must be in the range 1 to 65535.
                        Name must be an IANA_SVC_NAME.
                      x-kubernetes-int-or-string: true
                  required:
                  - port
                  type: object
                terminationGracePeriodSeconds:
                  description: |-
                    Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
                    The grace period is the duration in seconds after the processes running in the
pod are sent
                    a termination signal and the time when the processes are forcibly halted with
a kill signal.
                    Set this value longer than the expected cleanup time for your process.
                    If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
                    value overrides the value provided by the pod spec.
                    Value must be non-negative integer. The value zero indicates stop
immediately via
                    the kill signal (no opportunity to shut down).
                    This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
                    Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
                  format: int64
                  type: integer
                timeoutSeconds:
                  description: |-
                    Number of seconds after which the probe times out.
                    Defaults to 1 second. Minimum value is 1.
                    More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                  format: int32
                  type: integer
              type: object
            resources:
              description: |-
                Compute Resources required by this container.
                Cannot be updated.
```

```
                More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-
containers/
              properties:
                limits:
                  additionalProperties:
                    anyOf:
                    - type: integer
                    - type: string
                    pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                    x-kubernetes-int-or-string: true
                  description: |-
                    Limits describes the maximum amount of compute resources allowed.
                    More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                  type: object
                requests:
                  additionalProperties:
                    anyOf:
                    - type: integer
                    - type: string
                    pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                    x-kubernetes-int-or-string: true
                  description: |-
                    Requests describes the minimum amount of compute resources required.
                    If Requests is omitted for a container, it defaults to Limits if that is explicitly
specified,
                    otherwise to an implementation-defined value.
                    More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                  type: object
              type: object
            securityContext:
              description: |-
                SecurityContext defines the security options the container should be run with.
                If set, the fields of SecurityContext override the equivalent fields of
PodSecurityContext.
                More info: https://kubernetes.io/docs/tasks/configure-pod-container/security-
context/
              properties:
                allowPrivilegeEscalation:
                  description: |-
                    AllowPrivilegeEscalation controls whether a process can gain more
                    privileges than its parent process. This bool directly controls if
                    the no_new_privs flag will be set on the container process.
                    AllowPrivilegeEscalation is true always when the container is:
                    1) run as Privileged
                    2) has CAP_SYS_ADMIN
                    Note that this field cannot be set when spec.os.name is windows.
                  type: boolean
                capabilities:
                  description: |-
                    The capabilities to add/drop when running containers.
                    Defaults to the default set of capabilities granted by the container runtime.
                    Note that this field cannot be set when spec.os.name is windows.
```

```
      properties:
        add:
          description: Added capabilities
          items:
            description: Capability represent
              POSIX capabilities type
            type: string
          type: array
        drop:
          description: Removed capabilities
          items:
            description: Capability represent
              POSIX capabilities type
            type: string
          type: array
      type: object
    privileged:
      description: |-
        Run container in privileged mode.
        Processes in privileged containers are essentially equivalent to root on the
host.
        Defaults to false.
        Note that this field cannot be set when spec.os.name is windows.
      type: boolean
    procMount:
      description: |-
        procMount denotes the type of proc mount to use for the containers.
        The default is DefaultProcMount which uses the container runtime defaults for
        readonly paths and masked paths.
        This requires the ProcMountType feature flag to be enabled.
        Note that this field cannot be set when spec.os.name is windows.
      type: string
    readOnlyRootFilesystem:
      description: |-
        Whether this container has a read-only root filesystem.
        Default is false.
        Note that this field cannot be set when spec.os.name is windows.
      type: boolean
    runAsGroup:
      description: |-
        The GID to run the entrypoint of the container process.
        Uses runtime default if unset.
        May also be set in PodSecurityContext.  If set in both SecurityContext and
        PodSecurityContext, the value specified in SecurityContext takes precedence.
        Note that this field cannot be set when spec.os.name is windows.
      format: int64
      type: integer
    runAsNonRoot:
      description: |-
        Indicates that the container must run as a non-root user.
        If true, the Kubelet will validate the image at runtime to ensure that it
        does not run as UID 0 (root) and fail to start the container if it does.
        If unset or false, no such validation will be performed.
        May also be set in PodSecurityContext.  If set in both SecurityContext and
        PodSecurityContext, the value specified in SecurityContext takes precedence.
      type: boolean
```

```
                runAsUser:
                  description: |-
                    The UID to run the entrypoint of the container process.
                    Defaults to user specified in image metadata if unspecified.
                    May also be set in PodSecurityContext.  If set in both SecurityContext and
                    PodSecurityContext, the value specified in SecurityContext takes precedence.
                    Note that this field cannot be set when spec.os.name is windows.
                  format: int64
                  type: integer
                seLinuxOptions:
                  description: |-
                    The SELinux context to be applied to the container.
                    If unspecified, the container runtime will allocate a random SELinux context
for each
                    container.  May also be set in PodSecurityContext.  If set in both
SecurityContext and
                    PodSecurityContext, the value specified in SecurityContext takes precedence.
                    Note that this field cannot be set when spec.os.name is windows.
                  properties:
                    level:
                      description: Level is SELinux level
                        label that applies to the container.
                      type: string
                    role:
                      description: Role is a SELinux role
                        label that applies to the container.
                      type: string
                    type:
                      description: Type is a SELinux type
                        label that applies to the container.
                      type: string
                    user:
                      description: User is a SELinux user
                        label that applies to the container.
                      type: string
                  type: object
                seccompProfile:
                  description: |-
                    The seccomp options to use by this container. If seccomp options are
                    provided at both the pod & container level, the container options
                    override the pod options.
                    Note that this field cannot be set when spec.os.name is windows.
                  properties:
                    localhostProfile:
                      description: |-
                        localhostProfile indicates a profile defined in a file on the node should be
used.
                        The profile must be preconfigured on the node to work.
                        Must be a descending path, relative to the kubelet's configured seccomp
profile location.
                        Must only be set if type is "Localhost".
                      type: string
                    type:
                      description: |-
                        type indicates which kind of seccomp profile will be applied.
                        Valid options are:
```

```
                                    Localhost - a profile defined in a file on the node should be used.
                                    RuntimeDefault - the container runtime default profile should be used.
                                    Unconfined - no profile should be applied.
                                  type: string
                                required:
                                - type
                                type: object
                              windowsOptions:
                                description: |-
                                  The Windows specific settings applied to all containers.
                                  If unspecified, the options from the PodSecurityContext will be used.
                                  If set in both SecurityContext and PodSecurityContext, the value specified in
SecurityContext takes precedence.
                                  Note that this field cannot be set when spec.os.name is linux.
                                properties:
                                  gmsaCredentialSpec:
                                    description: |-
                                      GMSACredentialSpec is where the GMSA admission webhook
                                      (https://github.com/kubernetes-sigs/windows-gmsa) inlines the contents of
the
                                      GMSA credential spec named by the GMSACredentialSpecName field.
                                    type: string
                                  gmsaCredentialSpecName:
                                    description: GMSACredentialSpecName
                                      is the name of the GMSA credential
                                      spec to use.
                                    type: string
                                  hostProcess:
                                    description: |-
                                      HostProcess determines if a container should be run as a 'Host Process'
container.
                                      This field is alpha-level and will only be honored by components that
enable the
                                      WindowsHostProcessContainers feature flag. Setting this field without the
feature
                                      flag will result in errors when validating the Pod. All of a Pod's containers
must
                                      have the same effective HostProcess value (it is not allowed to have a mix
of HostProcess
                                      containers and non-HostProcess containers).  In addition, if HostProcess is
true
                                      then HostNetwork must also be set to true.
                                    type: boolean
                                  runAsUserName:
                                    description: |-
                                      The UserName in Windows to run the entrypoint of the container process.
                                      Defaults to the user specified in image metadata if unspecified.
                                      May also be set in PodSecurityContext. If set in both SecurityContext and
                                      PodSecurityContext, the value specified in SecurityContext takes
precedence.
                                    type: string
                                type: object
                            type: object
                          startupProbe:
```

```
                    description: |-
                      StartupProbe indicates that the Pod has successfully initialized.
                      If specified, no other probes are executed until this completes successfully.
                      If this probe fails, the Pod will be restarted, just as if the livenessProbe failed.
                      This can be used to provide different probe parameters at the beginning of a
Pod's lifecycle,
                      when it might take a long time to load data or warm a cache, than during steady-
state operation.
                      This cannot be updated.
                      More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                    properties:
                      exec:
                        description: Exec specifies the action to
                          take.
                        properties:
                          command:
                            description: |-
                              Command is the command line to execute inside the container, the working
directory for the
                              command  is root ('/') in the container's filesystem. The command is simply
exec'd, it is
                              not run inside a shell, so traditional shell instructions ('|', etc) won't work. To
use
                              a shell, you need to explicitly call out to that shell.
                              Exit status of 0 is treated as live/healthy and non-zero is unhealthy.
                            items:
                              type: string
                            type: array
                        type: object
                      failureThreshold:
                        description: |-
                          Minimum consecutive failures for the probe to be considered failed after
having succeeded.
                          Defaults to 3. Minimum value is 1.
                        format: int32
                        type: integer
                      grpc:
                        description: |-
                          GRPC specifies an action involving a GRPC port.
                          This is a beta field and requires enabling GRPCContainerProbe feature gate.
                        properties:
                          port:
                            description: Port number of the gRPC
                              service. Number must be in the range
                              1 to 65535.
                            format: int32
                            type: integer
                          service:
                            description: |-
                              Service is the name of the service to place in the gRPC
HealthCheckRequest
                              (see https://github.com/grpc/grpc/blob/master/doc/health-checking.md).


                              If this is not specified, the default behavior is defined by gRPC.
```

```
      type: string
    required:
    - port
    type: object
  httpGet:
    description: HTTPGet specifies the http
      request to perform.
    properties:
      host:
        description: |-
          Host name to connect to, defaults to the pod IP. You probably want to set
          "Host" in httpHeaders instead.
        type: string
      httpHeaders:
        description: Custom headers to set in
          the request. HTTP allows repeated
          headers.
        items:
          description: HTTPHeader describes
            a custom header to be used in HTTP
            probes
          properties:
            name:
              description: The header field
                name
              type: string
            value:
              description: The header field
                value
              type: string
          required:
          - name
          - value
          type: object
        type: array
      path:
        description: Path to access on the HTTP
          server.
        type: string
      port:
        anyOf:
        - type: integer
        - type: string
        description: |-
          Name or number of the port to access on the container.
          Number must be in the range 1 to 65535.
          Name must be an IANA_SVC_NAME.
        x-kubernetes-int-or-string: true
      scheme:
        description: |-
          Scheme to use for connecting to the host.
          Defaults to HTTP.
        type: string
    required:
    - port
    type: object
```

```
                        initialDelaySeconds:
                          description: |-
                            Number of seconds after the container has started before liveness probes are
initiated.
                            More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                          format: int32
                          type: integer
                        periodSeconds:
                          description: |-
                            How often (in seconds) to perform the probe.
                            Default to 10 seconds. Minimum value is 1.
                          format: int32
                          type: integer
                        successThreshold:
                          description: |-
                            Minimum consecutive successes for the probe to be considered successful
after having failed.
                            Defaults to 1. Must be 1 for liveness and startup. Minimum value is 1.
                          format: int32
                          type: integer
                        tcpSocket:
                          description: TCPSocket specifies an action
                            involving a TCP port.
                          properties:
                            host:
                              description: 'Optional: Host name to
                                connect to, defaults to the pod IP.'
                              type: string
                            port:
                              anyOf:
                              - type: integer
                              - type: string
                              description: |-
                                Number or name of the port to access on the container.
                                Number must be in the range 1 to 65535.
                                Name must be an IANA_SVC_NAME.
                              x-kubernetes-int-or-string: true
                          required:
                          - port
                          type: object
                        terminationGracePeriodSeconds:
                          description: |-
                            Optional duration in seconds the pod needs to terminate gracefully upon
probe failure.
                            The grace period is the duration in seconds after the processes running in the
pod are sent
                            a termination signal and the time when the processes are forcibly halted with
a kill signal.
                            Set this value longer than the expected cleanup time for your process.
                            If this value is nil, the pod's terminationGracePeriodSeconds will be used.
Otherwise, this
                            value overrides the value provided by the pod spec.
                            Value must be non-negative integer. The value zero indicates stop
immediately via
                            the kill signal (no opportunity to shut down).
```

```
                  This is a beta field and requires enabling ProbeTerminationGracePeriod
feature gate.
                  Minimum value is 1. spec.terminationGracePeriodSeconds is used if unset.
                format: int64
                type: integer
              timeoutSeconds:
                description: |-
                  Number of seconds after which the probe times out.
                  Defaults to 1 second. Minimum value is 1.
                  More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle#container-probes
                format: int32
                type: integer
            type: object
          stdin:
            description: |-
              Whether this container should allocate a buffer for stdin in the container runtime.
If this
              is not set, reads from stdin in the container will always result in EOF.
              Default is false.
            type: boolean
          stdinOnce:
            description: |-
              Whether the container runtime should close the stdin channel after it has been
opened by
              a single attach. When stdin is true the stdin stream will remain open across
multiple attach
              sessions. If stdinOnce is set to true, stdin is opened on container start, is empty
until the
              first client attaches to stdin, and then remains open and accepts data until the
client disconnects,
              at which time stdin is closed and remains closed until the container is restarted.
If this
              flag is false, a container processes that reads from stdin will never receive an
EOF.
              Default is false
            type: boolean
          terminationMessagePath:
            description: |-
              Optional: Path at which the file to which the container's termination message
              will be written is mounted into the container's filesystem.
              Message written is intended to be brief final status, such as an assertion failure
message.
              Will be truncated by the node if greater than 4096 bytes. The total message
length across
              all containers will be limited to 12kb.
              Defaults to /dev/termination-log.
              Cannot be updated.
            type: string
          terminationMessagePolicy:
            description: |-
              Indicate how the termination message should be populated. File will use the
contents of
              terminationMessagePath to populate the container status message on both
success and failure.
```

```
                FallbackToLogsOnError will use the last chunk of container log output if the
termination
                message file is empty and the container exited with an error.
                The log output is limited to 2048 bytes or 80 lines, whichever is smaller.
                Defaults to File.
                Cannot be updated.
              type: string
            tty:
              description: |-
                Whether this container should allocate a TTY for itself, also requires 'stdin' to be
true.
                Default is false.
              type: boolean
            volumeDevices:
              description: volumeDevices is the list of block
                devices to be used by the container.
              items:
                description: volumeDevice describes a mapping
                  of a raw block device within a container.
                properties:
                  devicePath:
                    description: devicePath is the path inside
                      of the container that the device will
                      be mapped to.
                    type: string
                  name:
                    description: name must match the name
                      of a persistentVolumeClaim in the pod
                    type: string
                required:
                - devicePath
                - name
                type: object
              type: array
            volumeMounts:
              description: |-
                Pod volumes to mount into the container's filesystem.
                Cannot be updated.
              items:
                description: VolumeMount describes a mounting
                  of a Volume within a container.
                properties:
                  mountPath:
                    description: |-
                      Path within the container at which the volume should be mounted.  Must
                      not contain ':'.
                    type: string
                  mountPropagation:
                    description: |-
                      mountPropagation determines how mounts are propagated from the host
                      to container and the other way around.
                      When not set, MountPropagationNone is used.
                      This field is beta in 1.10.
                    type: string
                  name:
                    description: This must match the Name
```

```
                                    of a Volume.
                                  type: string
                                readOnly:
                                  description: |-
                                    Mounted read-only if true, read-write otherwise (false or unspecified).
                                    Defaults to false.
                                  type: boolean
                                subPath:
                                  description: |-
                                    Path within the volume from which the container's volume should be
mounted.
                                    Defaults to "" (volume's root).
                                  type: string
                                subPathExpr:
                                  description: |-
                                    Expanded path within the volume from which the container's volume should
be mounted.
                                    Behaves similarly to SubPath but environment variable references
$(VAR_NAME) are expanded using the container's environment.
                                    Defaults to "" (volume's root).
                                    SubPathExpr and SubPath are mutually exclusive.
                                  type: string
                              required:
                              - mountPath
                              - name
                              type: object
                            type: array
                          workingDir:
                            description: |-
                              Container's working directory.
                              If not specified, the container runtime's default will be used, which
                              might be configured in the container image.
                              Cannot be updated.
                            type: string
                        required:
                        - name
                        type: object
                      type: array
                    nodeName:
                      description: |-
                        NodeName is a request to schedule this pod onto a specific node. If it is non-empty,
                        the scheduler simply schedules this pod onto that node, assuming that it fits
resource
                        requirements.
                      type: string
                    nodeSelector:
                      additionalProperties:
                        type: string
                      description: |-
                        NodeSelector is a selector which must be true for the pod to fit on a node.
                        Selector which must match a node's labels for the pod to be scheduled on that
node.
                        More info: https://kubernetes.io/docs/concepts/configuration/assign-pod-node/
                      type: object
                      x-kubernetes-map-type: atomic
                    os:
```

```
                  description: |-
                    Specifies the OS of the containers in the pod.
                    Some pod and container fields are restricted if this is set.


                    If the OS field is set to linux, the following fields must be unset:
                    -securityContext.windowsOptions


                    If the OS field is set to windows, following fields must be unset:
                    - spec.hostPID
                    - spec.hostIPC
                    - spec.securityContext.seLinuxOptions
                    - spec.securityContext.seccompProfile
                    - spec.securityContext.fsGroup
                    - spec.securityContext.fsGroupChangePolicy
                    - spec.securityContext.sysctls
                    - spec.shareProcessNamespace
                    - spec.securityContext.runAsUser
                    - spec.securityContext.runAsGroup
                    - spec.securityContext.supplementalGroups
                    - spec.containers[*].securityContext.seLinuxOptions
                    - spec.containers[*].securityContext.seccompProfile
                    - spec.containers[*].securityContext.capabilities
                    - spec.containers[*].securityContext.readOnlyRootFilesystem
                    - spec.containers[*].securityContext.privileged
                    - spec.containers[*].securityContext.allowPrivilegeEscalation
                    - spec.containers[*].securityContext.procMount
                    - spec.containers[*].securityContext.runAsUser
                    - spec.containers[*].securityContext.runAsGroup
                    This is a beta field and requires the IdentifyPodOS feature
                  properties:
                    name:
                      description: |-
                        Name is the name of the operating system. The currently supported values are
linux and windows.
                        Additional value may be defined in future and can be one of:
                        https://github.com/opencontainers/runtime-spec/blob/master/config.md#platform-
specific-configuration
                        Clients should expect to handle additional values and treat unrecognized values
in this field as os: null
                      type: string
                  required:
                  - name
                  type: object
                overhead:
                  additionalProperties:
                    anyOf:
                    - type: integer
                    - type: string
                    pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-
)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                    x-kubernetes-int-or-string: true
                  description: |-
                    Overhead represents the resource overhead associated with running a pod for a
given RuntimeClass.
```

```
                    This field will be autopopulated at admission time by the RuntimeClass admission
controller. If
                    the RuntimeClass admission controller is enabled, overhead must not be set in Pod
create requests.
                    The RuntimeClass admission controller will reject Pod create requests which have
the overhead already
                    set. If RuntimeClass is configured and selected in the PodSpec, Overhead will be
set to the value
                    defined in the corresponding RuntimeClass, otherwise it will remain unset and
treated as zero.
                    More info: https://git.k8s.io/enhancements/keps/sig-node/688-pod-
overhead/README.md
                  type: object
                preemptionPolicy:
                  description: |-
                    PreemptionPolicy is the Policy for preempting pods with lower priority.
                    One of Never, PreemptLowerPriority.
                    Defaults to PreemptLowerPriority if unset.
                  type: string
                priority:
                  description: |-
                    The priority value. Various system components use this field to find the
                    priority of the pod. When Priority Admission Controller is enabled, it
                    prevents users from setting this field. The admission controller populates
                    this field from PriorityClassName.
                    The higher the value, the higher the priority.
                  format: int32
                  type: integer
                priorityClassName:
                  description: |-
                    If specified, indicates the pod's priority. "system-node-critical" and
                    "system-cluster-critical" are two special keywords which indicate the
                    highest priorities with the former being the highest priority. Any other
                    name must be defined by creating a PriorityClass object with that name.
                    If not specified, the pod priority will be default or zero if there is no
                    default.
                  type: string
                readinessGates:
                  description: |-
                    If specified, all readiness gates will be evaluated for pod readiness.
                    A pod is ready when all its containers are ready AND
                    all conditions specified in the readiness gates have status equal to "True"
                    More info: https://git.k8s.io/enhancements/keps/sig-network/580-pod-readiness-
gates
                  items:
                    description: PodReadinessGate contains the reference
                      to a pod condition
                    properties:
                      conditionType:
                        description: ConditionType refers to a condition
                          in the pod's condition list with matching
                          type.
                        type: string
                    required:
                    - conditionType
                    type: object
```

```
                type: array
              restartPolicy:
                description: |-
                  Restart policy for all containers within the pod.
                  One of Always, OnFailure, Never.
                  Default to Always.
                  More info: https://kubernetes.io/docs/concepts/workloads/pods/pod-
lifecycle/#restart-policy
                type: string
              runtimeClassName:
                description: |-
                  RuntimeClassName refers to a RuntimeClass object in the node.k8s.io group, which
should be used
                  to run this pod.  If no RuntimeClass resource matches the named class, the pod will
not be run.
                  If unset or empty, the "legacy" RuntimeClass will be used, which is an implicit class
with an
                  empty definition that uses the default runtime handler.
                  More info: https://git.k8s.io/enhancements/keps/sig-node/585-runtime-class
                type: string
              schedulerName:
                description: |-
                  If specified, the pod will be dispatched by specified scheduler.
                  If not specified, the pod will be dispatched by default scheduler.
                type: string
              securityContext:
                description: |-
                  SecurityContext holds pod-level security attributes and common container settings.
                  Optional: Defaults to empty.  See type description for default values of each field.
                properties:
                  fsGroup:
                    description: |-
                      A special supplemental group that applies to all containers in a pod.
                      Some volume types allow the Kubelet to change the ownership of that volume
                      to be owned by the pod:


                      1. The owning GID will be the FSGroup
                      2. The setgid bit is set (new files created in the volume will be owned by
FSGroup)
                      3. The permission bits are OR'd with rw-rw----


                      If unset, the Kubelet will not modify the ownership and permissions of any
volume.
                      Note that this field cannot be set when spec.os.name is windows.
                    format: int64
                    type: integer
                  fsGroupChangePolicy:
                    description: |-
                      fsGroupChangePolicy defines behavior of changing ownership and permission of
the volume
                      before being exposed inside Pod. This field will only apply to
                      volume types which support fsGroup based ownership(and permissions).
                      It will have no effect on ephemeral volume types such as: secret, configmaps
                      and emptydir.
```

```
                    Valid values are "OnRootMismatch" and "Always". If not specified, "Always" is
used.
                    Note that this field cannot be set when spec.os.name is windows.
                  type: string
                runAsGroup:
                  description: |-
                    The GID to run the entrypoint of the container process.
                    Uses runtime default if unset.
                    May also be set in SecurityContext.  If set in both SecurityContext and
                    PodSecurityContext, the value specified in SecurityContext takes precedence
                    for that container.
                    Note that this field cannot be set when spec.os.name is windows.
                  format: int64
                  type: integer
                runAsNonRoot:
                  description: |-
                    Indicates that the container must run as a non-root user.
                    If true, the Kubelet will validate the image at runtime to ensure that it
                    does not run as UID 0 (root) and fail to start the container if it does.
                    If unset or false, no such validation will be performed.
                    May also be set in SecurityContext.  If set in both SecurityContext and
                    PodSecurityContext, the value specified in SecurityContext takes precedence.
                  type: boolean
                runAsUser:
                  description: |-
                    The UID to run the entrypoint of the container process.
                    Defaults to user specified in image metadata if unspecified.
                    May also be set in SecurityContext.  If set in both SecurityContext and
                    PodSecurityContext, the value specified in SecurityContext takes precedence
                    for that container.
                    Note that this field cannot be set when spec.os.name is windows.
                  format: int64
                  type: integer
                seLinuxOptions:
                  description: |-
                    The SELinux context to be applied to all containers.
                    If unspecified, the container runtime will allocate a random SELinux context for
each
                    container.  May also be set in SecurityContext.  If set in
                    both SecurityContext and PodSecurityContext, the value specified in
SecurityContext
                    takes precedence for that container.
                    Note that this field cannot be set when spec.os.name is windows.
                  properties:
                    level:
                      description: Level is SELinux level label
                        that applies to the container.
                      type: string
                    role:
                      description: Role is a SELinux role label
                        that applies to the container.
                      type: string
                    type:
                      description: Type is a SELinux type label
                        that applies to the container.
                      type: string
```

```
        user:
          description: User is a SELinux user label
            that applies to the container.
          type: string
      type: object
    seccompProfile:
      description: |-
        The seccomp options to use by the containers in this pod.
        Note that this field cannot be set when spec.os.name is windows.
      properties:
        localhostProfile:
          description: |-
            localhostProfile indicates a profile defined in a file on the node should be used.
            The profile must be preconfigured on the node to work.
            Must be a descending path, relative to the kubelet's configured seccomp profile
location.

            Must only be set if type is "Localhost".
          type: string
        type:
          description: |-
            type indicates which kind of seccomp profile will be applied.
            Valid options are:


            Localhost - a profile defined in a file on the node should be used.
            RuntimeDefault - the container runtime default profile should be used.
            Unconfined - no profile should be applied.
          type: string
      required:
      - type
      type: object
    supplementalGroups:
      description: |-
        A list of groups applied to the first process run in each container, in addition
        to the container's primary GID.  If unspecified, no groups will be added to
        any container.
        Note that this field cannot be set when spec.os.name is windows.
      items:
        format: int64
        type: integer
      type: array
    sysctls:
      description: |-
        Sysctls hold a list of namespaced sysctls used for the pod. Pods with
unsupported

        sysctls (by the container runtime) might fail to launch.
        Note that this field cannot be set when spec.os.name is windows.
      items:
        description: Sysctl defines a kernel parameter
          to be set
        properties:
          name:
            description: Name of a property to set
            type: string
          value:
            description: Value of a property to set
```

```
                type: string
              required:
              - name
              - value
              type: object
            type: array
          windowsOptions:
            description: |-
              The Windows specific settings applied to all containers.
              If unspecified, the options within a container's SecurityContext will be used.
              If set in both SecurityContext and PodSecurityContext, the value specified in
SecurityContext takes precedence.
              Note that this field cannot be set when spec.os.name is linux.
            properties:
              gmsaCredentialSpec:
                description: |-
                  GMSACredentialSpec is where the GMSA admission webhook
                  (https://github.com/kubernetes-sigs/windows-gmsa) inlines the contents of the
                  GMSA credential spec named by the GMSACredentialSpecName field.
                type: string
              gmsaCredentialSpecName:
                description: GMSACredentialSpecName is the
                  name of the GMSA credential spec to use.
                type: string
              hostProcess:
                description: |-
                  HostProcess determines if a container should be run as a 'Host Process'
container.
                  This field is alpha-level and will only be honored by components that enable
the
                  WindowsHostProcessContainers feature flag. Setting this field without the
feature
                  flag will result in errors when validating the Pod. All of a Pod's containers must
                  have the same effective HostProcess value (it is not allowed to have a mix of
HostProcess
                  containers and non-HostProcess containers).  In addition, if HostProcess is
true
                  then HostNetwork must also be set to true.
                type: boolean
              runAsUserName:
                description: |-
                  The UserName in Windows to run the entrypoint of the container process.
                  Defaults to the user specified in image metadata if unspecified.
                  May also be set in PodSecurityContext. If set in both SecurityContext and
                  PodSecurityContext, the value specified in SecurityContext takes precedence.
                type: string
            type: object
        type: object
      serviceAccount:
        description: |-
          DeprecatedServiceAccount is a depreciated alias for ServiceAccountName.
          Deprecated: Use serviceAccountName instead.
        type: string
      serviceAccountName:
        description: |-
          ServiceAccountName is the name of the ServiceAccount to use to run this pod.
```

```
                More info: https://kubernetes.io/docs/tasks/configure-pod-container/configure-service-account/
              type: string
            setHostnameAsFQDN:
              description: |-
                If true the pod's hostname will be configured as the pod's FQDN, rather than the leaf name (the default).
                In Linux containers, this means setting the FQDN in the hostname field of the kernel (the nodename field of struct utsname).
                In Windows containers, this means setting the registry value of hostname for the registry key HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\Tcpip\\Parameters to FQDN.
                If a pod does not have FQDN, this has no effect.
                Default to false.
              type: boolean
            shareProcessNamespace:
              description: |-
                Share a single process namespace between all of the containers in a pod.
                When this is set containers will be able to view and signal processes from other containers
                in the same pod, and the first process in each container will not be assigned PID 1.
                HostPID and ShareProcessNamespace cannot both be set.
                Optional: Default to false.
              type: boolean
            subdomain:
              description: |-
                If specified, the fully qualified Pod hostname will be "<hostname>.<subdomain>.<pod namespace>.svc.<cluster domain>".
                If not specified, the pod will not have a domainname at all.
              type: string
            terminationGracePeriodSeconds:
              description: |-
                Optional duration in seconds the pod needs to terminate gracefully. May be decreased in delete request.
                Value must be non-negative integer. The value zero indicates stop immediately via
                the kill signal (no opportunity to shut down).
                If this value is nil, the default grace period will be used instead.
                The grace period is the duration in seconds after the processes running in the pod are sent
                a termination signal and the time when the processes are forcibly halted with a kill signal.
                Set this value longer than the expected cleanup time for your process.
                Defaults to 30 seconds.
              format: int64
              type: integer
            tolerations:
              description: If specified, the pod's tolerations.
              items:
                description: |-
                  The pod this Toleration is attached to tolerates any taint that matches
                  the triple <key,value,effect> using the matching operator <operator>.
                properties:
                  effect:
                    description: |-
                      Effect indicates the taint effect to match. Empty means match all taint effects.
```

```
                When specified, allowed values are NoSchedule, PreferNoSchedule and
NoExecute.
              type: string
            key:
              description: |-
                Key is the taint key that the toleration applies to. Empty means match all taint
keys.
                If the key is empty, operator must be Exists; this combination means to match all
values and all keys.
              type: string
            operator:
              description: |-
                Operator represents a key's relationship to the value.
                Valid operators are Exists and Equal. Defaults to Equal.
                Exists is equivalent to wildcard for value, so that a pod can
                tolerate all taints of a particular category.
              type: string
            tolerationSeconds:
              description: |-
                TolerationSeconds represents the period of time the toleration (which must be
                of effect NoExecute, otherwise this field is ignored) tolerates the taint. By default,
                it is not set, which means tolerate the taint forever (do not evict). Zero and
                negative values will be treated as 0 (evict immediately) by the system.
              format: int64
              type: integer
            value:
              description: |-
                Value is the taint value the toleration matches to.
                If the operator is Exists, the value should be empty, otherwise just a regular
string.
              type: string
          type: object
        type: array
      topologySpreadConstraints:
        description: |-
          TopologySpreadConstraints describes how a group of pods ought to spread across
topology
          domains. Scheduler will schedule pods in a way which abides by the constraints.
          All topologySpreadConstraints are ANDed.
        items:
          description: TopologySpreadConstraint specifies
            how to spread matching pods among the given topology.
          properties:
            labelSelector:
              description: |-
                LabelSelector is used to find matching pods.
                Pods that match this label selector are counted to determine the number of pods
                in their corresponding topology domain.
              properties:
                matchExpressions:
                  description: matchExpressions is a list
                    of label selector requirements. The requirements
                    are ANDed.
                  items:
                    description: |-
```

```
                    A label selector requirement is a selector that contains values, a key, and an
operator that
                    relates the key and values.
                  properties:
                    key:
                      description: key is the label key
                        that the selector applies to.
                      type: string
                    operator:
                      description: |-
                        operator represents a key's relationship to a set of values.
                        Valid operators are In, NotIn, Exists and DoesNotExist.
                      type: string
                    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.
                      items:
                        type: string
                      type: array
                  required:
                  - key
                  - operator
                  type: object
                type: array
              matchLabels:
                additionalProperties:
                  type: string
                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
              type: object
              x-kubernetes-map-type: atomic
            maxSkew:
              description: |-
                MaxSkew describes the degree to which pods may be unevenly distributed.
                When `whenUnsatisfiable=DoNotSchedule`, it is the maximum permitted
difference
                between the number of matching pods in the target topology and the global
minimum.
                The global minimum is the minimum number of matching pods in an eligible
domain
                or zero if the number of eligible domains is less than MinDomains.
                For example, in a 3-zone cluster, MaxSkew is set to 1, and pods with the same
                labelSelector spread as 2/2/1:
                In this case, the global minimum is 1.
                | zone1 | zone2 | zone3 |
                |  P P  |  P P  |   P   |
```

```
                - if MaxSkew is 1, incoming pod can only be scheduled to zone3 to become
2/2/2;
                scheduling it onto zone1(zone2) would make the ActualSkew(3-1) on
zone1(zone2)
                violate MaxSkew(1).
                - if MaxSkew is 2, incoming pod can be scheduled onto any zone.
                When `whenUnsatisfiable=ScheduleAnyway`, it is used to give higher
precedence
                to topologies that satisfy it.
                It's a required field. Default value is 1 and 0 is not allowed.
              format: int32
              type: integer
            minDomains:
              description: |-
                MinDomains indicates a minimum number of eligible domains.
                When the number of eligible domains with matching topology keys is less than
minDomains,
                Pod Topology Spread treats "global minimum" as 0, and then the calculation of
Skew is performed.
                And when the number of eligible domains with matching topology keys equals or
greater than minDomains,
                this value has no effect on scheduling.
                As a result, when the number of eligible domains is less than minDomains,
                scheduler won't schedule more than maxSkew Pods to those domains.
                If value is nil, the constraint behaves as if MinDomains is equal to 1.
                Valid values are integers greater than 0.
                When value is not nil, WhenUnsatisfiable must be DoNotSchedule.


                For example, in a 3-zone cluster, MaxSkew is set to 2, MinDomains is set to 5
and pods with the same
                labelSelector spread as 2/2/2:
                | zone1 | zone2 | zone3 |
                |  P P  |  P P  |  P P  |
                The number of domains is less than 5(MinDomains), so "global minimum" is
treated as 0.
                In this situation, new pod with the same labelSelector cannot be scheduled,
                because computed skew will be 3(3 - 0) if new Pod is scheduled to any of the
three zones,
                it will violate MaxSkew.


                This is an alpha field and requires enabling MinDomainsInPodTopologySpread
feature gate.
              format: int32
              type: integer
            topologyKey:
              description: |-
                TopologyKey is the key of node labels. Nodes that have a label with this key
                and identical values are considered to be in the same topology.
                We consider each <key, value> as a "bucket", and try to put balanced number
                of pods into each bucket.
                We define a domain as a particular instance of a topology.
                Also, we define an eligible domain as a domain whose nodes match the node
selector.
```

```
                    e.g. If TopologyKey is "kubernetes.io/hostname", each Node is a domain of that
topology.
                    And, if TopologyKey is "topology.kubernetes.io/zone", each zone is a domain of
that topology.
                    It's a required field.
                  type: string
                whenUnsatisfiable:
                  description: |-
                    WhenUnsatisfiable indicates how to deal with a pod if it doesn't satisfy
                    the spread constraint.
                    - DoNotSchedule (default) tells the scheduler not to schedule it.
                    - ScheduleAnyway tells the scheduler to schedule the pod in any location,
                      but giving higher precedence to topologies that would help reduce the
                      skew.
                    A constraint is considered "Unsatisfiable" for an incoming pod
                    if and only if every possible node assignment for that pod would violate
                    "MaxSkew" on some topology.
                    For example, in a 3-zone cluster, MaxSkew is set to 1, and pods with the same
                    labelSelector spread as 3/1/1:
                    | zone1 | zone2 | zone3 |
                    | P P P |   P   |   P   |
                    If WhenUnsatisfiable is set to DoNotSchedule, incoming pod can only be
scheduled
                    to zone2(zone3) to become 3/2/1(3/1/2) as ActualSkew(2-1) on zone2(zone3)
satisfies
                    MaxSkew(1). In other words, the cluster can still be imbalanced, but scheduler
                    won't make it *more* imbalanced.
                    It's a required field.
                  type: string
              required:
              - maxSkew
              - topologyKey
              - whenUnsatisfiable
              type: object
            type: array
            x-kubernetes-list-map-keys:
            - topologyKey
            - whenUnsatisfiable
            x-kubernetes-list-type: map
          volumes:
            description: |-
              List of volumes that can be mounted by containers belonging to the pod.
              More info: https://kubernetes.io/docs/concepts/storage/volumes
            items:
              description: Volume represents a named volume in
                a pod that may be accessed by any container in
                the pod.
              properties:
                awsElasticBlockStore:
                  description: |-
                    awsElasticBlockStore represents an AWS Disk resource that is attached to a
                    kubelet's host machine and then exposed to the pod.
                    More info:
https://kubernetes.io/docs/concepts/storage/volumes#awselasticblockstore
                  properties:
                    fsType:
```

```
                                    description: |-
                                      fsType is the filesystem type of the volume that you want to mount.
                                      Tip: Ensure that the filesystem type is supported by the host operating
system.
                                      Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
                                      More info:
https://kubernetes.io/docs/concepts/storage/volumes#awselasticblockstore
                                      TODO: how do we prevent errors in the filesystem from compromising the
machine
                                    type: string
                                  partition:
                                    description: |-
                                      partition is the partition in the volume that you want to mount.
                                      If omitted, the default is to mount by volume name.
                                      Examples: For volume /dev/sda1, you specify the partition as "1".
                                      Similarly, the volume partition for /dev/sda is "0" (or you can leave the
property empty).
                                    format: int32
                                    type: integer
                                  readOnly:
                                    description: |-
                                      readOnly value true will force the readOnly setting in VolumeMounts.
                                      More info:
https://kubernetes.io/docs/concepts/storage/volumes#awselasticblockstore
                                    type: boolean
                                  volumeID:
                                    description: |-
                                      volumeID is unique ID of the persistent disk resource in AWS (Amazon EBS
volume).
                                      More info:
https://kubernetes.io/docs/concepts/storage/volumes#awselasticblockstore
                                    type: string
                                required:
                                - volumeID
                                type: object
                              azureDisk:
                                description: azureDisk represents an Azure Data
                                  Disk mount on the host and bind mount to the
                                  pod.
                                properties:
                                  cachingMode:
                                    description: 'cachingMode is the Host Caching
                                      mode: None, Read Only, Read Write.'
                                    type: string
                                  diskName:
                                    description: diskName is the Name of the
                                      data disk in the blob storage
                                    type: string
                                  diskURI:
                                    description: diskURI is the URI of data
                                      disk in the blob storage
                                    type: string
                                  fsType:
                                    description: |-
                                      fsType is Filesystem type to mount.
                                      Must be a filesystem type supported by the host operating system.
```

```
          Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
        type: string
      kind:
        description: 'kind expected values are Shared:
          multiple blob disks per storage account  Dedicated:
          single blob disk per storage account  Managed:
          azure managed data disk (only in managed
          availability set). defaults to shared'
        type: string
      readOnly:
        description: |-
          readOnly Defaults to false (read/write). ReadOnly here will force
          the ReadOnly setting in VolumeMounts.
        type: boolean
    required:
    - diskName
    - diskURI
    type: object
  azureFile:
    description: azureFile represents an Azure File
      Service mount on the host and bind mount to
      the pod.
    properties:
      readOnly:
        description: |-
          readOnly defaults to false (read/write). ReadOnly here will force
          the ReadOnly setting in VolumeMounts.
        type: boolean
      secretName:
        description: secretName is the  name of
          secret that contains Azure Storage Account
          Name and Key
        type: string
      shareName:
        description: shareName is the azure share
          Name
        type: string
    required:
    - secretName
    - shareName
    type: object
  cephfs:
    description: cephFS represents a Ceph FS mount
      on the host that shares a pod's lifetime
    properties:
      monitors:
        description: |-
          monitors is Required: Monitors is a collection of Ceph monitors
          More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-
it
        items:
          type: string
        type: array
      path:
        description: 'path is Optional: Used as
          the mounted root, rather than the full
```

```
                  Ceph tree, default is /'
                type: string
              readOnly:
                description: |-
                  readOnly is Optional: Defaults to false (read/write). ReadOnly here will force
                  the ReadOnly setting in VolumeMounts.
                  More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-
it
                type: boolean
              secretFile:
                description: |-
                  secretFile is Optional: SecretFile is the path to key ring for User, default is
/etc/ceph/user.secret
                  More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-
it
                type: string
              secretRef:
                description: |-
                  secretRef is Optional: SecretRef is reference to the authentication secret for
User, default is empty.
                  More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-
it
                properties:
                  name:
                    description: |-
                      Name of the referent.
                      More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                      TODO: Add other useful fields. apiVersion, kind, uid?
                    type: string
                type: object
                x-kubernetes-map-type: atomic
              user:
                description: |-
                  user is optional: User is the rados user name, default is admin
                  More info: https://examples.k8s.io/volumes/cephfs/README.md#how-to-use-
it
                type: string
            required:
            - monitors
            type: object
          cinder:
            description: |-
              cinder represents a cinder volume attached and mounted on kubelets host
machine.
              More info: https://examples.k8s.io/mysql-cinder-pd/README.md
            properties:
              fsType:
                description: |-
                  fsType is the filesystem type to mount.
                  Must be a filesystem type supported by the host operating system.
                  Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
                  More info: https://examples.k8s.io/mysql-cinder-pd/README.md
                type: string
              readOnly:
                description: |-
```

```
                        readOnly defaults to false (read/write). ReadOnly here will force
                        the ReadOnly setting in VolumeMounts.
                        More info: https://examples.k8s.io/mysql-cinder-pd/README.md
                      type: boolean
                    secretRef:
                      description: |-
                        secretRef is optional: points to a secret object containing parameters used to
connect
                        to OpenStack.
                      properties:
                        name:
                          description: |-
                            Name of the referent.
                            More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                            TODO: Add other useful fields. apiVersion, kind, uid?
                          type: string
                      type: object
                      x-kubernetes-map-type: atomic
                    volumeID:
                      description: |-
                        volumeID used to identify the volume in cinder.
                        More info: https://examples.k8s.io/mysql-cinder-pd/README.md
                      type: string
                  required:
                  - volumeID
                  type: object
                configMap:
                  description: configMap represents a configMap
                    that should populate this volume
                  properties:
                    defaultMode:
                      description: |-
                        defaultMode is optional: mode bits used to set permissions on created files by
default.
                        Must be an octal value between 0000 and 0777 or a decimal value between 0
and 511.
                        YAML accepts both octal and decimal values, JSON requires decimal values
for mode bits.
                        Defaults to 0644.
                        Directories within the path are not affected by this setting.
                        This might be in conflict with other options that affect the file
                        mode, like fsGroup, and the result can be other mode bits set.
                      format: int32
                      type: integer
                    items:
                      description: |-
                        items if unspecified, each key-value pair in the Data field of the referenced
                        ConfigMap will be projected into the volume as a file whose name is the
                        key and content is the value. If specified, the listed keys will be
                        projected into the specified paths, and unlisted keys will not be
                        present. If a key is specified which is not present in the ConfigMap,
                        the volume setup will error unless it is marked optional. Paths must be
                        relative and may not contain the '..' path or start with '..'.
                      items:
                        description: Maps a string key to a path
```

```
                within a volume.
              properties:
                key:
                  description: key is the key to project.
                  type: string
                mode:
                  description: |-
                    mode is Optional: mode bits used to set permissions on this file.
                    Must be an octal value between 0000 and 0777 or a decimal value
between 0 and 511.
                    YAML accepts both octal and decimal values, JSON requires decimal
values for mode bits.
                    If not specified, the volume defaultMode will be used.
                    This might be in conflict with other options that affect the file
                    mode, like fsGroup, and the result can be other mode bits set.
                  format: int32
                  type: integer
                path:
                  description: |-
                    path is the relative path of the file to map the key to.
                    May not be an absolute path.
                    May not contain the path element '..'.
                    May not start with the string '..'.
                  type: string
              required:
              - key
              - path
              type: object
            type: array
          name:
            description: |-
              Name of the referent.
              More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
              TODO: Add other useful fields. apiVersion, kind, uid?
            type: string
          optional:
            description: optional specify whether the
              ConfigMap or its keys must be defined
            type: boolean
        type: object
        x-kubernetes-map-type: atomic
      csi:
        description: csi (Container Storage Interface)
          represents ephemeral storage that is handled
          by certain external CSI drivers (Beta feature).
        properties:
          driver:
            description: |-
              driver is the name of the CSI driver that handles this volume.
              Consult with your admin for the correct name as registered in the cluster.
            type: string
          fsType:
            description: |-
              fsType to mount. Ex. "ext4", "xfs", "ntfs".
              If not provided, the empty value is passed to the associated CSI driver
```

```
                which will determine the default filesystem to apply.
              type: string
            nodePublishSecretRef:
              description: |-
                nodePublishSecretRef is a reference to the secret object containing
                sensitive information to pass to the CSI driver to complete the CSI
                NodePublishVolume and NodeUnpublishVolume calls.
                This field is optional, and  may be empty if no secret is required. If the
                secret object contains more than one secret, all secret references are passed.
              properties:
                name:
                  description: |-
                    Name of the referent.
                    More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                    TODO: Add other useful fields. apiVersion, kind, uid?
                  type: string
              type: object
              x-kubernetes-map-type: atomic
            readOnly:
              description: |-
                readOnly specifies a read-only configuration for the volume.
                Defaults to false (read/write).
              type: boolean
            volumeAttributes:
              additionalProperties:
                type: string
              description: |-
                volumeAttributes stores driver-specific properties that are passed to the CSI
                driver. Consult your driver's documentation for supported values.
              type: object
          required:
          - driver
          type: object
        downwardAPI:
          description: downwardAPI represents downward
            API about the pod that should populate this
            volume
          properties:
            defaultMode:
              description: |-
                Optional: mode bits to use on created files by default. Must be a
                Optional: mode bits used to set permissions on created files by default.
                Must be an octal value between 0000 and 0777 or a decimal value between 0
and 511.
                YAML accepts both octal and decimal values, JSON requires decimal values
for mode bits.
                Defaults to 0644.
                Directories within the path are not affected by this setting.
                This might be in conflict with other options that affect the file
                mode, like fsGroup, and the result can be other mode bits set.
              format: int32
              type: integer
            items:
              description: Items is a list of downward
                API volume file
```

```
                        items:
                          description: DownwardAPIVolumeFile represents
                            information to create the file containing
                            the pod field
                          properties:
                            fieldRef:
                              description: 'Required: Selects a
                                field of the pod: only annotations,
                                labels, name and namespace are supported.'
                              properties:
                                apiVersion:
                                  description: Version of the schema
                                    the FieldPath is written in
                                    terms of, defaults to "v1".
                                  type: string
                                fieldPath:
                                  description: Path of the field
                                    to select in the specified API
                                    version.
                                  type: string
                              required:
                              - fieldPath
                              type: object
                              x-kubernetes-map-type: atomic
                            mode:
                              description: |-
                                Optional: mode bits used to set permissions on this file, must be an octal
value
                                between 0000 and 0777 or a decimal value between 0 and 511.
                                YAML accepts both octal and decimal values, JSON requires decimal
values for mode bits.
                                If not specified, the volume defaultMode will be used.
                                This might be in conflict with other options that affect the file
                                mode, like fsGroup, and the result can be other mode bits set.
                              format: int32
                              type: integer
                            path:
                              description: 'Required: Path is  the
                                relative path name of the file to
                                be created. Must not be absolute
                                or contain the ''..'' path. Must
                                be utf-8 encoded. The first item
                                of the relative path must not start
                                with ''..'''
                              type: string
                            resourceFieldRef:
                              description: |-
                                Selects a resource of the container: only resources limits and requests
                                (limits.cpu, limits.memory, requests.cpu and requests.memory) are
currently supported.
                              properties:
                                containerName:
                                  description: 'Container name:
                                    required for volumes, optional
                                    for env vars'
                                  type: string
```

```
                                    divisor:
                                      anyOf:
                                      - type: integer
                                      - type: string
                                      description: Specifies the output
                                        format of the exposed resources,
                                        defaults to "1"
                                      pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                      x-kubernetes-int-or-string: true
                                    resource:
                                      description: 'Required: resource
                                        to select'
                                      type: string
                                  required:
                                  - resource
                                  type: object
                                  x-kubernetes-map-type: atomic
                              required:
                              - path
                              type: object
                            type: array
                        type: object
                      emptyDir:
                        description: |-
                          emptyDir represents a temporary directory that shares a pod's lifetime.
                          More info: https://kubernetes.io/docs/concepts/storage/volumes#emptydir
                        properties:
                          medium:
                            description: |-
                              medium represents what type of storage medium should back this directory.
                              The default is "" which means to use the node's default medium.
                              Must be an empty string (default) or Memory.
                              More info: https://kubernetes.io/docs/concepts/storage/volumes#emptydir
                            type: string
                          sizeLimit:
                            anyOf:
                            - type: integer
                            - type: string
                            description: |-
                              sizeLimit is the total amount of local storage required for this EmptyDir
volume.
                              The size limit is also applicable for memory medium.
                              The maximum usage on memory medium EmptyDir would be the minimum
value between
                              the SizeLimit specified here and the sum of memory limits of all containers in
a pod.
                              The default is nil which means that the limit is undefined.
                              More info: http://kubernetes.io/docs/user-guide/volumes#emptydir
                            pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                            x-kubernetes-int-or-string: true
                        type: object
                      ephemeral:
                        description: |-
                          ephemeral represents a volume that is handled by a cluster storage driver.
```

The volume's lifecycle is tied to the pod that defines it - it will be created before the pod starts,
and deleted when the pod is removed.

Use this if:
a) the volume is only needed while the pod runs,
b) features of normal volumes like restoring from snapshot or capacity
tracking are needed,
c) the storage driver is specified through a storage class, and
d) the storage driver supports dynamic volume provisioning through
a PersistentVolumeClaim (see EphemeralVolumeSource for more
information on the connection between this volume type
and PersistentVolumeClaim).

Use PersistentVolumeClaim or one of the vendor-specific
APIs for volumes that persist for longer than the lifecycle
of an individual pod.

Use CSI for light-weight local ephemeral volumes if the CSI driver is meant to
be used that way - see the documentation of the driver for
more information.

A pod can use both types of ephemeral volumes and
persistent volumes at the same time.
properties:
volumeClaimTemplate:
description: |-
Will be used to create a stand-alone PVC to provision the volume.
The pod in which this EphemeralVolumeSource is embedded will be the
owner of the PVC, i.e. the PVC will be deleted together with the
pod. The name of the PVC will be `<pod name>-<volume name>` where
`<volume name>` is the name from the `PodSpec.Volumes` array
entry. Pod validation will reject the pod if the concatenated name
is not valid for a PVC (for example, too long).

An existing PVC with that name that is not owned by the pod
will *not* be used for the pod to avoid using an unrelated
volume by mistake. Starting the pod is then blocked until
the unrelated PVC is removed. If such a pre-created PVC is
meant to be used by the pod, the PVC has to updated with an
owner reference to the pod once the pod exists. Normally
this should not be necessary, but it may be useful when
manually reconstructing a broken cluster.

This field is read-only and no changes will be made by Kubernetes
to the PVC after it has been created.

Required, must not be nil.
properties:

```
                    metadata:
                      description: |-
                        May contain labels and annotations that will be copied into the PVC
                        when creating it. No other fields are allowed and will be rejected during
                        validation.
                      properties:
                        annotations:
                          additionalProperties:
                            type: string
                          type: object
                        finalizers:
                          items:
                            type: string
                          type: array
                        labels:
                          additionalProperties:
                            type: string
                          type: object
                        name:
                          type: string
                        namespace:
                          type: string
                      type: object
                    spec:
                      description: |-
                        The specification for the PersistentVolumeClaim. The entire content is
                        copied unchanged into the PVC that gets created from this
                        template. The same fields as in a PersistentVolumeClaim
                        are also valid here.
                      properties:
                        accessModes:
                          description: |-
                            accessModes contains the desired access modes the volume should
have.
                            More info: https://kubernetes.io/docs/concepts/storage/persistent-
volumes#access-modes-1
                          items:
                            type: string
                          type: array
                        dataSource:
                          description: |-
                            dataSource field can be used to specify either:
                            * An existing VolumeSnapshot object
(snapshot.storage.k8s.io/VolumeSnapshot)
                            * An existing PVC (PersistentVolumeClaim)
                            If the provisioner or an external controller can support the specified data
source,
                            it will create a new volume based on the contents of the specified data
source.
                            If the AnyVolumeDataSource feature gate is enabled, this field will
always have
                            the same contents as the DataSourceRef field.
                          properties:
                            apiGroup:
                              description: |-
                                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.
              type: string
            kind:
              description: Kind is the type
                of resource being referenced
              type: string
            name:
              description: Name is the name
                of resource being referenced
              type: string
          required:
          - kind
          - name
          type: object
          x-kubernetes-map-type: atomic
        dataSourceRef:
          description: |-
            dataSourceRef specifies the object from which to populate the volume
with data, if a non-empty
            volume is desired. This may be any local object from a non-empty API
group (non
            core object) or a PersistentVolumeClaim object.
            When this field is specified, volume binding will only succeed if the type
of
            the specified object matches some installed volume populator or
dynamic
            provisioner.
            This field will replace the functionality of the DataSource field and as
such
            if both fields are non-empty, they must have the same value. For
backwards
            compatibility, both fields (DataSource and DataSourceRef) will be set to
the same
            value automatically if one of them is empty and the other is non-empty.
            There are two important differences between DataSource and
DataSourceRef:
            * While DataSource only allows two specific types of objects,
DataSourceRef
              allows any non-core object, as well as PersistentVolumeClaim objects.
            * While DataSource ignores disallowed values (dropping them),
DataSourceRef
              preserves all values, and generates an error if a disallowed value is
              specified.
            (Beta) Using this field requires the AnyVolumeDataSource feature gate
to be enabled.
          properties:
            apiGroup:
              description: |-
                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.
              type: string
            kind:
```

```
                                    description: Kind is the type
                                      of resource being referenced
                                    type: string
                                  name:
                                    description: Name is the name
                                      of resource being referenced
                                    type: string
                                required:
                                - kind
                                - name
                                type: object
                                x-kubernetes-map-type: atomic
                              resources:
                                description: |-
                                  resources represents the minimum resources the volume should have.
                                  If RecoverVolumeExpansionFailure feature is enabled users are allowed
to specify resource requirements
                                  that are lower than previous value but must still be higher than capacity
recorded in the
                                  status field of the claim.
                                  More info: https://kubernetes.io/docs/concepts/storage/persistent-
volumes#resources
                                properties:
                                  limits:
                                    additionalProperties:
                                      anyOf:
                                      - type: integer
                                      - type: string
                                      pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                      x-kubernetes-int-or-string: true
                                    description: |-
                                      Limits describes the maximum amount of compute resources allowed.
                                      More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                                    type: object
                                  requests:
                                    additionalProperties:
                                      anyOf:
                                      - type: integer
                                      - type: string
                                      pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                      x-kubernetes-int-or-string: true
                                    description: |-
                                      Requests describes the minimum amount of compute resources
required.
                                      If Requests is omitted for a container, it defaults to Limits if that is
explicitly specified,
                                      otherwise to an implementation-defined value.
                                      More info: https://kubernetes.io/docs/concepts/configuration/manage-
resources-containers/
                                    type: object
                                type: object
                              selector:
                                description: selector is a label
```

```
          query over volumes to consider
          for binding.
        properties:
          matchExpressions:
            description: matchExpressions
              is a list of label selector
              requirements. The requirements
              are ANDed.
            items:
              description: |-
                A label selector requirement is a selector that contains values, a key,
and an operator that
                relates the key and values.
              properties:
                key:
                  description: key is the
                    label key that the selector
                    applies to.
                  type: string
                operator:
                  description: |-
                    operator represents a key's relationship to a set of values.
                    Valid operators are In, NotIn, Exists and DoesNotExist.
                  type: string
                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.
                  items:
                    type: string
                  type: array
              required:
              - key
              - operator
              type: object
            type: array
          matchLabels:
            additionalProperties:
              type: string
            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
        type: object
        x-kubernetes-map-type: atomic
      storageClassName:
        description: |-
```

```
                storageClassName is the name of the StorageClass required by the
claim.
                More info: https://kubernetes.io/docs/concepts/storage/persistent-
volumes#class-1
              type: string
            volumeMode:
              description: |-
                volumeMode defines what type of volume is required by the claim.
                Value of Filesystem is implied when not included in claim spec.
              type: string
            volumeName:
              description: volumeName is the binding
                reference to the PersistentVolume
                backing this claim.
              type: string
          type: object
        required:
        - spec
        type: object
      type: object
    fc:
      description: fc represents a Fibre Channel resource
        that is attached to a kubelet's host machine
        and then exposed to the pod.
      properties:
        fsType:
          description: |-
            fsType is the filesystem type to mount.
            Must be a filesystem type supported by the host operating system.
            Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
            TODO: how do we prevent errors in the filesystem from compromising the
machine
          type: string
        lun:
          description: 'lun is Optional: FC target
            lun number'
          format: int32
          type: integer
        readOnly:
          description: |-
            readOnly is Optional: Defaults to false (read/write). ReadOnly here will force
            the ReadOnly setting in VolumeMounts.
          type: boolean
        targetWWNs:
          description: 'targetWWNs is Optional: FC
            target worldwide names (WWNs)'
          items:
            type: string
          type: array
        wwids:
          description: |-
            wwids Optional: FC volume world wide identifiers (wwids)
            Either wwids or combination of targetWWNs and lun must be set, but not both
simultaneously.
          items:
            type: string
```

```
          type: array
        type: object
      flexVolume:
        description: |-
          flexVolume represents a generic volume resource that is
          provisioned/attached using an exec based plugin.
        properties:
          driver:
            description: driver is the name of the driver
              to use for this volume.
            type: string
          fsType:
            description: |-
              fsType is the filesystem type to mount.
              Must be a filesystem type supported by the host operating system.
              Ex. "ext4", "xfs", "ntfs". The default filesystem depends on FlexVolume script.
            type: string
          options:
            additionalProperties:
              type: string
            description: 'options is Optional: this
              field holds extra command options if any.'
            type: object
          readOnly:
            description: |-
              readOnly is Optional: defaults to false (read/write). ReadOnly here will force
              the ReadOnly setting in VolumeMounts.
            type: boolean
          secretRef:
            description: |-
              secretRef is Optional: secretRef is reference to the secret object containing
              sensitive information to pass to the plugin scripts. This may be
              empty if no secret object is specified. If the secret object
              contains more than one secret, all secrets are passed to the plugin
              scripts.
            properties:
              name:
                description: |-
                  Name of the referent.
                  More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                  TODO: Add other useful fields. apiVersion, kind, uid?
                type: string
            type: object
            x-kubernetes-map-type: atomic
        required:
        - driver
        type: object
      flocker:
        description: flocker represents a Flocker volume
          attached to a kubelet's host machine. This
          depends on the Flocker control service being
          running
        properties:
          datasetName:
            description: |-
```

```
                datasetName is Name of the dataset stored as metadata -> name on the
dataset for Flocker
                should be considered as deprecated
              type: string
            datasetUUID:
              description: datasetUUID is the UUID of
                the dataset. This is unique identifier
                of a Flocker dataset
              type: string
          type: object
        gcePersistentDisk:
          description: |-
            gcePersistentDisk represents a GCE Disk resource that is attached to a
            kubelet's host machine and then exposed to the pod.
            More info:
https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk
          properties:
            fsType:
              description: |-
                fsType is filesystem type of the volume that you want to mount.
                Tip: Ensure that the filesystem type is supported by the host operating
system.
                Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
                More info:
https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk
                TODO: how do we prevent errors in the filesystem from compromising the
machine
              type: string
            partition:
              description: |-
                partition is the partition in the volume that you want to mount.
                If omitted, the default is to mount by volume name.
                Examples: For volume /dev/sda1, you specify the partition as "1".
                Similarly, the volume partition for /dev/sda is "0" (or you can leave the
property empty).
                More info:
https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk
              format: int32
              type: integer
            pdName:
              description: |-
                pdName is unique name of the PD resource in GCE. Used to identify the disk
in GCE.
                More info:
https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk
              type: string
            readOnly:
              description: |-
                readOnly here will force the ReadOnly setting in VolumeMounts.
                Defaults to false.
                More info:
https://kubernetes.io/docs/concepts/storage/volumes#gcepersistentdisk
              type: boolean
          required:
          - pdName
          type: object
```

```
                    gitRepo:
                      description: |-
                        gitRepo represents a git repository at a particular revision.
                        DEPRECATED: GitRepo is deprecated. To provision a container with a git repo,
mount an
                        EmptyDir into an InitContainer that clones the repo using git, then mount the
EmptyDir
                        into the Pod's container.
                      properties:
                        directory:
                          description: |-
                            directory is the target directory name.
                            Must not contain or start with '..'.  If '.' is supplied, the volume directory will be
the
                            git repository.  Otherwise, if specified, the volume will contain the git
repository in
                            the subdirectory with the given name.
                          type: string
                        repository:
                          description: repository is the URL
                          type: string
                        revision:
                          description: revision is the commit hash
                            for the specified revision.
                          type: string
                      required:
                      - repository
                      type: object
                    glusterfs:
                      description: |-
                        glusterfs represents a Glusterfs mount on the host that shares a pod's lifetime.
                        More info: https://examples.k8s.io/volumes/glusterfs/README.md
                      properties:
                        endpoints:
                          description: |-
                            endpoints is the endpoint name that details Glusterfs topology.
                            More info: https://examples.k8s.io/volumes/glusterfs/README.md#create-a-
pod
                          type: string
                        path:
                          description: |-
                            path is the Glusterfs volume path.
                            More info: https://examples.k8s.io/volumes/glusterfs/README.md#create-a-
pod
                          type: string
                        readOnly:
                          description: |-
                            readOnly here will force the Glusterfs volume to be mounted with read-only
permissions.
                            Defaults to false.
                            More info: https://examples.k8s.io/volumes/glusterfs/README.md#create-a-
pod
                          type: boolean
                      required:
                      - endpoints
                      - path
```

```
      type: object
    hostPath:
      description: |-
        hostPath represents a pre-existing file or directory on the host
        machine that is directly exposed to the container. This is generally
        used for system agents or other privileged things that are allowed
        to see the host machine. Most containers will NOT need this.
        More info: https://kubernetes.io/docs/concepts/storage/volumes#hostpath
        ---
        TODO(jonesdl) We need to restrict who can use host directory mounts and who
can/can not
        mount host directories as read/write.
      properties:
        path:
          description: |-
            path of the directory on the host.
            If the path is a symlink, it will follow the link to the real path.
            More info: https://kubernetes.io/docs/concepts/storage/volumes#hostpath
          type: string
        type:
          description: |-
            type for HostPath Volume
            Defaults to ""
            More info: https://kubernetes.io/docs/concepts/storage/volumes#hostpath
          type: string
      required:
      - path
      type: object
    iscsi:
      description: |-
        iscsi represents an ISCSI Disk resource that is attached to a
        kubelet's host machine and then exposed to the pod.
        More info: https://examples.k8s.io/volumes/iscsi/README.md
      properties:
        chapAuthDiscovery:
          description: chapAuthDiscovery defines whether
            support iSCSI Discovery CHAP authentication
          type: boolean
        chapAuthSession:
          description: chapAuthSession defines whether
            support iSCSI Session CHAP authentication
          type: boolean
        fsType:
          description: |-
            fsType is the filesystem type of the volume that you want to mount.
            Tip: Ensure that the filesystem type is supported by the host operating
system.
            Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
            More info: https://kubernetes.io/docs/concepts/storage/volumes#iscsi
            TODO: how do we prevent errors in the filesystem from compromising the
machine
          type: string
        initiatorName:
          description: |-
            initiatorName is the custom iSCSI Initiator Name.
```

```
                  If initiatorName is specified with iscsiInterface simultaneously, new iSCSI
interface
                  <target portal>:<volume name> will be created for the connection.
                type: string
              iqn:
                description: iqn is the target iSCSI Qualified
                  Name.
                type: string
              iscsiInterface:
                description: |-
                  iscsiInterface is the interface Name that uses an iSCSI transport.
                  Defaults to 'default' (tcp).
                type: string
              lun:
                description: lun represents iSCSI Target
                  Lun number.
                format: int32
                type: integer
              portals:
                description: |-
                  portals is the iSCSI Target Portal List. The portal is either an IP or
ip_addr:port if the port
                  is other than default (typically TCP ports 860 and 3260).
                items:
                  type: string
                type: array
              readOnly:
                description: |-
                  readOnly here will force the ReadOnly setting in VolumeMounts.
                  Defaults to false.
                type: boolean
              secretRef:
                description: secretRef is the CHAP Secret
                  for iSCSI target and initiator authentication
                properties:
                  name:
                    description: |-
                      Name of the referent.
                      More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                      TODO: Add other useful fields. apiVersion, kind, uid?
                    type: string
                type: object
                x-kubernetes-map-type: atomic
              targetPortal:
                description: |-
                  targetPortal is iSCSI Target Portal. The Portal is either an IP or ip_addr:port if
the port
                  is other than default (typically TCP ports 860 and 3260).
                type: string
            required:
            - iqn
            - lun
            - targetPortal
            type: object
          name:
```

```
                  description: |-
                    name of the volume.
                    Must be a DNS_LABEL and unique within the pod.
                    More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                  type: string
                nfs:
                  description: |-
                    nfs represents an NFS mount on the host that shares a pod's lifetime
                    More info: https://kubernetes.io/docs/concepts/storage/volumes#nfs
                  properties:
                    path:
                      description: |-
                        path that is exported by the NFS server.
                        More info: https://kubernetes.io/docs/concepts/storage/volumes#nfs
                      type: string
                    readOnly:
                      description: |-
                        readOnly here will force the NFS export to be mounted with read-only
permissions.
                        Defaults to false.
                        More info: https://kubernetes.io/docs/concepts/storage/volumes#nfs
                      type: boolean
                    server:
                      description: |-
                        server is the hostname or IP address of the NFS server.
                        More info: https://kubernetes.io/docs/concepts/storage/volumes#nfs
                      type: string
                  required:
                  - path
                  - server
                  type: object
                persistentVolumeClaim:
                  description: |-
                    persistentVolumeClaimVolumeSource represents a reference to a
                    PersistentVolumeClaim in the same namespace.
                    More info: https://kubernetes.io/docs/concepts/storage/persistent-
volumes#persistentvolumeclaims
                  properties:
                    claimName:
                      description: |-
                        claimName is the name of a PersistentVolumeClaim in the same namespace
as the pod using this volume.
                        More info: https://kubernetes.io/docs/concepts/storage/persistent-
volumes#persistentvolumeclaims
                      type: string
                    readOnly:
                      description: |-
                        readOnly Will force the ReadOnly setting in VolumeMounts.
                        Default false.
                      type: boolean
                  required:
                  - claimName
                  type: object
                photonPersistentDisk:
                  description: photonPersistentDisk represents
```

```
      a PhotonController persistent disk attached
      and mounted on kubelets host machine
    properties:
      fsType:
        description: |-
          fsType is the filesystem type to mount.
          Must be a filesystem type supported by the host operating system.
          Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
        type: string
      pdID:
        description: pdID is the ID that identifies
          Photon Controller persistent disk
        type: string
    required:
    - pdID
    type: object
  portworxVolume:
    description: portworxVolume represents a portworx
      volume attached and mounted on kubelets host
      machine
    properties:
      fsType:
        description: |-
          fSType represents the filesystem type to mount
          Must be a filesystem type supported by the host operating system.
          Ex. "ext4", "xfs". Implicitly inferred to be "ext4" if unspecified.
        type: string
      readOnly:
        description: |-
          readOnly defaults to false (read/write). ReadOnly here will force
          the ReadOnly setting in VolumeMounts.
        type: boolean
      volumeID:
        description: volumeID uniquely identifies
          a Portworx volume
        type: string
    required:
    - volumeID
    type: object
  projected:
    description: projected items for all in one
      resources secrets, configmaps, and downward
      API
    properties:
      defaultMode:
        description: |-
          defaultMode are the mode bits used to set permissions on created files by
default.
          Must be an octal value between 0000 and 0777 or a decimal value between 0
and 511.
          YAML accepts both octal and decimal values, JSON requires decimal values
for mode bits.
          Directories within the path are not affected by this setting.
          This might be in conflict with other options that affect the file
          mode, like fsGroup, and the result can be other mode bits set.
        format: int32
```

```
                type: integer
              sources:
                description: sources is the list of volume
                  projections
                items:
                  description: Projection that may be projected
                    along with other supported volume types
                  properties:
                    configMap:
                      description: configMap information
                        about the configMap data to project
                      properties:
                        items:
                          description: |-
                            items if unspecified, each key-value pair in the Data field of the
referenced
                            ConfigMap will be projected into the volume as a file whose name is the
                            key and content is the value. If specified, the listed keys will be
                            projected into the specified paths, and unlisted keys will not be
                            present. If a key is specified which is not present in the ConfigMap,
                            the volume setup will error unless it is marked optional. Paths must be
                            relative and may not contain the '..' path or start with '..'.
                          items:
                            description: Maps a string key
                              to a path within a volume.
                            properties:
                              key:
                                description: key is the
                                  key to project.
                                type: string
                              mode:
                                description: |-
                                  mode is Optional: mode bits used to set permissions on this file.
                                  Must be an octal value between 0000 and 0777 or a decimal value
between 0 and 511.
                                  YAML accepts both octal and decimal values, JSON requires
decimal values for mode bits.
                                  If not specified, the volume defaultMode will be used.
                                  This might be in conflict with other options that affect the file
                                  mode, like fsGroup, and the result can be other mode bits set.
                                format: int32
                                type: integer
                              path:
                                description: |-
                                  path is the relative path of the file to map the key to.
                                  May not be an absolute path.
                                  May not contain the path element '..'.
                                  May not start with the string '..'.
                                type: string
                            required:
                            - key
                            - path
                            type: object
                          type: array
                        name:
                          description: |-
```

```
                                    Name of the referent.
                                    More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                                    TODO: Add other useful fields. apiVersion, kind, uid?
                                  type: string
                                optional:
                                  description: optional specify
                                    whether the ConfigMap or its
                                    keys must be defined
                                  type: boolean
                              type: object
                              x-kubernetes-map-type: atomic
                            downwardAPI:
                              description: downwardAPI information
                                about the downwardAPI data to project
                              properties:
                                items:
                                  description: Items is a list of
                                    DownwardAPIVolume file
                                  items:
                                    description: DownwardAPIVolumeFile
                                      represents information to
                                      create the file containing
                                      the pod field
                                    properties:
                                      fieldRef:
                                        description: 'Required:
                                          Selects a field of the
                                          pod: only annotations,
                                          labels, name and namespace
                                          are supported.'
                                        properties:
                                          apiVersion:
                                            description: Version
                                              of the schema the
                                              FieldPath is written
                                              in terms of, defaults
                                              to "v1".
                                            type: string
                                          fieldPath:
                                            description: Path of
                                              the field to select
                                              in the specified API
                                              version.
                                            type: string
                                        required:
                                        - fieldPath
                                        type: object
                                        x-kubernetes-map-type: atomic
                                      mode:
                                        description: |-
                                          Optional: mode bits used to set permissions on this file, must be an
octal value
                                          between 0000 and 0777 or a decimal value between 0 and 511.
                                          YAML accepts both octal and decimal values, JSON requires
decimal values for mode bits.
```

```
                                    If not specified, the volume defaultMode will be used.
                                    This might be in conflict with other options that affect the file
                                    mode, like fsGroup, and the result can be other mode bits set.
                                  format: int32
                                  type: integer
                                path:
                                  description: 'Required:
                                    Path is  the relative
                                    path name of the file
                                    to be created. Must not
                                    be absolute or contain
                                    the ".." path. Must
                                    be utf-8 encoded. The
                                    first item of the relative
                                    path must not start with
                                    ".."'
                                  type: string
                                resourceFieldRef:
                                  description: |-
                                    Selects a resource of the container: only resources limits and
requests
                                    (limits.cpu, limits.memory, requests.cpu and requests.memory) are
currently supported.
                                  properties:
                                    containerName:
                                      description: 'Container
                                        name: required for
                                        volumes, optional
                                        for env vars'
                                      type: string
                                    divisor:
                                      anyOf:
                                      - type: integer
                                      - type: string
                                      description: Specifies
                                        the output format
                                        of the exposed resources,
                                        defaults to "1"
                                      pattern: ^(\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-
9]+))(([KMGTPE]i)|[numkMGTPE]|([eE](\+|-)?(([0-9]+(\.[0-9]*)?)|(\.[0-9]+))))?$
                                      x-kubernetes-int-or-string: true
                                    resource:
                                      description: 'Required:
                                        resource to select'
                                      type: string
                                  required:
                                  - resource
                                  type: object
                                  x-kubernetes-map-type: atomic
                              required:
                              - path
                              type: object
                            type: array
                          type: object
                        secret:
                          description: secret information about
```

```
                the secret data to project
              properties:
                items:
                  description: |-
                    items if unspecified, each key-value pair in the Data field of the
referenced
                    Secret will be projected into the volume as a file whose name is the
                    key and content is the value. If specified, the listed keys will be
                    projected into the specified paths, and unlisted keys will not be
                    present. If a key is specified which is not present in the Secret,
                    the volume setup will error unless it is marked optional. Paths must be
                    relative and may not contain the '..' path or start with '..'.
                  items:
                    description: Maps a string key
                      to a path within a volume.
                    properties:
                      key:
                        description: key is the
                          key to project.
                        type: string
                      mode:
                        description: |-
                          mode is Optional: mode bits used to set permissions on this file.
                          Must be an octal value between 0000 and 0777 or a decimal value
between 0 and 511.
                          YAML accepts both octal and decimal values, JSON requires
decimal values for mode bits.
                          If not specified, the volume defaultMode will be used.
                          This might be in conflict with other options that affect the file
                          mode, like fsGroup, and the result can be other mode bits set.
                        format: int32
                        type: integer
                      path:
                        description: |-
                          path is the relative path of the file to map the key to.
                          May not be an absolute path.
                          May not contain the path element '..'.
                          May not start with the string '..'.
                        type: string
                    required:
                    - key
                    - path
                    type: object
                  type: array
                name:
                  description: |-
                    Name of the referent.
                    More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                    TODO: Add other useful fields. apiVersion, kind, uid?
                  type: string
                optional:
                  description: optional field specify
                    whether the Secret or its key
                    must be defined
                  type: boolean
```

```
                type: object
                x-kubernetes-map-type: atomic
              serviceAccountToken:
                description: serviceAccountToken is
                  information about the serviceAccountToken
                  data to project
                properties:
                  audience:
                    description: |-
                      audience is the intended audience of the token. A recipient of a token
                      must identify itself with an identifier specified in the audience of the
                      token, and otherwise should reject the token. The audience defaults to
the
                      identifier of the apiserver.
                    type: string
                  expirationSeconds:
                    description: |-
                      expirationSeconds is the requested duration of validity of the service
                      account token. As the token approaches expiration, the kubelet volume
                      plugin will proactively rotate the service account token. The kubelet will
                      start trying to rotate the token if the token is older than 80 percent of
                      its time to live or if the token is older than 24 hours.Defaults to 1 hour
                      and must be at least 10 minutes.
                    format: int64
                    type: integer
                  path:
                    description: |-
                      path is the path relative to the mount point of the file to project the
                      token into.
                    type: string
                required:
                - path
                type: object
            type: object
          type: array
        type: object
      quobyte:
        description: quobyte represents a Quobyte mount
          on the host that shares a pod's lifetime
        properties:
          group:
            description: |-
              group to map volume access to
              Default is no group
            type: string
          readOnly:
            description: |-
              readOnly here will force the Quobyte volume to be mounted with read-only
permissions.
              Defaults to false.
            type: boolean
          registry:
            description: |-
              registry represents a single or multiple Quobyte Registry services
              specified as a string as host:port pair (multiple entries are separated with
commas)
```

```
                which acts as the central registry for volumes
              type: string
            tenant:
              description: |-
                tenant owning the given Quobyte volume in the Backend
                Used with dynamically provisioned Quobyte volumes, value is set by the
plugin
              type: string
            user:
              description: |-
                user to map volume access to
                Defaults to serivceaccount user
              type: string
            volume:
              description: volume is a string that references
                an already created Quobyte volume by name.
              type: string
          required:
          - registry
          - volume
          type: object
        rbd:
          description: |-
            rbd represents a Rados Block Device mount on the host that shares a pod's
lifetime.
            More info: https://examples.k8s.io/volumes/rbd/README.md
          properties:
            fsType:
              description: |-
                fsType is the filesystem type of the volume that you want to mount.
                Tip: Ensure that the filesystem type is supported by the host operating
system.
                Examples: "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
                More info: https://kubernetes.io/docs/concepts/storage/volumes#rbd
                TODO: how do we prevent errors in the filesystem from compromising the
machine
              type: string
            image:
              description: |-
                image is the rados image name.
                More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it
              type: string
            keyring:
              description: |-
                keyring is the path to key ring for RBDUser.
                Default is /etc/ceph/keyring.
                More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it
              type: string
            monitors:
              description: |-
                monitors is a collection of Ceph monitors.
                More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it
              items:
                type: string
              type: array
            pool:
```

```
                  description: |-
                    pool is the rados pool name.
                    Default is rbd.
                    More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it
                  type: string
                readOnly:
                  description: |-
                    readOnly here will force the ReadOnly setting in VolumeMounts.
                    Defaults to false.
                    More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it
                  type: boolean
                secretRef:
                  description: |-
                    secretRef is name of the authentication secret for RBDUser. If provided
                    overrides keyring.
                    Default is nil.
                    More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it
                  properties:
                    name:
                      description: |-
                        Name of the referent.
                        More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                        TODO: Add other useful fields. apiVersion, kind, uid?
                      type: string
                  type: object
                  x-kubernetes-map-type: atomic
                user:
                  description: |-
                    user is the rados user name.
                    Default is admin.
                    More info: https://examples.k8s.io/volumes/rbd/README.md#how-to-use-it
                  type: string
              required:
              - image
              - monitors
              type: object
            scaleIO:
              description: scaleIO represents a ScaleIO persistent
                volume attached and mounted on Kubernetes
                nodes.
              properties:
                fsType:
                  description: |-
                    fsType is the filesystem type to mount.
                    Must be a filesystem type supported by the host operating system.
                    Ex. "ext4", "xfs", "ntfs".
                    Default is "xfs".
                  type: string
                gateway:
                  description: gateway is the host address
                    of the ScaleIO API Gateway.
                  type: string
                protectionDomain:
                  description: protectionDomain is the name
                    of the ScaleIO Protection Domain for the
```

```
                    configured storage.
                  type: string
                readOnly:
                  description: |-
                    readOnly Defaults to false (read/write). ReadOnly here will force
                    the ReadOnly setting in VolumeMounts.
                  type: boolean
                secretRef:
                  description: |-
                    secretRef references to the secret for ScaleIO user and other
                    sensitive information. If this is not provided, Login operation will fail.
                  properties:
                    name:
                      description: |-
                        Name of the referent.
                        More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                        TODO: Add other useful fields. apiVersion, kind, uid?
                      type: string
                  type: object
                  x-kubernetes-map-type: atomic
                sslEnabled:
                  description: sslEnabled Flag enable/disable
                    SSL communication with Gateway, default
                    false
                  type: boolean
                storageMode:
                  description: |-
                    storageMode indicates whether the storage for a volume should be
ThickProvisioned or ThinProvisioned.
                    Default is ThinProvisioned.
                  type: string
                storagePool:
                  description: storagePool is the ScaleIO
                    Storage Pool associated with the protection
                    domain.
                  type: string
                system:
                  description: system is the name of the storage
                    system as configured in ScaleIO.
                  type: string
                volumeName:
                  description: |-
                    volumeName is the name of a volume already created in the ScaleIO system
                    that is associated with this volume source.
                  type: string
              required:
              - gateway
              - secretRef
              - system
              type: object
            secret:
              description: |-
                secret represents a secret that should populate this volume.
                More info: https://kubernetes.io/docs/concepts/storage/volumes#secret
              properties:
```

```
                defaultMode:
                  description: |-
                    defaultMode is Optional: mode bits used to set permissions on created files by
default.
                    Must be an octal value between 0000 and 0777 or a decimal value between 0
and 511.
                    YAML accepts both octal and decimal values, JSON requires decimal values
                    for mode bits. Defaults to 0644.
                    Directories within the path are not affected by this setting.
                    This might be in conflict with other options that affect the file
                    mode, like fsGroup, and the result can be other mode bits set.
                  format: int32
                  type: integer
                items:
                  description: |-
                    items If unspecified, each key-value pair in the Data field of the referenced
                    Secret will be projected into the volume as a file whose name is the
                    key and content is the value. If specified, the listed keys will be
                    projected into the specified paths, and unlisted keys will not be
                    present. If a key is specified which is not present in the Secret,
                    the volume setup will error unless it is marked optional. Paths must be
                    relative and may not contain the '..' path or start with '..'.
                  items:
                    description: Maps a string key to a path
                      within a volume.
                    properties:
                      key:
                        description: key is the key to project.
                        type: string
                      mode:
                        description: |-
                          mode is Optional: mode bits used to set permissions on this file.
                          Must be an octal value between 0000 and 0777 or a decimal value
between 0 and 511.
                          YAML accepts both octal and decimal values, JSON requires decimal
values for mode bits.
                          If not specified, the volume defaultMode will be used.
                          This might be in conflict with other options that affect the file
                          mode, like fsGroup, and the result can be other mode bits set.
                        format: int32
                        type: integer
                      path:
                        description: |-
                          path is the relative path of the file to map the key to.
                          May not be an absolute path.
                          May not contain the path element '..'.
                          May not start with the string '..'.
                        type: string
                    required:
                    - key
                    - path
                    type: object
                  type: array
                optional:
                  description: optional field specify whether
                    the Secret or its keys must be defined
```

```
                type: boolean
              secretName:
                description: |-
                  secretName is the name of the secret in the pod's namespace to use.
                  More info: https://kubernetes.io/docs/concepts/storage/volumes#secret
                type: string
            type: object
          storageos:
            description: storageOS represents a StorageOS
              volume attached and mounted on Kubernetes
              nodes.
            properties:
              fsType:
                description: |-
                  fsType is the filesystem type to mount.
                  Must be a filesystem type supported by the host operating system.
                  Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
                type: string
              readOnly:
                description: |-
                  readOnly defaults to false (read/write). ReadOnly here will force
                  the ReadOnly setting in VolumeMounts.
                type: boolean
              secretRef:
                description: |-
                  secretRef specifies the secret to use for obtaining the StorageOS API
                  credentials.  If not specified, default values will be attempted.
                properties:
                  name:
                    description: |-
                      Name of the referent.
                      More info: https://kubernetes.io/docs/concepts/overview/working-with-
objects/names/#names
                      TODO: Add other useful fields. apiVersion, kind, uid?
                    type: string
                type: object
                x-kubernetes-map-type: atomic
              volumeName:
                description: |-
                  volumeName is the human-readable name of the StorageOS volume.
Volume
                  names are only unique within a namespace.
                type: string
              volumeNamespace:
                description: |-
                  volumeNamespace specifies the scope of the volume within StorageOS.  If no
                  namespace is specified then the Pod's namespace will be used.  This allows
the
                  Kubernetes name scoping to be mirrored within StorageOS for tighter
integration.
                  Set VolumeName to any name to override the default behaviour.
                  Set to "default" if you are not using namespaces within StorageOS.
                  Namespaces that do not pre-exist within StorageOS will be created.
                type: string
            type: object
          vsphereVolume:
```

```
                description: vsphereVolume represents a vSphere
                  volume attached and mounted on kubelets host
                  machine
                properties:
                  fsType:
                    description: |-
                      fsType is filesystem type to mount.
                      Must be a filesystem type supported by the host operating system.
                      Ex. "ext4", "xfs", "ntfs". Implicitly inferred to be "ext4" if unspecified.
                    type: string
                  storagePolicyID:
                    description: storagePolicyID is the storage
                      Policy Based Management (SPBM) profile
                      ID associated with the StoragePolicyName.
                    type: string
                  storagePolicyName:
                    description: storagePolicyName is the storage
                      Policy Based Management (SPBM) profile
                      name.
                    type: string
                  volumePath:
                    description: volumePath is the path that
                      identifies vSphere volume vmdk
                    type: string
                required:
                - volumePath
                type: object
            required:
            - name
            type: object
          type: array
        required:
        - containers
        type: object
      type: object
    ttlSecondsAfterFinished:
      format: int32
      type: integer
  required:
  - template
  type: object
type: object
schedule:
  minLength: 0
  type: string
successfulJobsHistoryLimit:
  format: int32
  minimum: 0
  type: integer
required:
- jobTemplate
- schedule
type: object
status:
properties:
  active:
```

```
            type: string
          datacenter:
            type: string
        type: object
      type: object
  served: true
  storage: true
  subresources:
    status: {}
```

05-namespace.yaml

```
apiVersion: v1
kind: Namespace
metadata:
  name: kubecronic #название пространства имен в kubernetes для установки оператора
```

10-configmap.yaml

```
apiVersion: v1
kind: ConfigMap
metadata:
  annotations:
    project/owner: kubecronic
  name: kubecronic
  namespace: kubecronic #название пространства имен в kubernetes для установки оператора
data:
  ENV: production/dc1 #окружение для установки kubecronic. По умолчанию - production + имя кластера
  KUBECRONIC_DATACENTER: dc1 #название кластера. Должно быть уникальным
  LOG_LEVEL: info
  REDIS_ADDR: 10.10.10.10:6379 #адрес и порт экземпляра redis
  SENTRY_DSN: https://qwerty@sentry.some.domain/1
  SIGNAL_CONSUL_ADDRESSES: consul:8500
  SIGNAL_CONSUL_KEY: service/maintenance
immutable: false
```

10-secret.yaml

```
apiVersion: v1
kind: Secret
metadata:
  name: kubecronic-secret
  namespace: kubecronic #название пространства имен в kubernetes для установки оператора
type: Opaque
stringData:
  REDIS_PASSWORD: "password" #пароль для доступа к экземпляру redis
  SIGNAL_CONSUL_TOKEN: "token"
```

20-rbac.yaml

```
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: vseinstrumenti:kubecronic-controller
rules:
  - apiGroups:
```

```
    - kubecronic.vseinstrumenti.ru
  resources:
    - cronjobs
    - cronjobs/status
  verbs:
    - get
    - list
    - watch
    - create
    - delete
    - deletecollection
    - patch
    - update
- apiGroups:
    - batch
  resources:
    - jobs
  verbs:
    - get
    - list
    - watch
    - create
    - delete
    - deletecollection
    - patch
    - update
- apiGroups:
    - ""
  resources:
    - pods
  verbs:
    - get
    - list
    - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: vseinstrumenti:kubecronic-controller
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: vseinstrumenti:kubecronic-controller
subjects:
  - kind: ServiceAccount
    name: default
    namespace: kubecronic #название пространства имен в kubernetes для установки оператора
```

50-deployment.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  annotations:
    project/owner: kubecronic
```

```
  name: kubecronic
  namespace: kubecronic #название пространства имен в kubernetes для установки оператора
spec:
  progressDeadlineSeconds: 600
  replicas: 1
  revisionHistoryLimit: 2
  selector:
    matchLabels:
      app: kubecronic
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 50%
    type: RollingUpdate
  template:
    metadata:
      labels:
        app: kubecronic
    spec:
      automountServiceAccountToken: true
      containers:
      - command:
        - /usr/bin/kubecronic
        envFrom:
        - secretRef:
            name: kubecronic-secret
            optional: false
        - configMapRef:
            name: kubecronic
            optional: false
        image: kubecronic:latest #путь до docker-registry с образом kubecronic
        imagePullPolicy: IfNotPresent
        name: app
        ports:
        - containerPort: 9000
          name: http
          protocol: TCP
        resources:
          limits:
            cpu: 400m
            memory: 512Mi
          requests:
            cpu: 10m
            memory: 512Mi
      restartPolicy: Always
      shareProcessNamespace: false
      terminationGracePeriodSeconds: 30
```

Прменить полученные манифесты во всех кластерах kubernetes командой **kubectl apply -f '*.yaml'**