Feature Gates (removed)

This page contains list of feature gates that have been removed. The information on this page is for reference. A removed feature gate is different from a GA'ed or deprecated one in that a removed one is no longer recognized as a valid feature gate. However, a GA'ed or a deprecated feature gate is still recognized by the corresponding Kubernetes components although they are unable to cause any behavior differences in a cluster.

For feature gates that are still recognized by the Kubernetes components, please refer to the Alpha/Beta feature gate table or the Graduated/Deprecated feature gate table

Feature gates that are removed

In the following table:

- The "From" column contains the Kubernetes release when a feature is introduced or its release stage is changed.
- The "To" column, if not empty, contains the last Kubernetes release in which you can still use a feature gate. If the feature stage is either "Deprecated" or "GA", the "To" column is the Kubernetes release when the feature is removed.

Feature	Default	Stage	From	To
`Accelerators`	`false`	Alpha	1.6	1.10
`Accelerators`	–	Deprecated	1.11	1.11
`AdmissionWebhookMatchConditions`	`false`	Alpha	1.27	1.27
`AdmissionWebhookMatchConditions`	`true`	Beta	1.28	1.29
`AdmissionWebhookMatchConditions`	`true`	GA	1.30	1.32
`AdvancedAuditing`	`false`	Alpha	1.7	1.7
`AdvancedAuditing`	`true`	Beta	1.8	1.11
`AdvancedAuditing`	`true`	GA	1.12	1.27
`AffinityInAnnotations`	`false`	Alpha	1.6	1.7
`AffinityInAnnotations`	–	Deprecated	1.8	1.8
`AggregatedDiscoveryEndpoint`	`false`	Alpha	1.26	1.26
`AggregatedDiscoveryEndpoint`	`true`	Beta	1.27	1.29
`AggregatedDiscoveryEndpoint`	`true`	GA	1.30	1.32
`AllowExtTrafficLocalEndpoints`	`false`	Beta	1.4	1.6
`AllowExtTrafficLocalEndpoints`	`true`	GA	1.7	1.9
`AllowInsecureBackendProxy`	`true`	Beta	1.17	1.20
`AllowInsecureBackendProxy`	`true`	GA	1.21	1.25
`APIListChunking`	`false`	Alpha	1.8	1.8
`APIListChunking`	`true`	Beta	1.9	1.28
`APIListChunking`	`true`	GA	1.29	1.32
`APIPriorityAndFairness`	`false`	Alpha	1.18	1.19
`APIPriorityAndFairness`	`true`	Beta	1.20	1.28
`APIPriorityAndFairness`	`true`	GA	1.29	1.30
`APISelfSubjectReview`	`false`	Alpha	1.26	1.26
`APISelfSubjectReview`	`true`	Beta	1.27	1.27
`APISelfSubjectReview`	`true`	GA	1.28	1.29
`AppArmor`	`true`	Beta	1.4	1.30
`AppArmor`	`true`	GA	1.31	1.32
`AppArmorFields`	`true`	Beta	1.30	1.30
`AppArmorFields`	`true`	GA	1.31	1.32
`AttachVolumeLimit`	`false`	Alpha	1.11	1.11
`AttachVolumeLimit`	`true`	Beta	1.12	1.16
`AttachVolumeLimit`	`true`	GA	1.17	1.21
`BalanceAttachedNodeVolumes`	`false`	Alpha	1.11	1.21
`BalanceAttachedNodeVolumes`	`false`	Deprecated	1.22	1.22
`BlockVolume`	`false`	Alpha	1.9	1.12
`BlockVolume`	`true`	Beta	1.13	1.17
`BlockVolume`	`true`	GA	1.18	1.21
`BoundServiceAccountTokenVolume`	`false`	Alpha	1.13	1.20
`BoundServiceAccountTokenVolume`	`true`	Beta	1.21	1.21
`BoundServiceAccountTokenVolume`	`true`	GA	1.22	1.23

Feature	Default	Stage	From	To
CloudDualStackNodeIPs	false	Alpha	1.27	1.28
CloudDualStackNodeIPs	true	Beta	1.29	1.29
CloudDualStackNodeIPs	true	GA	1.30	1.31
ConfigurableFSGroupPolicy	false	Alpha	1.18	1.19
ConfigurableFSGroupPolicy	true	Beta	1.20	1.22
ConfigurableFSGroupPolicy	true	GA	1.23	1.25
ConsistentHTTPGetHandlers	true	GA	1.25	1.30
ControllerManagerLeaderMigration	false	Alpha	1.21	1.21
ControllerManagerLeaderMigration	true	Beta	1.22	1.23
ControllerManagerLeaderMigration	true	GA	1.24	1.26
CPUManager	false	Alpha	1.8	1.9
CPUManager	true	Beta	1.10	1.25
CPUManager	true	GA	1.26	1.32
CRIContainerLogRotation	false	Alpha	1.10	1.10
CRIContainerLogRotation	true	Beta	1.11	1.20
CRIContainerLogRotation	true	GA	1.21	1.22
CronJobControllerV2	false	Alpha	1.20	1.20
CronJobControllerV2	true	Beta	1.21	1.21
CronJobControllerV2	true	GA	1.22	1.23
CronJobTimeZone	false	Alpha	1.24	1.24
CronJobTimeZone	true	Beta	1.25	1.26
CronJobTimeZone	true	GA	1.27	1.28
CSIBlockVolume	false	Alpha	1.11	1.13
CSIBlockVolume	true	Beta	1.14	1.17
CSIBlockVolume	true	GA	1.18	1.21
CSIDriverRegistry	false	Alpha	1.12	1.13
CSIDriverRegistry	true	Beta	1.14	1.17
CSIDriverRegistry	true	GA	1.18	1.21
CSIInlineVolume	false	Alpha	1.15	1.15
CSIInlineVolume	true	Beta	1.16	1.24
CSIInlineVolume	true	GA	1.25	1.26
CSIMigration	false	Alpha	1.14	1.16
CSIMigration	true	Beta	1.17	1.24
CSIMigration	true	GA	1.25	1.26
CSIMigrationAWS	false	Alpha	1.14	1.16
CSIMigrationAWS	false	Beta	1.17	1.22
CSIMigrationAWS	true	Beta	1.23	1.24
CSIMigrationAWS	true	GA	1.25	1.26
CSIMigrationAWSComplete	false	Alpha	1.17	1.20
CSIMigrationAWSComplete	–	Deprecated	1.21	1.21
CSIMigrationAzureDisk	false	Alpha	1.15	1.18
CSIMigrationAzureDisk	false	Beta	1.19	1.22
CSIMigrationAzureDisk	true	Beta	1.23	1.23
CSIMigrationAzureDisk	true	GA	1.24	1.26
CSIMigrationAzureDiskComplete	false	Alpha	1.17	1.20
CSIMigrationAzureDiskComplete	–	Deprecated	1.21	1.21
CSIMigrationAzureFile	false	Alpha	1.15	1.20
CSIMigrationAzureFile	false	Beta	1.21	1.23
CSIMigrationAzureFile	true	Beta	1.24	1.25
CSIMigrationAzureFile	true	GA	1.26	1.29
CSIMigrationAzureFileComplete	false	Alpha	1.17	1.20
CSIMigrationAzureFileComplete	–	Deprecated	1.21	1.21
CSIMigrationGCE	false	Alpha	1.14	1.16
CSIMigrationGCE	false	Beta	1.17	1.22
CSIMigrationGCE	true	Beta	1.23	1.24
CSIMigrationGCE	true	GA	1.25	1.27
CSIMigrationGCEComplete	false	Alpha	1.17	1.20
CSIMigrationGCEComplete	–	Deprecated	1.21	1.21

Feature	Default	Stage	From	To
CSIMigrationOpenStack	false	Alpha	1.14	1.17
CSIMigrationOpenStack	true	Beta	1.18	1.23
CSIMigrationOpenStack	true	GA	1.24	1.25
CSIMigrationOpenStackComplete	false	Alpha	1.17	1.20
CSIMigrationOpenStackComplete	–	Deprecated	1.21	1.21
CSIMigrationRBD	false	Alpha	1.23	1.27
CSIMigrationRBD	false	Deprecated	1.28	1.30
CSIMigrationvSphere	false	Alpha	1.18	1.18
CSIMigrationvSphere	false	Beta	1.19	1.24
CSIMigrationvSphere	true	Beta	1.25	1.25
CSIMigrationvSphere	true	GA	1.26	1.28
CSIMigrationvSphereComplete	false	Beta	1.19	1.21
CSIMigrationvSphereComplete	–	Deprecated	1.22	1.22
CSINodeExpandSecret	false	Alpha	1.25	1.26
CSINodeExpandSecret	true	Beta	1.27	1.28
CSINodeExpandSecret	true	GA	1.29	1.30
CSINodeInfo	false	Alpha	1.12	1.13
CSINodeInfo	true	Beta	1.14	1.16
CSINodeInfo	true	GA	1.17	1.22
CSIPersistentVolume	false	Alpha	1.9	1.9
CSIPersistentVolume	true	Beta	1.10	1.12
CSIPersistentVolume	true	GA	1.13	1.16
CSIServiceAccountToken	false	Alpha	1.20	1.20
CSIServiceAccountToken	true	Beta	1.21	1.21
CSIServiceAccountToken	true	GA	1.22	1.24
CSIStorageCapacity	false	Alpha	1.19	1.20
CSIStorageCapacity	true	Beta	1.21	1.23
CSIStorageCapacity	true	GA	1.24	1.27
CSIVolumeFSGroupPolicy	false	Alpha	1.19	1.19
CSIVolumeFSGroupPolicy	true	Beta	1.20	1.22
CSIVolumeFSGroupPolicy	true	GA	1.23	1.25
CSRDuration	true	Beta	1.22	1.23
CSRDuration	true	GA	1.24	1.25
CustomPodDNS	false	Alpha	1.9	1.9
CustomPodDNS	true	Beta	1.10	1.13
CustomPodDNS	true	GA	1.14	1.16
CustomResourceDefaulting	false	Alpha	1.15	1.15
CustomResourceDefaulting	true	Beta	1.16	1.16
CustomResourceDefaulting	true	GA	1.17	1.18
CustomResourcePublishOpenAPI	false	Alpha	1.14	1.14
CustomResourcePublishOpenAPI	true	Beta	1.15	1.15
CustomResourcePublishOpenAPI	true	GA	1.16	1.18
CustomResourceSubresources	false	Alpha	1.10	1.10
CustomResourceSubresources	true	Beta	1.11	1.15
CustomResourceSubresources	true	GA	1.16	1.18
CustomResourceValidation	false	Alpha	1.8	1.8
CustomResourceValidation	true	Beta	1.9	1.15
CustomResourceValidation	true	GA	1.16	1.18
CustomResourceValidationExpressions	false	Alpha	1.23	1.24
CustomResourceValidationExpressions	true	Beta	1.25	1.28
CustomResourceValidationExpressions	true	GA	1.29	1.30
CustomResourceWebhookConversion	false	Alpha	1.13	1.14
CustomResourceWebhookConversion	true	Beta	1.15	1.15
CustomResourceWebhookConversion	true	GA	1.16	1.18
DaemonSetUpdateSurge	false	Alpha	1.21	1.21
DaemonSetUpdateSurge	true	Beta	1.22	1.24
DaemonSetUpdateSurge	true	GA	1.25	1.26
DefaultHostNetworkHostPortsInPodTemplates	false	Deprecated	1.28	1.30

Feature	Default	Stage	From	To
DefaultPodTopologySpread	false	Alpha	1.19	1.19
DefaultPodTopologySpread	true	Beta	1.20	1.23
DefaultPodTopologySpread	true	GA	1.24	1.25
DelegateFSGroupToCSIDriver	false	Alpha	1.22	1.22
DelegateFSGroupToCSIDriver	true	Beta	1.23	1.25
DelegateFSGroupToCSIDriver	true	GA	1.26	1.27
DevicePluginCDIDevices	false	Alpha	1.28	1.28
DevicePluginCDIDevices	true	Beta	1.29	1.30
DevicePluginCDIDevices	true	GA	1.31	1.33
DevicePlugins	false	Alpha	1.8	1.9
DevicePlugins	true	Beta	1.10	1.25
DevicePlugins	true	GA	1.26	1.27
DisableAcceleratorUsageMetrics	false	Alpha	1.19	1.19
DisableAcceleratorUsageMetrics	true	Beta	1.20	1.24
DisableAcceleratorUsageMetrics	true	GA	1.25	1.27
DisableCloudProviders	false	Alpha	1.22	1.28
DisableCloudProviders	true	Beta	1.29	1.30
DisableCloudProviders	true	GA	1.31	1.32
DisableKubeletCloudCredentialProviders	false	Alpha	1.23	1.28
DisableKubeletCloudCredentialProviders	true	Beta	1.29	1.30
DisableKubeletCloudCredentialProviders	true	GA	1.31	1.32
DownwardAPIHugePages	false	Alpha	1.20	1.20
DownwardAPIHugePages	false	Beta	1.21	1.21
DownwardAPIHugePages	true	Beta	1.22	1.26
DownwardAPIHugePages	true	GA	1.27	1.28
DRAControlPlaneController	false	Alpha	1.26	1.31
DryRun	false	Alpha	1.12	1.12
DryRun	true	Beta	1.13	1.18
DryRun	true	GA	1.19	1.27
DynamicAuditing	false	Alpha	1.13	1.18
DynamicAuditing	–	Deprecated	1.19	1.19
DynamicKubeletConfig	false	Alpha	1.4	1.10
DynamicKubeletConfig	true	Beta	1.11	1.21
DynamicKubeletConfig	false	Deprecated	1.22	1.25
DynamicProvisioningScheduling	false	Alpha	1.11	1.11
DynamicProvisioningScheduling	–	Deprecated	1.12	–
DynamicVolumeProvisioning	true	Alpha	1.3	1.7
DynamicVolumeProvisioning	true	GA	1.8	1.12
EfficientWatchResumption	false	Alpha	1.20	1.20
EfficientWatchResumption	true	Beta	1.21	1.23
EfficientWatchResumption	true	GA	1.24	1.32
EnableAggregatedDiscoveryTimeout	true	Deprecated	1.16	1.17
EnableEquivalenceClassCache	false	Alpha	1.8	1.12
EnableEquivalenceClassCache	–	Deprecated	1.13	1.23
EndpointSlice	false	Alpha	1.16	1.16
EndpointSlice	false	Beta	1.17	1.17
EndpointSlice	true	Beta	1.18	1.20
EndpointSlice	true	GA	1.21	1.24
EndpointSliceNodeName	false	Alpha	1.20	1.20
EndpointSliceNodeName	true	GA	1.21	1.24
EndpointSliceProxying	false	Alpha	1.18	1.18
EndpointSliceProxying	true	Beta	1.19	1.21
EndpointSliceProxying	true	GA	1.22	1.24
EndpointSliceTerminatingCondition	false	Alpha	1.20	1.21
EndpointSliceTerminatingCondition	true	Beta	1.22	1.25
EndpointSliceTerminatingCondition	true	GA	1.26	1.27
EphemeralContainers	false	Alpha	1.16	1.22
EphemeralContainers	true	Beta	1.23	1.24

Feature	Default	Stage	From	To
EphemeralContainers	true	GA	1.25	1.26
EvenPodsSpread	false	Alpha	1.16	1.17
EvenPodsSpread	true	Beta	1.18	1.18
EvenPodsSpread	true	GA	1.19	1.21
ExpandCSIVolumes	false	Alpha	1.14	1.15
ExpandCSIVolumes	true	Beta	1.16	1.23
ExpandCSIVolumes	true	GA	1.24	1.26
ExpandedDNSConfig	false	Alpha	1.22	1.25
ExpandedDNSConfig	true	Beta	1.26	1.27
ExpandedDNSConfig	true	GA	1.28	1.29
ExpandInUsePersistentVolumes	false	Alpha	1.11	1.14
ExpandInUsePersistentVolumes	true	Beta	1.15	1.23
ExpandInUsePersistentVolumes	true	GA	1.24	1.26
ExpandPersistentVolumes	false	Alpha	1.8	1.10
ExpandPersistentVolumes	true	Beta	1.11	1.23
ExpandPersistentVolumes	true	GA	1.24	1.26
ExperimentalCriticalPodAnnotation	false	Alpha	1.5	1.12
ExperimentalCriticalPodAnnotation	false	Deprecated	1.13	1.16
ExperimentalHostUserNamespaceDefaulting	false	Beta	1.5	1.27
ExperimentalHostUserNamespaceDefaulting	false	Deprecated	1.28	1.29
ExternalPolicyForExternalIP	true	GA	1.18	1.22
GCERegionalPersistentDisk	true	Beta	1.10	1.12
GCERegionalPersistentDisk	true	GA	1.13	1.16
GenericEphemeralVolume	false	Alpha	1.19	1.20
GenericEphemeralVolume	true	Beta	1.21	1.22
GenericEphemeralVolume	true	GA	1.23	1.24
GRPCContainerProbe	false	Alpha	1.23	1.23
GRPCContainerProbe	true	Beta	1.24	1.26
GRPCContainerProbe	true	GA	1.27	1.28
HPAContainerMetrics	false	Alpha	1.20	1.26
HPAContainerMetrics	true	Beta	1.27	1.29
HPAContainerMetrics	true	GA	1.30	1.31
HugePages	false	Alpha	1.8	1.9
HugePages	true	Beta	1.10	1.13
HugePages	true	GA	1.14	1.16
HugePageStorageMediumSize	false	Alpha	1.18	1.18
HugePageStorageMediumSize	true	Beta	1.19	1.21
HugePageStorageMediumSize	true	GA	1.22	1.24
HyperVContainer	false	Alpha	1.10	1.19
HyperVContainer	false	Deprecated	1.20	1.20
IdentifyPodOS	false	Alpha	1.23	1.23
IdentifyPodOS	true	Beta	1.24	1.24
IdentifyPodOS	true	GA	1.25	1.26
ImmutableEphemeralVolumes	false	Alpha	1.18	1.18
ImmutableEphemeralVolumes	true	Beta	1.19	1.20
ImmutableEphemeralVolumes	true	GA	1.21	1.24
IndexedJob	false	Alpha	1.21	1.21
IndexedJob	true	Beta	1.22	1.23
IndexedJob	true	GA	1.24	1.25
IngressClassNamespacedParams	false	Alpha	1.21	1.21
IngressClassNamespacedParams	true	Beta	1.22	1.22
IngressClassNamespacedParams	true	GA	1.23	1.24
Initializers	false	Alpha	1.7	1.13
Initializers	–	Deprecated	1.14	1.14
InTreePluginAWSUnregister	false	Alpha	1.21	1.30
InTreePluginAzureDiskUnregister	false	Alpha	1.21	1.30
InTreePluginAzureFileUnregister	false	Alpha	1.21	1.30
InTreePluginGCEUnregister	false	Alpha	1.21	1.30

Feature	Default	Stage	From	To
InTreePluginOpenStackUnregister	false	Alpha	1.21	1.30
InTreePluginRBDUnregister	false	Alpha	1.23	1.27
InTreePluginRBDUnregister	false	Deprecated	1.28	1.30
InTreePluginvSphereUnregister	false	Alpha	1.21	1.30
IPTablesOwnershipCleanup	false	Alpha	1.25	1.26
IPTablesOwnershipCleanup	true	Beta	1.27	1.27
IPTablesOwnershipCleanup	true	GA	1.28	1.29
IPv6DualStack	false	Alpha	1.15	1.20
IPv6DualStack	true	Beta	1.21	1.22
IPv6DualStack	true	GA	1.23	1.24
JobMutableNodeSchedulingDirectives	true	Beta	1.23	1.26
JobMutableNodeSchedulingDirectives	true	GA	1.27	1.28
JobPodFailurePolicy	false	Alpha	1.25	1.25
JobPodFailurePolicy	true	Beta	1.26	1.30
JobPodFailurePolicy	true	GA	1.31	1.32
JobReadyPods	false	Alpha	1.23	1.23
JobReadyPods	true	Beta	1.24	1.28
JobReadyPods	true	GA	1.29	1.30
JobTrackingWithFinalizers	false	Alpha	1.22	1.22
JobTrackingWithFinalizers	false	Beta	1.23	1.24
JobTrackingWithFinalizers	true	Beta	1.25	1.25
JobTrackingWithFinalizers	true	GA	1.26	1.28
KMSv2	false	Alpha	1.25	1.26
KMSv2	true	Beta	1.27	1.28
KMSv2	true	GA	1.29	1.31
KMSv2KDF	false	Beta	1.28	1.28
KMSv2KDF	true	GA	1.29	1.31
KubeletConfigFile	false	Alpha	1.8	1.9
KubeletConfigFile	–	Deprecated	1.10	1.10
KubeletCredentialProviders	false	Alpha	1.20	1.23
KubeletCredentialProviders	true	Beta	1.24	1.25
KubeletCredentialProviders	true	GA	1.26	1.28
KubeletPluginsWatcher	false	Alpha	1.11	1.11
KubeletPluginsWatcher	true	Beta	1.12	1.12
KubeletPluginsWatcher	true	GA	1.13	1.16
KubeletPodResources	false	Alpha	1.13	1.14
KubeletPodResources	true	Beta	1.15	1.27
KubeletPodResources	true	GA	1.28	1.29
KubeletPodResourcesGetAllocatable	false	Alpha	1.21	1.22
KubeletPodResourcesGetAllocatable	true	Beta	1.23	1.27
KubeletPodResourcesGetAllocatable	true	GA	1.28	1.29
KubeProxyDrainingTerminatingNodes	false	Alpha	1.28	1.30
KubeProxyDrainingTerminatingNodes	true	Beta	1.30	1.30
KubeProxyDrainingTerminatingNodes	true	GA	1.31	1.32
LegacyNodeRoleBehavior	false	Alpha	1.16	1.18
LegacyNodeRoleBehavior	true	Beta	1.19	1.20
LegacyNodeRoleBehavior	false	GA	1.21	1.22
LegacyServiceAccountTokenCleanUp	false	Alpha	1.28	1.28
LegacyServiceAccountTokenCleanUp	true	Beta	1.29	1.29
LegacyServiceAccountTokenCleanUp	true	GA	1.30	1.31
LegacyServiceAccountTokenNoAutoGeneration	true	Beta	1.24	1.25
LegacyServiceAccountTokenNoAutoGeneration	true	GA	1.26	1.28
LegacyServiceAccountTokenTracking	false	Alpha	1.26	1.26
LegacyServiceAccountTokenTracking	true	Beta	1.27	1.27
LegacyServiceAccountTokenTracking	true	GA	1.28	1.29
LocalStorageCapacityIsolation	false	Alpha	1.7	1.9
LocalStorageCapacityIsolation	true	Beta	1.10	1.24
LocalStorageCapacityIsolation	true	GA	1.25	1.26

Feature	Default	Stage	From	To
MinDomainsInPodTopologySpread	false	Alpha	1.24	1.24
MinDomainsInPodTopologySpread	false	Beta	1.25	1.26
MinDomainsInPodTopologySpread	true	Beta	1.27	1.29
MinDomainsInPodTopologySpread	true	GA	1.30	1.31
MinimizeIPTablesRestore	false	Alpha	1.26	1.26
MinimizeIPTablesRestore	true	Beta	1.27	1.27
MinimizeIPTablesRestore	true	GA	1.28	1.29
MixedProtocolLBService	false	Alpha	1.20	1.23
MixedProtocolLBService	true	Beta	1.24	1.25
MixedProtocolLBService	true	GA	1.26	1.27
MountContainers	false	Alpha	1.9	1.16
MountContainers	false	Deprecated	1.17	1.17
MountPropagation	false	Alpha	1.8	1.9
MountPropagation	true	Beta	1.10	1.11
MountPropagation	true	GA	1.12	1.14
MultiCIDRRangeAllocator	false	Alpha	1.25	1.28
NamespaceDefaultLabelName	true	Beta	1.21	1.21
NamespaceDefaultLabelName	true	GA	1.22	1.23
NetworkPolicyEndPort	false	Alpha	1.21	1.21
NetworkPolicyEndPort	true	Beta	1.22	1.24
NetworkPolicyEndPort	true	GA	1.25	1.26
NetworkPolicyStatus	false	Alpha	1.24	1.27
NewVolumeManagerReconstruction	false	Alpha	1.25	1.26
NewVolumeManagerReconstruction	true	Beta	1.27	1.29
NewVolumeManagerReconstruction	true	GA	1.30	1.31
NodeDisruptionExclusion	false	Alpha	1.16	1.18
NodeDisruptionExclusion	true	Beta	1.19	1.20
NodeDisruptionExclusion	true	GA	1.21	1.22
NodeLease	false	Alpha	1.12	1.13
NodeLease	true	Beta	1.14	1.16
NodeLease	true	GA	1.17	1.23
NodeOutOfServiceVolumeDetach	false	Alpha	1.24	1.25
NodeOutOfServiceVolumeDetach	true	Beta	1.26	1.27
NodeOutOfServiceVolumeDetach	true	GA	1.28	1.31
NonPreemptingPriority	false	Alpha	1.15	1.18
NonPreemptingPriority	true	Beta	1.19	1.23
NonPreemptingPriority	true	GA	1.24	1.25
OpenAPIV3	false	Alpha	1.23	1.23
OpenAPIV3	true	Beta	1.24	1.26
OpenAPIV3	true	GA	1.27	1.28
PDBUnhealthyPodEvictionPolicy	false	Alpha	1.26	1.26
PDBUnhealthyPodEvictionPolicy	true	Beta	1.27	1.30
PDBUnhealthyPodEvictionPolicy	true	GA	1.31	1.32
PersistentLocalVolumes	false	Alpha	1.7	1.9
PersistentLocalVolumes	true	Beta	1.10	1.13
PersistentLocalVolumes	true	GA	1.14	1.16
PersistentVolumeLastPhaseTransitionTime	false	Alpha	1.28	1.28
PersistentVolumeLastPhaseTransitionTime	true	Beta	1.29	1.30
PersistentVolumeLastPhaseTransitionTime	true	GA	1.31	1.32
PodAffinityNamespaceSelector	false	Alpha	1.21	1.21
PodAffinityNamespaceSelector	true	Beta	1.22	1.23
PodAffinityNamespaceSelector	true	GA	1.24	1.25
PodDisruptionBudget	false	Alpha	1.3	1.4
PodDisruptionBudget	true	Beta	1.5	1.20
PodDisruptionBudget	true	GA	1.21	1.25
PodDisruptionConditions	false	Alpha	1.25	1.25
PodDisruptionConditions	true	Beta	1.26	1.30
PodDisruptionConditions	true	GA	1.31	1.33

Feature	Default	Stage	From	To
PodHasNetworkCondition	false	Alpha	1.25	1.27
PodHostIPs	false	Alpha	1.28	1.28
PodHostIPs	true	Beta	1.29	1.30
PodHostIPs	true	GA	1.30	1.31
PodOverhead	false	Alpha	1.16	1.17
PodOverhead	true	Beta	1.18	1.23
PodOverhead	true	GA	1.24	1.25
PodPriority	false	Alpha	1.8	1.10
PodPriority	true	Beta	1.11	1.13
PodPriority	true	GA	1.14	1.18
PodReadinessGates	false	Alpha	1.11	1.11
PodReadinessGates	true	Beta	1.12	1.13
PodReadinessGates	true	GA	1.14	1.16
PodSecurity	false	Alpha	1.22	1.22
PodSecurity	true	Beta	1.23	1.24
PodSecurity	true	GA	1.25	1.27
PodShareProcessNamespace	false	Alpha	1.10	1.11
PodShareProcessNamespace	true	Beta	1.12	1.16
PodShareProcessNamespace	true	GA	1.17	1.19
PreferNominatedNode	false	Alpha	1.21	1.21
PreferNominatedNode	true	Beta	1.22	1.23
PreferNominatedNode	true	GA	1.24	1.25
ProbeTerminationGracePeriod	false	Alpha	1.21	1.21
ProbeTerminationGracePeriod	false	Beta	1.22	1.24
ProbeTerminationGracePeriod	true	Beta	1.25	1.27
ProbeTerminationGracePeriod	true	GA	1.28	1.28
ProxyTerminatingEndpoints	false	Alpha	1.22	1.25
ProxyTerminatingEndpoints	true	Beta	1.26	1.27
ProxyTerminatingEndpoints	true	GA	1.28	1.29
PVCProtection	false	Alpha	1.9	1.9
PVCProtection	–	Deprecated	1.10	1.10
ReadOnlyAPIDataVolumes	true	Beta	1.8	1.9
ReadOnlyAPIDataVolumes	–	GA	1.10	1.10
ReadWriteOncePod	false	Alpha	1.22	1.26
ReadWriteOncePod	true	Beta	1.27	1.28
ReadWriteOncePod	true	GA	1.29	1.30
RemainingItemCount	false	Alpha	1.15	1.15
RemainingItemCount	true	Beta	1.16	1.28
RemainingItemCount	true	GA	1.29	1.32
RemoveSelfLink	false	Alpha	1.16	1.19
RemoveSelfLink	true	Beta	1.20	1.23
RemoveSelfLink	true	GA	1.24	1.29
RequestManagement	false	Alpha	1.15	1.16
RequestManagement	–	Deprecated	1.17	1.17
ResourceLimitsPriorityFunction	false	Alpha	1.9	1.18
ResourceLimitsPriorityFunction	–	Deprecated	1.19	1.19
ResourceQuotaScopeSelectors	false	Alpha	1.11	1.11
ResourceQuotaScopeSelectors	true	Beta	1.12	1.16
ResourceQuotaScopeSelectors	true	GA	1.17	1.18
RetroactiveDefaultStorageClass	false	Alpha	1.25	1.25
RetroactiveDefaultStorageClass	true	Beta	1.26	1.27
RetroactiveDefaultStorageClass	true	GA	1.28	1.28
RootCAConfigMap	false	Alpha	1.13	1.19
RootCAConfigMap	true	Beta	1.20	1.20
RootCAConfigMap	true	GA	1.21	1.22
RotateKubeletClientCertificate	true	Beta	1.8	1.18
RotateKubeletClientCertificate	true	GA	1.19	1.21
RunAsGroup	true	Beta	1.14	1.20

Feature	Default	Stage	From	To
RunAsGroup	true	GA	1.21	1.22
RuntimeClass	false	Alpha	1.12	1.13
RuntimeClass	true	Beta	1.14	1.19
RuntimeClass	true	GA	1.20	1.24
ScheduleDaemonSetPods	false	Alpha	1.11	1.11
ScheduleDaemonSetPods	true	Beta	1.12	1.16
ScheduleDaemonSetPods	true	GA	1.17	1.18
SCTPSupport	false	Alpha	1.12	1.18
SCTPSupport	true	Beta	1.19	1.19
SCTPSupport	true	GA	1.20	1.22
SeccompDefault	false	Alpha	1.22	1.24
SeccompDefault	true	Beta	1.25	1.26
SeccompDefault	true	GA	1.27	1.28
SecurityContextDeny	false	Alpha	1.27	1.29
SelectorIndex	false	Alpha	1.18	1.18
SelectorIndex	true	Beta	1.19	1.19
SelectorIndex	true	GA	1.20	1.25
ServerSideApply	false	Alpha	1.14	1.15
ServerSideApply	true	Beta	1.16	1.21
ServerSideApply	true	GA	1.22	1.31
ServerSideFieldValidation	false	Alpha	1.23	1.24
ServerSideFieldValidation	true	Beta	1.25	1.26
ServerSideFieldValidation	true	GA	1.27	1.31
ServiceAccountIssuerDiscovery	false	Alpha	1.18	1.19
ServiceAccountIssuerDiscovery	true	Beta	1.20	1.20
ServiceAccountIssuerDiscovery	true	GA	1.21	1.23
ServiceAppProtocol	false	Alpha	1.18	1.18
ServiceAppProtocol	true	Beta	1.19	1.19
ServiceAppProtocol	true	GA	1.20	1.22
ServiceInternalTrafficPolicy	false	Alpha	1.21	1.21
ServiceInternalTrafficPolicy	true	Beta	1.22	1.25
ServiceInternalTrafficPolicy	true	GA	1.26	1.27
ServiceIPStaticSubrange	false	Alpha	1.24	1.24
ServiceIPStaticSubrange	true	Beta	1.25	1.25
ServiceIPStaticSubrange	true	GA	1.26	1.27
ServiceLBNodePortControl	false	Alpha	1.20	1.21
ServiceLBNodePortControl	true	Beta	1.22	1.23
ServiceLBNodePortControl	true	GA	1.24	1.25
ServiceLoadBalancerClass	false	Alpha	1.21	1.21
ServiceLoadBalancerClass	true	Beta	1.22	1.23
ServiceLoadBalancerClass	true	GA	1.24	1.25
ServiceLoadBalancerFinalizer	false	Alpha	1.15	1.15
ServiceLoadBalancerFinalizer	true	Beta	1.16	1.16
ServiceLoadBalancerFinalizer	true	GA	1.17	1.20
ServiceNodeExclusion	false	Alpha	1.8	1.18
ServiceNodeExclusion	true	Beta	1.19	1.20
ServiceNodeExclusion	true	GA	1.21	1.22
ServiceNodePortStaticSubrange	false	Alpha	1.27	1.27
ServiceNodePortStaticSubrange	true	Beta	1.28	1.28
ServiceNodePortStaticSubrange	true	GA	1.29	1.30
ServiceTopology	false	Alpha	1.17	1.19
ServiceTopology	false	Deprecated	1.20	1.22
SetHostnameAsFQDN	false	Alpha	1.19	1.19
SetHostnameAsFQDN	true	Beta	1.20	1.21
SetHostnameAsFQDN	true	GA	1.22	1.24
SkipReadOnlyValidationGCE	false	Alpha	1.28	1.28
SkipReadOnlyValidationGCE	true	Deprecated	1.29	1.30
StableLoadBalancerNodeSet	true	Beta	1.27	1.29

Feature	Default	Stage	From	To
`StableLoadBalancerNodeSet`	`true`	GA	1.30	1.31
`StartupProbe`	`false`	Alpha	1.16	1.17
`StartupProbe`	`true`	Beta	1.18	1.19
`StartupProbe`	`true`	GA	1.20	1.23
`StatefulSetMinReadySeconds`	`false`	Alpha	1.22	1.22
`StatefulSetMinReadySeconds`	`true`	Beta	1.23	1.24
`StatefulSetMinReadySeconds`	`true`	GA	1.25	1.26
`StorageObjectInUseProtection`	`true`	Beta	1.10	1.10
`StorageObjectInUseProtection`	`true`	GA	1.11	1.24
`StreamingProxyRedirects`	`false`	Beta	1.5	1.5
`StreamingProxyRedirects`	`true`	Beta	1.6	1.17
`StreamingProxyRedirects`	`true`	Deprecated	1.18	1.21
`StreamingProxyRedirects`	`false`	Deprecated	1.22	1.24
`SupportIPVSProxyMode`	`false`	Alpha	1.8	1.8
`SupportIPVSProxyMode`	`false`	Beta	1.9	1.9
`SupportIPVSProxyMode`	`true`	Beta	1.10	1.10
`SupportIPVSProxyMode`	`true`	GA	1.11	1.20
`SupportNodePidsLimit`	`false`	Alpha	1.14	1.14
`SupportNodePidsLimit`	`true`	Beta	1.15	1.19
`SupportNodePidsLimit`	`true`	GA	1.20	1.23
`SupportPodPidsLimit`	`false`	Alpha	1.10	1.13
`SupportPodPidsLimit`	`true`	Beta	1.14	1.19
`SupportPodPidsLimit`	`true`	GA	1.20	1.23
`SuspendJob`	`false`	Alpha	1.21	1.21
`SuspendJob`	`true`	Beta	1.22	1.23
`SuspendJob`	`true`	GA	1.24	1.25
`Sysctls`	`true`	Beta	1.11	1.20
`Sysctls`	`true`	GA	1.21	1.22
`TaintBasedEvictions`	`false`	Alpha	1.6	1.12
`TaintBasedEvictions`	`true`	Beta	1.13	1.17
`TaintBasedEvictions`	`true`	GA	1.18	1.20
`TaintNodesByCondition`	`false`	Alpha	1.8	1.11
`TaintNodesByCondition`	`true`	Beta	1.12	1.16
`TaintNodesByCondition`	`true`	GA	1.17	1.18
`TokenRequest`	`false`	Alpha	1.10	1.11
`TokenRequest`	`true`	Beta	1.12	1.19
`TokenRequest`	`true`	GA	1.20	1.21
`TokenRequestProjection`	`false`	Alpha	1.11	1.11
`TokenRequestProjection`	`true`	Beta	1.12	1.19
`TokenRequestProjection`	`true`	GA	1.20	1.21
`TopologyManager`	`false`	Alpha	1.16	1.17
`TopologyManager`	`true`	Beta	1.18	1.26
`TopologyManager`	`true`	GA	1.27	1.28
`TTLAfterFinished`	`false`	Alpha	1.12	1.20
`TTLAfterFinished`	`true`	Beta	1.21	1.22
`TTLAfterFinished`	`true`	GA	1.23	1.24
`UserNamespacesStatelessPodsSupport`	`false`	Alpha	1.25	1.27
`ValidateProxyRedirects`	`false`	Alpha	1.12	1.13
`ValidateProxyRedirects`	`true`	Beta	1.14	1.21
`ValidateProxyRedirects`	`true`	Deprecated	1.22	1.24
`ValidatingAdmissionPolicy`	`false`	Alpha	1.26	1.27
`ValidatingAdmissionPolicy`	`false`	Beta	1.28	1.29
`ValidatingAdmissionPolicy`	`true`	GA	1.30	1.31
`VolumeCapacityPriority`	`false`	Alpha	1.21	1.32
`VolumePVCDataSource`	`false`	Alpha	1.15	1.15
`VolumePVCDataSource`	`true`	Beta	1.16	1.17
`VolumePVCDataSource`	`true`	GA	1.18	1.21
`VolumeScheduling`	`false`	Alpha	1.9	1.9

Feature	Default	Stage	From	To
`VolumeScheduling`	`true`	Beta	1.10	1.12
`VolumeScheduling`	`true`	GA	1.13	1.16
`VolumeSnapshotDataSource`	`false`	Alpha	1.12	1.16
`VolumeSnapshotDataSource`	`true`	Beta	1.17	1.19
`VolumeSnapshotDataSource`	`true`	GA	1.20	1.22
`VolumeSubpath`	`true`	GA	1.10	1.24
`VolumeSubpathEnvExpansion`	`false`	Alpha	1.14	1.14
`VolumeSubpathEnvExpansion`	`true`	Beta	1.15	1.16
`VolumeSubpathEnvExpansion`	`true`	GA	1.17	1.24
`WarningHeaders`	`true`	Beta	1.19	1.21
`WarningHeaders`	`true`	GA	1.22	1.24
`WatchBookmark`	`false`	Alpha	1.15	1.15
`WatchBookmark`	`true`	Beta	1.16	1.16
`WatchBookmark`	`true`	GA	1.17	1.32
`WindowsEndpointSliceProxying`	`false`	Alpha	1.19	1.20
`WindowsEndpointSliceProxying`	`true`	Beta	1.21	1.21
`WindowsEndpointSliceProxying`	`true`	GA	1.22	1.24
`WindowsGMSA`	`false`	Alpha	1.14	1.15
`WindowsGMSA`	`true`	Beta	1.16	1.17
`WindowsGMSA`	`true`	GA	1.18	1.18
`WindowsHostProcessContainers`	`false`	Alpha	1.22	1.22
`WindowsHostProcessContainers`	`true`	Beta	1.23	1.25
`WindowsHostProcessContainers`	`true`	GA	1.26	1.27
`WindowsRunAsUserName`	`false`	Alpha	1.16	1.16
`WindowsRunAsUserName`	`true`	Beta	1.17	1.17
`WindowsRunAsUserName`	`true`	GA	1.18	1.20
`ZeroLimitedNominalConcurrencyShares`	`false`	Beta	1.29	1.29
`ZeroLimitedNominalConcurrencyShares`	`true`	GA	1.30	1.31

Descriptions for removed feature gates

`Accelerators`

: Provided an early form of plugin to enable Nvidia GPU support when using Docker Engine; no longer available. See Device Plugins for an alternative.

`AdmissionWebhookMatchConditions`

: Enable match conditions on mutating & validating admission webhooks.

`AdvancedAuditing`

: Enable advanced auditing

`AffinityInAnnotations`

: Enable setting Pod affinity or anti-affinity.

`AggregatedDiscoveryEndpoint`

: Enable a single HTTP endpoint `/discovery/<version>` which supports native HTTP caching with ETags containing all APIResources known to the API server.

`AllowExtTrafficLocalEndpoints`

: Enable a service to route external requests to node local endpoints.

`AllowInsecureBackendProxy`

: Enable the users to skip TLS verification of kubelets on Pod log requests.

`APIListChunking`

: Enable the API clients to retrieve (`LIST` or `GET`) resources from API server in chunks.

`APIPriorityAndFairness`

Enable managing request concurrency with prioritization and fairness at each server. (Renamed from `RequestManagement`)

`APISelfSubjectReview`

Activate the `SelfSubjectReview` API which allows users to see the requesting subject's authentication information. See API access to authentication information for a client for more details.

`AppArmor`

Enable use of AppArmor mandatory access control for Pods running on Linux nodes. See AppArmor Tutorial for more details.

`AppArmorFields`

Enable AppArmor related security context settings.

For more information about AppArmor and Kubernetes, read the AppArmor section within security features in the Linux kernel.

`AttachVolumeLimit`

Enable volume plugins to report limits on number of volumes that can be attached to a node. See dynamic volume limits for more details.

`BalanceAttachedNodeVolumes`

Include volume count on node to be considered for balanced resource allocation while scheduling. A node which has closer CPU, memory utilization, and volume count is favored by the scheduler while making decisions.

`BlockVolume`

Enable the definition and consumption of raw block devices in Pods. See Raw Block Volume Support for more details.

`BoundServiceAccountTokenVolume`

Migrate ServiceAccount volumes to use a projected volume consisting of a ServiceAccountTokenVolumeProjection. Cluster admins can use metric `serviceaccount_stale_tokens_total` to monitor workloads that are depending on the extended tokens. If there are no such workloads, turn off extended tokens by starting `kube-apiserver` with flag `--service-account-extend-token-expiration=false`.

Check Bound Service Account Tokens for more details.

`CloudDualStackNodeIPs`

Enables dual-stack `kubelet --node-ip` with external cloud providers. See Configure IPv4/IPv6 dual-stack for more details.

`ConfigurableFSGroupPolicy`

Allows user to configure volume permission change policy for fsGroups when mounting a volume in a Pod. See Configure volume permission and ownership change policy for Pods for more details.

`ConsistentHTTPGetHandlers`

Normalize HTTP get URL and Header passing for lifecycle handlers with probers.

`ControllerManagerLeaderMigration`

Enables Leader Migration for kube-controller-manager and cloud-controller-manager which allows a cluster operator to live migrate controllers from the kube-controller-manager into an external controller-manager (e.g. the cloud-controller-manager) in an HA cluster without downtime.

`CPUManager`

Enable container level CPU affinity support, see CPU Management Policies.

`CRIContainerLogRotation`

Enable container log rotation for CRI container runtime. The default max size of a log file is 10MB and the default max number of log files allowed for a container is 5. These values can be configured in the kubelet config. See logging at node level for more details.

`CronJobControllerV2`

Use an alternative implementation of the CronJob controller. Otherwise, version 1 of the same controller is selected.

`CronJobTimeZone`

Allow the use of the `timeZone` optional field in CronJobs

`CSIBlockVolume`

Enable external CSI volume drivers to support block storage. See `csi` raw block volume support for more details.

`CSIDriverRegistry`

: Enable all logic related to the CSIDriver API object in `csi.storage.k8s.io`.

`CSIInlineVolume`

: Enable CSI Inline volumes support for pods.

`CSIMigration`

: Enables shims and translation logic to route volume operations from in-tree plugins to corresponding pre-installed CSI plugins

`CSIMigrationAWS`

: Enables shims and translation logic to route volume operations from the AWS-EBS in-tree plugin to EBS CSI plugin. Supports falling back to in-tree EBS plugin for mount operations to nodes that have the feature disabled or that do not have EBS CSI plugin installed and configured. Does not support falling back for provision operations, for those the CSI plugin must be installed and configured.

`CSIMigrationAWSComplete`

: Stops registering the EBS in-tree plugin in kubelet and volume controllers and enables shims and translation logic to route volume operations from the AWS-EBS in-tree plugin to EBS CSI plugin. Requires CSIMigration and CSIMigrationAWS feature flags enabled and EBS CSI plugin installed and configured on all nodes in the cluster. This flag has been deprecated in favor of the `InTreePluginAWSUnregister` feature flag which prevents the registration of in-tree EBS plugin.

`CSIMigrationAzureDisk`

: Enables shims and translation logic to route volume operations from the Azure-Disk in-tree plugin to AzureDisk CSI plugin. Supports falling back to in-tree AzureDisk plugin for mount operations to nodes that have the feature disabled or that do not have AzureDisk CSI plugin installed and configured. Does not support falling back for provision operations, for those the CSI plugin must be installed and configured. Requires CSIMigration feature flag enabled.

`CSIMigrationAzureDiskComplete`

: Stops registering the Azure-Disk in-tree plugin in kubelet and volume controllers and enables shims and translation logic to route volume operations from the Azure-Disk in-tree plugin to AzureDisk CSI plugin. Requires CSIMigration and CSIMigrationAzureDisk feature flags enabled and AzureDisk CSI plugin installed and configured on all nodes in the cluster. This flag has been deprecated in favor of the `InTreePluginAzureDiskUnregister` feature flag which prevents the registration of in-tree AzureDisk plugin.

`CSIMigrationAzureFile`

: Enables shims and translation logic to route volume operations from the Azure-File in-tree plugin to AzureFile CSI plugin. Supports falling back to in-tree AzureFile plugin for mount operations to nodes that have the feature disabled or that do not have AzureFile CSI plugin installed and configured. Does not support falling back for provision operations, for those the CSI plugin must be installed and configured. Requires CSIMigration feature flag enabled.

`CSIMigrationAzureFileComplete`

: Stops registering the Azure-File in-tree plugin in kubelet and volume controllers and enables shims and translation logic to route volume operations from the Azure-File in-tree plugin to AzureFile CSI plugin. Requires CSIMigration and CSIMigrationAzureFile feature flags enabled and AzureFile CSI plugin installed and configured on all nodes in the cluster. This flag has been deprecated in favor of the `InTreePluginAzureFileUnregister` feature flag which prevents the registration of in-tree AzureFile plugin.

`CSIMigrationGCE`

: Enables shims and translation logic to route volume operations from the GCE-PD in-tree plugin to PD CSI plugin. Supports falling back to in-tree GCE plugin for mount operations to nodes that have the feature disabled or that do not have PD CSI plugin installed and configured. Does not support falling back for provision operations, for those the CSI plugin must be installed and configured. Requires CSIMigration feature flag enabled.

`CSIMigrationGCEComplete`

: Stops registering the GCE-PD in-tree plugin in kubelet and volume controllers and enables shims and translation logic to route volume operations from the GCE-PD in-tree plugin to PD CSI plugin. Requires CSIMigration and CSIMigrationGCE feature flags enabled and PD CSI plugin installed and configured on all nodes in the cluster. This flag has been deprecated in favor of the `InTreePluginGCEUnregister` feature flag which prevents the registration of in-tree GCE PD plugin.

`CSIMigrationOpenStack`

: Enables shims and translation logic to route volume operations from the Cinder in-tree plugin to Cinder CSI plugin. Supports falling back to in-tree Cinder plugin for mount operations to nodes that have the feature disabled or that do not have Cinder CSI plugin installed and configured. Does not support falling back for provision operations, for those the CSI plugin must be installed and configured. Requires CSIMigration feature flag enabled.

`CSIMigrationOpenStackComplete`

Stops registering the Cinder in-tree plugin in kubelet and volume controllers and enables shims and translation logic to route volume operations from the Cinder in-tree plugin to Cinder CSI plugin. Requires CSIMigration and CSIMigrationOpenStack feature flags enabled and Cinder CSI plugin installed and configured on all nodes in the cluster. This flag has been deprecated in favor of the `InTreePluginOpenStackUnregister` feature flag which prevents the registration of in-tree openstack cinder plugin.

`CSIMigrationRBD`

Enables shims and translation logic to route volume operations from the RBD in-tree plugin to Ceph RBD CSI plugin. Requires CSIMigration and csiMigrationRBD feature flags enabled and Ceph CSI plugin installed and configured in the cluster.

This feature gate was deprecated in favor of the `InTreePluginRBDUnregister` feature gate, which prevents the registration of in-tree RBD plugin.

`CSIMigrationvSphere`

Enables shims and translation logic to route volume operations from the vSphere in-tree plugin to vSphere CSI plugin. Supports falling back to in-tree vSphere plugin for mount operations to nodes that have the feature disabled or that do not have vSphere CSI plugin installed and configured. Does not support falling back for provision operations, for those the CSI plugin must be installed and configured. Requires CSIMigration feature flag enabled.

`CSIMigrationvSphereComplete`

Stops registering the vSphere in-tree plugin in kubelet and volume controllers and enables shims and translation logic to route volume operations from the vSphere in-tree plugin to vSphere CSI plugin. Requires CSIMigration and CSIMigrationvSphere feature flags enabled and vSphere CSI plugin installed and configured on all nodes in the cluster. This flag has been deprecated in favor of the `InTreePluginvSphereUnregister` feature flag which prevents the registration of in-tree vsphere plugin.

`CSINodeExpandSecret`

Enable passing secret authentication data to a CSI driver for use during a `NodeExpandVolume` CSI operation.

`CSINodeInfo`

Enable all logic related to the CSINodeInfo API object in `csi.storage.k8s.io`.

`CSIPersistentVolume`

Enable discovering and mounting volumes provisioned through a CSI (Container Storage Interface) compatible volume plugin.

`CSIServiceAccountToken`

Enable CSI drivers to receive the pods' service account token that they mount volumes for. See Token Requests.

`CSIStorageCapacity`

Enables CSI drivers to publish storage capacity information and the Kubernetes scheduler to use that information when scheduling pods. See Storage Capacity. Check the `csi` volume type documentation for more details.

`CSIVolumeFSGroupPolicy`

Allows CSIDrivers to use the `fsGroupPolicy` field. This field controls whether volumes created by a CSIDriver support volume ownership and permission modifications when these volumes are mounted.

`CSRDuration`

Allows clients to request a duration for certificates issued via the Kubernetes CSR API.

`CustomPodDNS`

Enable customizing the DNS settings for a Pod using its `dnsConfig` property. Check Pod's DNS Config for more details.

`CustomResourceDefaulting`

Enable CRD support for default values in OpenAPI v3 validation schemas.

`CustomResourcePublishOpenAPI`

Enables publishing of CRD OpenAPI specs.

`CustomResourceSubresources`

Enable `/status` and `/scale` subresources on resources created from CustomResourceDefinition.

`CustomResourceValidation`

Enable schema based validation on resources created from CustomResourceDefinition.

`CustomResourceValidationExpressions`

Enable expression language validation in CRD which will validate customer resource based on validation rules written in the `x-kubernetes-validations` extension.

`CustomResourceWebhookConversion`

Enable webhook-based conversion on resources created from CustomResourceDefinition.

`DaemonSetUpdateSurge`

Enables the DaemonSet workloads to maintain availability during update per node. See Perform a Rolling Update on a DaemonSet.

`DefaultHostNetworkHostPortsInPodTemplates`

This feature gate controls the point at which a default value for `.spec.containers[*].ports[*].hostPort` is assigned, for Pods using `hostNetwork: true`. The default since Kubernetes v1.28 is to only set a default value in Pods.

Enabling this means a default will be assigned even to the `.spec` of an embedded PodTemplate (for example, in a Deployment), which is the way that older releases of Kubernetes worked. You should migrate your code so that it does not rely on the legacy behavior.

`DefaultPodTopologySpread`

Enables the use of `PodTopologySpread` scheduling plugin to do default spreading.

`DelegateFSGroupToCSIDriver`

If supported by the CSI driver, delegates the role of applying `fsGroup` from a Pod's `securityContext` to the driver by passing `fsGroup` through the NodeStageVolume and NodePublishVolume CSI calls.

`DevicePluginCDIDevices`

Enable support to CDI device IDs in the Device Plugin API.

`DevicePlugins`

Enable the device-plugins based resource provisioning on nodes.

`DisableAcceleratorUsageMetrics`

Disable accelerator metrics collected by the kubelet.

`DisableCloudProviders`

Enabling this feature gate deactivated functionality in `kube-apiserver`, `kube-controller-manager` and `kubelet` that related to the `--cloud-provider` command line argument.

In Kubernetes v1.31 and later, the only valid values for `--cloud-provider` are the empty string (no cloud provider integration), or "external" (integration via a separate cloud-controller-manager).

`DisableKubeletCloudCredentialProviders`

Enabling the feature gate deactivated the legacy in-tree functionality within the kubelet, that allowed the kubelet to to authenticate to a cloud provider container registry for container image pulls.

`DownwardAPIHugePages`

Enables usage of hugepages in downward API.

`DRAControlPlaneController`

Enables support for resources with custom parameters and a lifecycle that is independent of a Pod. Allocation of resources is handled by a resource driver's control plane controller.

`DryRun`

Enable server-side dry run requests so that validation, merging, and mutation can be tested without committing.

`DynamicAuditing`

Used to enable dynamic auditing before v1.19.

`DynamicKubeletConfig`

Enable the dynamic configuration of kubelet. The feature is no longer supported outside of supported skew policy. The feature gate was removed from kubelet in 1.24.

`DynamicProvisioningScheduling`

Extend the default scheduler to be aware of volume topology and handle PV provisioning. This feature was superseded by the `VolumeScheduling` feature in v1.12.

`DynamicVolumeProvisioning`

Enable the dynamic provisioning of persistent volumes to Pods.

`EfficientWatchResumption`

Allows for storage-originated bookmark (progress notify) events to be delivered to the users. This is only applied to watch operations.

`EnableAggregatedDiscoveryTimeout`

Enable the five second timeout on aggregated discovery calls.

`EnableEquivalenceClassCache`

Enable the scheduler to cache equivalence of nodes when scheduling Pods.

`EndpointSlice`

Enables EndpointSlices for more scalable and extensible network endpoints. See Enabling EndpointSlices.

`EndpointSliceNodeName`

Enables EndpointSlice `nodeName` field.

`EndpointSliceProxying`

When enabled, kube-proxy running on Linux will use EndpointSlices as the primary data source instead of Endpoints, enabling scalability and performance improvements. See Enabling Endpoint Slices.

`EndpointSliceTerminatingCondition`

Enables EndpointSlice `terminating` and `serving` condition fields.

`EphemeralContainers`

Enable the ability to add ephemeral containers to running Pods.

`EvenPodsSpread`

Enable pods to be scheduled evenly across topology domains. See Pod Topology Spread Constraints.

`ExpandCSIVolumes`

Enable the expanding of CSI volumes.

`ExpandedDNSConfig`

Enable kubelet and kube-apiserver to allow more DNS search paths and longer list of DNS search paths. This feature requires container runtime support(Containerd: v1.5.6 or higher, CRI-O: v1.22 or higher). See Expanded DNS Configuration.

`ExpandInUsePersistentVolumes`

Enable expanding in-use PVCs. See Resizing an in-use PersistentVolumeClaim.

`ExpandPersistentVolumes`

Enable the expanding of persistent volumes. See Expanding Persistent Volumes Claims.

`ExperimentalCriticalPodAnnotation`

Enable annotating specific pods as *critical* so that their scheduling is guaranteed. This feature is deprecated by Pod Priority and Preemption as of v1.13.

`ExperimentalHostUserNamespaceDefaulting`

Enabling the defaulting user namespace to host. This is for containers that are using other host namespaces, host mounts, or containers that are privileged or using specific non-namespaced capabilities (e.g. `MKNODE`, `SYS_MODULE` etc.). This should only be enabled if user namespace remapping is enabled in the Docker daemon.

`ExternalPolicyForExternalIP`

Fix a bug where ExternalTrafficPolicy is not applied to Service ExternalIPs.

`GCERegionalPersistentDisk`

Enable the regional PD feature on GCE.

`GenericEphemeralVolume`

Enables ephemeral, inline volumes that support all features of normal volumes (can be provided by third-party storage vendors, storage capacity tracking, restore from snapshot, etc.). See Ephemeral Volumes.

`GRPCContainerProbe`

Enables the gRPC probe method for {Liveness,Readiness,Startup}Probe. See Configure Liveness, Readiness and Startup Probes.

`HPAContainerMetrics`

Allow HorizontalPodAutoscalers to scale based on metrics from individual containers within target pods.

`HugePages`

Enable the allocation and consumption of pre-allocated huge pages.

`HugePageStorageMediumSize`

Enable support for multiple sizes pre-allocated huge pages.

`HyperVContainer`

Enable Hyper-V isolation for Windows containers.

`IdentifyPodOS`

Allows the Pod OS field to be specified. This helps in identifying the OS of the pod authoritatively during the API server admission time.

`ImmutableEphemeralVolumes`

Allows for marking individual Secrets and ConfigMaps as immutable for better safety and performance.

`IndexedJob`

Allows the Job controller to manage Pod completions per completion index.

`IngressClassNamespacedParams`

Allow namespace-scoped parameters reference in `IngressClass` resource. This feature adds two fields - `Scope` and `Namespace` to `IngressClass.spec.parameters`.

`Initializers`

Allow asynchronous coordination of object creation using the Initializers admission plugin.

`InTreePluginAWSUnregister`

Stops registering the aws-ebs in-tree plugin in kubelet and volume controllers.

`InTreePluginAzureDiskUnregister`

Stops registering the azuredisk in-tree plugin in kubelet and volume controllers.

`InTreePluginAzureFileUnregister`

Stops registering the azurefile in-tree plugin in kubelet and volume controllers.

`InTreePluginGCEUnregister`

Stops registering the gce-pd in-tree plugin in kubelet and volume controllers.

`InTreePluginOpenStackUnregister`

Stops registering the OpenStack cinder in-tree plugin in kubelet and volume controllers.

`InTreePluginRBDUnregister`

Stops registering the RBD in-tree plugin within kubelet and volume controllers.

`InTreePluginvSphereUnregister`

Stops registering the vSphere in-tree plugin in kubelet and volume controllers.

`IPTablesOwnershipCleanup`

This causes kubelet to no longer create legacy iptables rules.

`IPv6DualStack`

Enable dual stack support for IPv6.

`JobMutableNodeSchedulingDirectives`

Allows updating node scheduling directives in the pod template of Job.

`JobPodFailurePolicy`

Allow users to specify handling of pod failures based on container exit codes and pod conditions.

`JobReadyPods`

Enables tracking the number of Pods that have a `Ready` condition. The count of `Ready` pods is recorded in the status of a Job status.

`JobTrackingWithFinalizers`

Enables tracking Job completions without relying on Pods remaining in the cluster indefinitely. The Job controller uses Pod finalizers and a field in the Job status to keep track of the finished Pods to count towards completion.

`KMSv2`

Enables KMS v2 API for encryption at rest. See Using a KMS Provider for data encryption for more details.

`KMSv2KDF`

Enables KMS v2 to generate single use data encryption keys. See Using a KMS Provider for data encryption for more details. If the `KMSv2` feature gate is not enabled in your cluster, the value of the `KMSv2KDF` feature gate has no effect.

`KubeletConfigFile`

Enable loading kubelet configuration from a file specified using a config file. See setting kubelet parameters via a config file for more details.

`KubeletCredentialProviders`

Enable kubelet exec credential providers for image pull credentials.

`KubeletPluginsWatcher`

Enable probe-based plugin watcher utility to enable kubelet to discover plugins such as CSI volume drivers.

`KubeletPodResources`

Enable the kubelet's pod resources gRPC endpoint. See Support Device Monitoring for more details.

`KubeletPodResourcesGetAllocatable`

Enable the kubelet's pod resources `GetAllocatableResources` functionality. This API augments the resource allocation reporting

`KubeProxyDrainingTerminatingNodes`

Implement connection draining for terminating nodes for `externalTrafficPolicy: Cluster` services.

`LegacyNodeRoleBehavior`

When disabled, legacy behavior in service load balancers and node disruption will ignore the `node-role.kubernetes.io/master` label in favor of the feature-specific labels provided by `NodeDisruptionExclusion` and `ServiceNodeExclusion`.

`LegacyServiceAccountTokenCleanUp`

Enable cleaning up Secret-based service account tokens when they are not used in a specified time (default to be one year).

`LegacyServiceAccountTokenNoAutoGeneration`

Stop auto-generation of Secret-based service account tokens.

`LegacyServiceAccountTokenTracking`

Track usage of Secret-based service account tokens.

`LocalStorageCapacityIsolation`

Enable the consumption of local ephemeral storage and also the `sizeLimit` property of an emptyDir volume.

`MinDomainsInPodTopologySpread`

Enable `minDomains` in Pod topology spread constraints.

`MinimizeIPTablesRestore`

Enables new performance improvement logics in the kube-proxy iptables mode.

`MixedProtocolLBService`

Enable using different protocols in the same `LoadBalancer` type Service instance.

`MountContainers`

Enable using utility containers on host as the volume mounter.

`MountPropagation`

Enable sharing volume mounted by one container to other containers or pods. For more details, please see mount propagation.

`MultiCIDRRangeAllocator`

Enables the MultiCIDR range allocator.

`NamespaceDefaultLabelName`

Configure the API Server to set an immutable label `kubernetes.io/metadata.name` on all namespaces, containing the namespace name.

`NetworkPolicyEndPort`

Allows you to define ports in a NetworkPolicy rule as a range of port numbers.

`NetworkPolicyStatus`

Enable the `status` subresource for NetworkPolicy objects.

`NewVolumeManagerReconstruction`

Enables improved discovery of mounted volumes during kubelet startup. Since the associated code had been significantly refactored, Kubernetes versions 1.25 to 1.29 allowed you to opt-out in case the kubelet got stuck at the startup, or did not unmount volumes from terminated Pods.

This refactoring was behind the `SELinuxMountReadWriteOncePod` feature gate in Kubernetes releases 1.25 and 1.26.

`NodeDisruptionExclusion`

Enable use of the Node label `node.kubernetes.io/exclude-disruption` which prevents nodes from being evacuated during zone failures.

`NodeLease`

Enable the new Lease API to report node heartbeats, which could be used as a node health signal.

`NodeOutOfServiceVolumeDetach`

When a Node is marked out-of-service using the `node.kubernetes.io/out-of-service` taint, Pods on the node will be forcefully deleted if they can not tolerate this taint, and the volume detach operations for Pods terminating on the node will happen immediately. The deleted Pods can recover quickly on different nodes.

`NonPreemptingPriority`

Enable `preemptionPolicy` field for PriorityClass and Pod.

`OpenAPIV3`

Enables the API server to publish OpenAPI v3.

`PDBUnhealthyPodEvictionPolicy`

Enables the `unhealthyPodEvictionPolicy` field of a `PodDisruptionBudget`. This specifies when unhealthy pods should be considered for eviction. Please see Unhealthy Pod Eviction Policy for more details.

`PersistentLocalVolumes`

Enable the usage of `local` volume type in Pods. Pod affinity has to be specified if requesting a `local` volume.

`PersistentVolumeLastPhaseTransitionTime`

Adds a new field to PersistentVolume which holds a timestamp of when the volume last transitioned its phase.

`PodAffinityNamespaceSelector`

Enable the Pod Affinity Namespace Selector and CrossNamespacePodAffinity quota scope features.

`PodDisruptionBudget`

: Enable the PodDisruptionBudget feature.

`PodDisruptionConditions`

: Enabled support for appending a dedicated pod condition indicating that the pod is being deleted due to a disruption.

`PodHasNetworkCondition`

: Enable the kubelet to mark the PodHasNetwork condition on pods. This was renamed to `PodReadyToStartContainersCondition` in 1.28.

`PodHostIPs`

: Enable the `status.hostIPs` field for pods and the downward API. The field lets you expose host IP addresses to workloads.

`PodOverhead`

: Enable the PodOverhead feature to account for pod overheads.

`PodPriority`

: Enable the descheduling and preemption of Pods based on their priorities.

`PodReadinessGates`

: Enable the setting of `PodReadinessGate` field for extending Pod readiness evaluation. See Pod readiness gate for more details.

`PodSecurity`

: Enables the `PodSecurity` admission plugin.

`PodShareProcessNamespace`

: Enable the setting of `shareProcessNamespace` in a Pod for sharing a single process namespace between containers running in a pod. More details can be found in Share Process Namespace between Containers in a Pod.

`PreferNominatedNode`

: This flag tells the scheduler whether the nominated nodes will be checked first before looping through all the other nodes in the cluster.

`ProbeTerminationGracePeriod`

: Enable setting probe-level `terminationGracePeriodSeconds` on pods. See the enhancement proposal for more details.

`ProxyTerminatingEndpoints`

: Enable the kube-proxy to handle terminating endpoints when `ExternalTrafficPolicy=Local`.

`PVCProtection`

: Enable the prevention of a PersistentVolumeClaim (PVC) from being deleted when it is still used by any Pod.

`ReadOnlyAPIDataVolumes`

: Set `configMap`, `secret`, `downwardAPI` and `projected` volumes to be mounted read-only.

  Since Kubernetes v1.10, these volume types are always read-only and you cannot opt out.

`ReadWriteOncePod`

: Enables the usage of `ReadWriteOncePod` PersistentVolume access mode.

`RemainingItemCount`

: Allow the API servers to show a count of remaining items in the response to a chunking list request.

`RemoveSelfLink`

: Sets the `.metadata.selfLink` field to blank (empty string) for all objects and collections. This field has been deprecated since the Kubernetes v1.16 release. When this feature is enabled, the `.metadata.selfLink` field remains part of the Kubernetes API, but is always unset.

`RequestManagement`

: Enables managing request concurrency with prioritization and fairness at each API server. Deprecated by `APIPriorityAndFairness` since 1.17.

`ResourceLimitsPriorityFunction`

: Enable a scheduler priority function that assigns a lowest possible score of 1 to a node that satisfies at least one of the input Pod's cpu and memory limits. The intent is to break ties between nodes with same scores.

`ResourceQuotaScopeSelectors`

: Enable resource quota scope selectors.

`RetroactiveDefaultStorageClass`

: Allow assigning StorageClass to unbound PVCs retroactively.

`RootCAConfigMap`

: Configure the `kube-controller-manager` to publish a ConfigMap named `kube-root-ca.crt` to every namespace. This ConfigMap contains a CA bundle used for verifying connections to the kube-apiserver. See Bound Service Account Tokens for more details.

`RotateKubeletClientCertificate`

: Enable the rotation of the client TLS certificate on the kubelet. See kubelet configuration for more details.

`RunAsGroup`

: Enable control over the primary group ID set on the init processes of containers.

`RuntimeClass`

: Enable the RuntimeClass feature for selecting container runtime configurations.

`ScheduleDaemonSetPods`

: Enable DaemonSet Pods to be scheduled by the default scheduler instead of the DaemonSet controller.

`SCTPSupport`

: Enables the *SCTP* `protocol` value in Pod, Service, Endpoints, EndpointSlice, and NetworkPolicy definitions.

`SeccompDefault`

: Enables the use of `RuntimeDefault` as the default seccomp profile for all workloads. The seccomp profile is specified in the `securityContext` of a Pod and/or a Container.

`SecurityContextDeny`

: This gate signals that the `SecurityContextDeny` admission controller is deprecated.

`SelectorIndex`

: Allows label and field based indexes in API server watch cache to accelerate list operations.

`ServerSideApply`

: Enables the Sever Side Apply (SSA) feature on the API Server.

`ServerSideFieldValidation`

: Enables server-side field validation. This means the validation of resource schema is performed at the API server side rather than the client side (for example, the `kubectl create` or `kubectl apply` command line).

`ServiceAccountIssuerDiscovery`

: Enable OIDC discovery endpoints (issuer and JWKS URLs) for the service account issuer in the API server. See Configure Service Accounts for Pods for more details.

`ServiceAppProtocol`

: Enables the `appProtocol` field on Services and Endpoints.

`ServiceInternalTrafficPolicy`

: Enables the `internalTrafficPolicy` field on Services

`ServiceIPStaticSubrange`

: Enables a strategy for Services ClusterIP allocations, whereby the ClusterIP range is subdivided. Dynamic allocated ClusterIP addresses will be allocated preferently from the upper range allowing users to assign static ClusterIPs from the lower range with a low risk of collision. See Avoiding collisions for more details.

`ServiceLBNodePortControl`

: Enables the `allocateLoadBalancerNodePorts` field on Services.

`ServiceLoadBalancerClass`

: Enables the `loadBalancerClass` field on Services. See Specifying class of load balancer implementation for more details.

`ServiceLoadBalancerFinalizer`

: Enable finalizer protection for Service load balancers.

`ServiceNodeExclusion`

: Enable the exclusion of nodes from load balancers created by a cloud provider. A node is eligible for exclusion if labelled with "`node.kubernetes.io/exclude-from-external-load-balancers`".

`ServiceNodePortStaticSubrange`

: Enables the use of different port allocation strategies for NodePort Services. For more details, see reserve NodePort ranges to avoid collisions.

`ServiceTopology`

: Enable service to route traffic based upon the Node topology of the cluster.

`SetHostnameAsFQDN`

: Enable the ability of setting Fully Qualified Domain Name(FQDN) as the hostname of a pod. See Pod's `setHostnameAsFQDN` field.

`SkipReadOnlyValidationGCE`

: Skip validation that GCE PersistentDisk volumes are in read-only mode.

`StableLoadBalancerNodeSet`

: Enables less load balancer re-configurations by the service controller (KCCM) as an effect of changing node state.

`StartupProbe`

: Enable the startup probe in the kubelet.

`StatefulSetMinReadySeconds`

: Allows `minReadySeconds` to be respected by the StatefulSet controller.

`StorageObjectInUseProtection`

: Postpone the deletion of PersistentVolume or PersistentVolumeClaim objects if they are still being used.

`StreamingProxyRedirects`

: Instructs the API server to intercept (and follow) redirects from the backend (kubelet) for streaming requests. Examples of streaming requests include the `exec`, `attach` and `port-forward` requests.

`SupportIPVSProxyMode`

: Enable providing in-cluster service load balancing using IPVS. See service proxies for more details.

`SupportNodePidsLimit`

: Enable the support to limiting PIDs on the Node. The parameter `pid=<number>` in the `--system-reserved` and `--kube-reserved` options can be specified to ensure that the specified number of process IDs will be reserved for the system as a whole and for Kubernetes system daemons respectively.

`SupportPodPidsLimit`

: Enable the support to limiting PIDs in Pods.

`SuspendJob`

: Enable support to suspend and resume Jobs. For more details, see the Jobs docs.

`Sysctls`

: Enable support for namespaced kernel parameters (sysctls) that can be set for each pod. See sysctls for more details.

`TaintBasedEvictions`

Enable evicting pods from nodes based on taints on Nodes and tolerations on Pods. See taints and tolerations for more details.

`TaintNodesByCondition`

Enable automatic tainting nodes based on node conditions.

`TokenRequest`

Enable the `TokenRequest` endpoint on service account resources.

`TokenRequestProjection`

Enable the injection of service account tokens into a Pod through a `projected` volume.

`TopologyManager`

Enable a mechanism to coordinate fine-grained hardware resource assignments for different components in Kubernetes. See Control Topology Management Policies on a node.

`TTLAfterFinished`

Allow a TTL controller to clean up resources after they finish execution.

`UserNamespacesStatelessPodsSupport`

Enable user namespace support for stateless Pods. This feature gate was superseded by the `UserNamespacesSupport` feature gate in the Kubernetes v1.28 release.

`ValidateProxyRedirects`

This flag controls whether the API server should validate that redirects are only followed to the same host. Only used if the `StreamingProxyRedirects` flag is enabled.

`ValidatingAdmissionPolicy`

Enable ValidatingAdmissionPolicy support for CEL validations be used in Admission Control.

`VolumeCapacityPriority`

Enable support for prioritizing nodes in different topologies based on available PV capacity. This feature is renamed to `StorageCapacityScoring` in v1.33.

`VolumePVCDataSource`

Enable support for specifying an existing PVC as a DataSource.

`VolumeScheduling`

Enable volume topology aware scheduling and make the PersistentVolumeClaim (PVC) binding aware of scheduling decisions. It also enables the usage of `local` volume type when used together with the `PersistentLocalVolumes` feature gate.

`VolumeSnapshotDataSource`

Enable volume snapshot data source support.

`VolumeSubpath`

Allow mounting a subpath of a volume in a container.

`VolumeSubpathEnvExpansion`

Enable `subPathExpr` field for expanding environment variables into a `subPath`.

`WarningHeaders`

Allow sending warning headers in API responses.

`WatchBookmark`

Enable support for watch bookmark events.

`WindowsEndpointSliceProxying`

When enabled, kube-proxy running on Windows will use EndpointSlices as the primary data source instead of Endpoints, enabling scalability and performance improvements. See Enabling Endpoint Slices.

`WindowsGMSA`

Enables passing of GMSA credential specs from pods to container runtimes.

`WindowsHostProcessContainers`

Enables support for Windows HostProcess containers.

`WindowsRunAsUserName`

Enable support for running applications in Windows containers with as a non-default user. See Configuring RunAsUserName for more details.

`ZeroLimitedNominalConcurrencyShares`

Allow priority & fairness in the API server to use a zero value for the `nominalConcurrencyShares` field of the `limited` section of a priority level.

---

kubeadm Configuration (v1beta4)

Overview

Package v1beta4 defines the v1beta4 version of the kubeadm configuration file format. This version improves on the v1beta3 format by fixing some minor issues and adding a few new fields.

A list of changes since v1beta3:

v1.34:

- Add "ECDSA-P384" to the allowed encryption algorithm options for `ClusterConfiguration.encryptionAlgorithm`.

v1.33:

- Add an `EtcdUpgrade` field to `UpgradeConfiguration.plan` that can be used to control whether the etcd upgrade plan should be displayed.

v1.31:

- Support custom environment variables in control plane components under `ClusterConfiguration`. Use `apiServer.extraEnvs`, `controllerManager.extraEnvs`, `scheduler.extraEnvs`, `etcd.local.extraEnvs`.
- The `ResetConfiguration` API type is now supported in v1beta4. Users are able to reset a node by passing a `--config` file to `kubeadm reset`.
- Dry run mode is now configureable in `InitConfiguration` and `JoinConfiguration`B.
- Replace the existing string/string extra argument maps with structured extra arguments that support duplicates. The change applies to `ClusterConfiguration` - `apiServer.extraArgs`, `controllerManager.extraArgs`, `scheduler.extraArgs`, `etcd.local.extraArgs`. Also to `nodeRegistration.kubeletExtraArgs`.
- Add `ClusterConfiguration.encryptionAlgorithm` that can be used to set the asymmetric encryption algorithm used for this cluster's keys and certificates. Can be one of `"RSA-2048"` (default), `"RSA-3072"`, `"RSA-4096"` or `"ECDSA-P256"`.
- Add `ClusterConfiguration.dns.disabled` and `ClusterConfiguration.proxy.disabled` that can be used to disable the CoreDNS and kube-proxy addons during cluster initialization. Skipping the related addons phases, during cluster creation will set the same fields to `true`.
- Add the `nodeRegistration.imagePullSerial` field in `InitConfiguration` and `JoinConfiguration`, which can be used to control if kubeadm pulls images serially or in parallel.
- The `UpgradeConfiguration` kubeadm API is now supported in v1beta4 when passing `--config` to `kubeadm upgrade` subcommands. Usage of component configuration for `kubelet` and `kube-proxy`, `InitConfiguration` and `ClusterConfiguration` is deprecated and will be ignored when passing `--config` to `upgrade` subcommands.
- Add a `Timeouts` structure to `InitConfiguration`, `JoinConfiguration`, `ResetConfiguration` and `UpgradeConfiguration` that can be used to configure various timeouts. The `ClusterConfiguration.timeoutForControlPlane` field is replaced by `Timeouts.controlPlaneComponentHealthCheck`. The `JoinConfiguration.discovery.timeout` is replaced by `timeouts.Discovery`.
- Add a `certificateValidityPeriod` and `caCertificateValidityPeriod` fields to `ClusterConfiguration`. These fields can be used to control the validity period of certificates generated by kubeadm during sub-commands such as `init`, `join`, `upgrade` and `certs`. Default values continue to be 1 year for non-CA certificates and 10 years for CA certificates. Only non-CA certificates continue to be renewable by `kubeadm certs renew`.

Migration from old kubeadm config versions

- kubeadm v1.15.x and newer can be used to migrate from v1beta1 to v1beta2.
- kubeadm v1.22.x and newer no longer support v1beta1 and older APIs, but can be used to migrate v1beta2 to v1beta3.
- kubeadm v1.27.x and newer no longer support v1beta2 and older APIs.
- kubeadm v1.31.x and newer can be used to migrate from v1beta3 to v1beta4.

Basics

The preferred way to configure kubeadm is to pass a YAML configuration file with the `--config` option. Some of the configuration options defined in the kubeadm config file are also available as command line flags, but only the most common/simple use case are supported with this approach.

A kubeadm config file could contain multiple configuration types separated using three dashes (`---`).

kubeadm supports the following configuration types:

```
apiVersion: kubeadm.k8s.io/v1beta4
kind: InitConfiguration

apiVersion: kubeadm.k8s.io/v1beta4
kind: ClusterConfiguration

apiVersion: kubelet.config.k8s.io/v1beta1
kind: KubeletConfiguration

apiVersion: kubeproxy.config.k8s.io/v1alpha1
kind: KubeProxyConfiguration

apiVersion: kubeadm.k8s.io/v1beta4
kind: JoinConfiguration

apiVersion: kubeadm.k8s.io/v1beta4
kind: ResetConfiguration

apiVersion: kubeadm.k8s.io/v1beta4
kind: UpgradeConfiguration
```

To print the defaults for `init` and `join` actions use the following commands:

```
kubeadm config print init-defaults
kubeadm config print join-defaults
kubeadm config print reset-defaults
kubeadm config print upgrade-defaults
```

The list of configuration types that must be included in a configuration file depends by the action you are performing (`init` or `join`) and by the configuration options you are going to use (defaults or advanced customization).

If some configuration types are not provided, or provided only partially, kubeadm will use default values; defaults provided by kubeadm includes also enforcing consistency of values across components when required (e.g. `--cluster-cidr` flag on controller manager and `clusterCIDR` on kube-proxy).

Users are always allowed to override default values, with the only exception of a small subset of setting with relevance for security (e.g. enforce authorization-mode Node and RBAC on api server).

If the user provides a configuration types that is not expected for the action you are performing, kubeadm will ignore those types and print a warning.

Kubeadm init configuration types

When executing kubeadm init with the `--config` option, the following configuration types could be used: InitConfiguration, ClusterConfiguration, KubeProxyConfiguration, KubeletConfiguration, but only one between InitConfiguration and ClusterConfiguration is mandatory.

```
apiVersion: kubeadm.k8s.io/v1beta4
kind: InitConfiguration
bootstrapTokens:
...
nodeRegistration:
...
```

The InitConfiguration type should be used to configure runtime settings, that in case of kubeadm init are the configuration of the bootstrap token and all the setting which are specific to the node where kubeadm is executed, including:

- NodeRegistration, that holds fields that relate to registering the new node to the cluster; use it to customize the node name, the CRI socket to use or any other settings that should apply to this node only (e.g. the node ip).
- LocalAPIEndpoint, that represents the endpoint of the instance of the API server to be deployed on this node; use it e.g. to customize the API server advertise address.

```
apiVersion: kubeadm.k8s.io/v1beta4
kind: ClusterConfiguration
networking:
  ...
etcd:
  ...
apiServer:
  extraArgs:
```

```
    ...
  extraVolumes:
    ...
...
```

The ClusterConfiguration type should be used to configure cluster-wide settings, including settings for:

- `networking` that holds configuration for the networking topology of the cluster; use it e.g. to customize Pod subnet or services subnet.
- `etcd`: use it e.g. to customize the local etcd or to configure the API server for using an external etcd cluster.
- kube-apiserver, kube-scheduler, kube-controller-manager configurations; use it to customize control-plane components by adding customized setting or overriding kubeadm default settings.

```
apiVersion: kubeproxy.config.k8s.io/v1alpha1
kind: KubeProxyConfiguration
...
```

The KubeProxyConfiguration type should be used to change the configuration passed to kube-proxy instances deployed in the cluster. If this object is not provided or provided only partially, kubeadm applies defaults.

See https://kubernetes.io/docs/reference/command-line-tools-reference/kube-proxy/ or https://pkg.go.dev/k8s.io/kube-proxy/config/v1alpha1#KubeProxyConfiguration for kube-proxy official documentation.

```
apiVersion: kubelet.config.k8s.io/v1beta1
kind: KubeletConfiguration
...
```

The KubeletConfiguration type should be used to change the configurations that will be passed to all kubelet instances deployed in the cluster. If this object is not provided or provided only partially, kubeadm applies defaults.

See https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/ or https://pkg.go.dev/k8s.io/kubelet/config/v1beta1#KubeletConfiguration for kubelet official documentation.

Here is a fully populated example of a single YAML file containing multiple configuration types to be used during a `kubeadm init` run.

```
apiVersion: kubeadm.k8s.io/v1beta4
kind: InitConfiguration
bootstrapTokens:
  - token: "9a08jv.c0izixklcxtmnze7"
    description: "kubeadm bootstrap token"
    ttl: "24h"
  - token: "783bde.3f89s0fje9f38fhf"
    description: "another bootstrap token"
    usages:
  - authentication
  - signing
    groups:
  - system:bootstrappers:kubeadm:default-node-token

nodeRegistration:
  name: "ec2-10-100-0-1"
  criSocket: "unix:///var/run/containerd/containerd.sock"
  taints:
    - key: "kubeadmNode"
      value: "someValue"
      effect: "NoSchedule"
  kubeletExtraArgs:
    - name: v
      value: "5"
  ignorePreflightErrors:
    - IsPrivilegedUser
  imagePullPolicy: "IfNotPresent"
  imagePullSerial: true

localAPIEndpoint:
  advertiseAddress: "10.100.0.1"
  bindPort: 6443
certificateKey: "e6a2eb8581237ab72a4f494f30285ec12a9694d750b9785706a83bfcbbbd2204"
skipPhases:
  - preflight
timeouts:
  controlPlaneComponentHealthCheck: "60s"
  kubenetesAPICall: "40s"
---
apiVersion: kubeadm.k8s.io/v1beta4
kind: ClusterConfiguration
etcd:

  # one of local or external
  local:
    imageRepository: "registry.k8s.io"
    imageTag: "3.2.24"
```

```
    dataDir: "/var/lib/etcd"
    extraArgs:
      - name: listen-client-urls
        value: http://10.100.0.1:2379
    extraEnvs:
      - name: SOME_VAR
        value: SOME_VALUE
    serverCertSANs:
      - ec2-10-100-0-1.compute-1.amazonaws.com
    peerCertSANs:
      - 10.100.0.1
  # external:
  #   endpoints:
  #     - 10.100.0.1:2379
  #     - 10.100.0.2:2379
  #   caFile: "/etcd/kubernetes/pki/etcd/etcd-ca.crt"
  #   certFile: "/etcd/kubernetes/pki/etcd/etcd.crt"
  #   keyFile: "/etcd/kubernetes/pki/etcd/etcd.key"

networking:
  serviceSubnet: "10.96.0.0/16"
  podSubnet: "10.244.0.0/24"
  dnsDomain: "cluster.local"
kubernetesVersion: "v1.21.0"
controlPlaneEndpoint: "10.100.0.1:6443"
apiServer:
  extraArgs:
    - name: authorization-mode
      value: Node,RBAC
  extraEnvs:
    - name: SOME_VAR
      value: SOME_VALUE
  extraVolumes:
    - name: "some-volume"
      hostPath: "/etc/some-path"
      mountPath: "/etc/some-pod-path"
      readOnly: false
      pathType: File
  certSANs:
    - "10.100.1.1"
    - "ec2-10-100-0-1.compute-1.amazonaws.com"

controllerManager:
  extraArgs:
    - name: node-cidr-mask-size
      value: "20"
  extraVolumes:
    - name: "some-volume"
      hostPath: "/etc/some-path"
      mountPath: "/etc/some-pod-path"
      readOnly: false
      pathType: File

scheduler:
  extraArgs:
    - name: address
      value: 10.100.0.1
  extraVolumes:
    - name: "some-volume"
      hostPath: "/etc/some-path"
      mountPath: "/etc/some-pod-path"
      readOnly: false
      pathType: File

certificatesDir: "/etc/kubernetes/pki"
imageRepository: "registry.k8s.io"
clusterName: "example-cluster"
encryptionAlgorithm: ECDSA-P256
dns:
  disabled: true  # disable CoreDNS
proxy:
  disabled: true   # disable kube-proxy

---
apiVersion: kubelet.config.k8s.io/v1beta1
kind: KubeletConfiguration
# kubelet specific options here
---
apiVersion: kubeproxy.config.k8s.io/v1alpha1
kind: KubeProxyConfiguration
# kube-proxy specific options here
```

Kubeadm join configuration types

When executing kubeadm join with the --config option, the JoinConfiguration type should be provided.

```yaml
apiVersion: kubeadm.k8s.io/v1beta4
kind: JoinConfiguration
discovery:
  bootstrapToken:
    apiServerEndpoint: some-address:6443
    token: abcdef.0123456789abcdef
    unsafeSkipCAVerification: true
  tlsBootstrapToken: abcdef.0123456789abcdef
```

The JoinConfiguration type should be used to configure runtime settings, that in case of kubeadm join are the discovery method used for accessing the cluster info and all the setting which are specific to the node where kubeadm is executed, including:

- `nodeRegistration`, that holds fields that relate to registering the new node to the cluster; use it to customize the node name, the CRI socket to use or any other settings that should apply to this node only (e.g. the node ip).
- `apiEndpoint`, that represents the endpoint of the instance of the API server to be eventually deployed on this node.

Kubeadm reset configuration types

When executing `kubeadm reset` with the `--config` option, the `ResetConfiguration` type should be provided.

```yaml
apiVersion: kubeadm.k8s.io/v1beta4
kind: ResetConfiguration
...
```

Kubeadm upgrade configuration types

When executing `kubeadm upgrade` with the `--config` option, the `UpgradeConfiguration` type should be provided.

```yaml
apiVersion: kubeadm.k8s.io/v1beta4
kind: UpgradeConfiguration
apply:
  ...
diff:
  ...
node:
  ...
plan:
  ...
```

The `UpgradeConfiguration` structure includes a few substructures that only apply to different subcommands of `kubeadm upgrade`. For example, the `apply` substructure will be used with the `kubeadm upgrade apply` subcommand and all other substructures will be ignored in such a case.

Resource Types

`BootstrapToken`

Appears in:

- InitConfiguration
- InitConfiguration

BootstrapToken describes one bootstrap token, stored as a Secret in the cluster

Field	Description
`token` **[Required]** BootstrapTokenString	`token` is used for establishing bidirectional trust between nodes and control-planes. Used for joining nodes in the cluster.
`description` `string`	`description` sets a human-friendly message why this token exists and what it's used for, so other administrators can know its purpose.
`ttl` meta/v1.Duration	`ttl` defines the time to live for this token. Defaults to `24h`. `expires` and `ttl` are mutually exclusive.
`expires` meta/v1.Time	`expires` specifies the timestamp when this token expires. Defaults to being set dynamically at runtime based on the `ttl`. `expires` and `ttl` are mutually exclusive.
`usages` `[]string`	`usages` describes the ways in which this token can be used. Can by default be used for establishing bidirectional trust, but that can be changed here.

Field	Description
groups []string	groups specifies the extra groups that this token will authenticate as when/if used for authentication

BootstrapTokenString

Appears in:

- BootstrapToken

BootstrapTokenString is a token of the format abcdef.abcdef0123456789 that is used for both validation of the practically of the API server from a joining node's point of view and as an authentication method for the node in the bootstrap phase of "kubeadm join". This token is and should be short-lived.

Field	Description
- **[Required]** string	No description provided.
- **[Required]** string	No description provided.

ClusterConfiguration

ClusterConfiguration contains cluster-wide configuration for a kubeadm cluster.

Field	Description
apiVersion string	kubeadm.k8s.io/v1beta4
kind string	ClusterConfiguration
etcd Etcd	etcd holds the configuration for etcd.
networking Networking	networking holds configuration for the networking topology of the cluster.
kubernetesVersion string	kubernetesVersion is the target version of the control plane.
controlPlaneEndpoint string	controlPlaneEndpoint sets a stable IP address or DNS name for the control plane; It can be a valid IP address or a RFC-1123 DNS subdomain, both with optional TCP port. In case the controlPlaneEndpoint is not specified, the advertiseAddress + bindPort are used; in case the controlPlaneEndpoint is specified but without a TCP port, the bindPort is used. Possible usages are: • In a cluster with more than one control plane instances, this field should be assigned the address of the external load balancer in front of the control plane instances. • In environments with enforced node recycling, the controlPlaneEndpoint could be used for assigning a stable DNS to the control plane.
apiServer APIServer	apiServer contains extra settings for the API server.
controllerManager ControlPlaneComponent	controllerManager contains extra settings for the controller manager.
scheduler ControlPlaneComponent	scheduler contains extra settings for the scheduler.
dns DNS	dns defines the options for the DNS add-on installed in the cluster.
proxy **[Required]** Proxy	proxy defines the options for the proxy add-on installed in the cluster.
certificatesDir string	certificatesDir specifies where to store or look for all required certificates.
imageRepository string	imageRepository sets the container registry to pull images from. If empty, registry.k8s.io will be used by default. In case of kubernetes version is a CI build (kubernetes version starts with ci/) gcr.io/k8s-staging-ci-images will be used as a default for control plane components and for kube-proxy, while registry.k8s.io will be used for all the other images.
featureGates map[string]bool	featureGates contains the feature gates enabled by the user.
clusterName string	The cluster name.
encryptionAlgorithm EncryptionAlgorithmType	encryptionAlgorithm holds the type of asymmetric encryption algorithm used for keys and certificates. Can be one of "RSA-2048" (default), "RSA-3072", "RSA-4096", "ECDSA-P256" or "ECDSA-P384".
certificateValidityPeriod meta/v1.Duration	certificateValidityPeriod specifies the validity period for a non-CA certificate generated by kubeadm. Default value: `8760h`` (365 days * 24 hours = 1 year)

Field	Description
caCertificateValidityPeriod meta/v1.Duration	caCertificateValidityPeriod specifies the validity period for a CA certificate generated by kubeadm. Default value: 87600h (365 days * 24 hours * 10 = 10 years)

InitConfiguration

InitConfiguration contains a list of elements that is specific "kubeadm init"-only runtime information. kubeadm init-only information. These fields are solely used the first time kubeadm init runs. After that, the information in the fields IS NOT uploaded to the kubeadm-config ConfigMap that is used by kubeadm upgrade for instance. These fields must be omitempty.

Field	Description
apiVersion string	kubeadm.k8s.io/v1beta4
kind string	InitConfiguration
bootstrapTokens []BootstrapToken	bootstrapTokens is respected at kubeadm init time and describes a set of Bootstrap Tokens to create. This information IS NOT uploaded to the kubeadm cluster configmap, partly because of its sensitive nature
dryRun **[Required]** bool	dryRun tells if the dry run mode is enabled, don't apply any change in dry run mode, just out put what would be done.
nodeRegistration NodeRegistrationOptions	nodeRegistration holds fields that relate to registering the new control-plane node to the cluster.
localAPIEndpoint APIEndpoint	localAPIEndpoint represents the endpoint of the API server instance that's deployed on this control plane node. In HA setups, this differs from ClusterConfiguration.controlPlaneEndpoint in the sense that controlPlaneEndpoint is the global endpoint for the cluster, which then loadbalances the requests to each individual API server. This configuration object lets you customize what IP/DNS name and port the local API server advertises it's accessible on. By default, kubeadm tries to auto-detect the IP of the default interface and use that, but in case that process fails you may set the desired value here.
certificateKey string	certificateKey sets the key with which certificates and keys are encrypted prior to being uploaded in a Secret in the cluster during the uploadcerts init phase. The certificate key is a hex encoded string that is an AES key of size 32 bytes.
skipPhases []string	skipPhases is a list of phases to skip during command execution. The list of phases can be obtained with the kubeadm init --help command. The flag --skip-phases takes precedence over this field.
patches Patches	patches contains options related to applying patches to components deployed by kubeadm during kubeadm init.
timeouts Timeouts	timeouts holds various timeouts that apply to kubeadm commands.

JoinConfiguration

JoinConfiguration contains elements describing a particular node.

Field	Description
apiVersion string	kubeadm.k8s.io/v1beta4
kind string	JoinConfiguration
dryRun bool	dryRun tells if the dry run mode is enabled, don't apply any change if it is set, just output what would be done.
nodeRegistration NodeRegistrationOptions	nodeRegistration holds fields that relate to registering the new control-plane node to the cluster
caCertPath string	caCertPath is the path to the SSL certificate authority used to secure communications between node and control-plane. Defaults to "/etc/kubernetes/pki/ca.crt".
discovery **[Required]** Discovery	discovery specifies the options for the kubelet to use during the TLS bootstrap process.
controlPlane JoinControlPlane	controlPlane defines the additional control plane instance to be deployed on the joining node. If nil, no additional control plane instance will be deployed.
skipPhases []string	skipPhases is a list of phases to skip during command execution. The list of phases can be obtained with the kubeadm join --help command. The flag --skip-phases takes precedence over this field.
patches Patches	patches contains options related to applying patches to components deployed by kubeadm during kubeadm join.
timeouts Timeouts	timeouts holds various timeouts that apply to kubeadm commands.

ResetConfiguration

ResetConfiguration contains a list of fields that are specifically `kubeadm reset`-only runtime information.

Field	Description
`apiVersion` string	`kubeadm.k8s.io/v1beta4`
`kind` string	`ResetConfiguration`
`cleanupTmpDir` bool	`cleanupTmpDir` specifies whether the "/etc/kubernetes/tmp" directory should be cleaned during the reset process.
`certificatesDir` string	`certificatesDir` specifies the directory where the certificates are stored. If specified, it will be cleaned during the reset process.
`criSocket` string	`criSocket` is used to retrieve container runtime inforomation and used for the removal of the containers. If `criSocket` is not specified by flag or config file, kubeadm will try to detect one valid CRI socket instead.
`dryRun` bool	`dryRun` tells if the dry run mode is enabled, don't apply any change if it is set and just output what would be done.
`force` bool	The `force` flag instructs kubeadm to reset the node without prompting for confirmation.
`ignorePreflightErrors` `[]`string	`ignorePreflightErrors` provides a list of pre-flight errors to be ignored during the reset process, e.g. `IsPrivilegedUser,Swap`. Value `all` ignores errors from all checks.
`skipPhases` `[]`string	`skipPhases` is a list of phases to skip during command execution. The list of phases can be obtained with the `kubeadm reset phase --help` command.
`unmountFlags` `[]`string	`unmountFlags` is a list of `unmount2()` syscall flags that kubeadm can use when unmounting directories during "reset". This flag can be one of: `"MNT_FORCE"`, `"MNT_DETACH"`, `"MNT_EXPIRE"`, `"UMOUNT_NOFOLLOW"`. By default this list is empty.
`timeouts` Timeouts	Timeouts holds various timeouts that apply to kubeadm commands.

UpgradeConfiguration

UpgradeConfiguration contains a list of options that are specific to `kubeadm upgrade` subcommands.

Field	Description
`apiVersion` string	`kubeadm.k8s.io/v1beta4`
`kind` string	`UpgradeConfiguration`
`apply` UpgradeApplyConfiguration	`apply` holds a list of options that are specific to the `kubeadm upgrade apply` command.
`diff` UpgradeDiffConfiguration	`diff` holds a list of options that are specific to the `kubeadm upgrade diff` command.
`node` UpgradeNodeConfiguration	`node` holds a list of options that are specific to the `kubeadm upgrade node` command.
`plan` UpgradePlanConfiguration	`plan` holds a list of options that are specific to the `kubeadm upgrade plan` command.
`timeouts` Timeouts	`timeouts` holds various timeouts that apply to kubeadm commands.

APIEndpoint

Appears in:

- InitConfiguration
- JoinControlPlane

APIEndpoint struct contains elements of API server instance deployed on a node.

Field	Description
`advertiseAddress` string	`dvertiseAddress` sets the IP address for the API server to advertise.
`bindPort` int32	`bindPort` sets the secure port for the API Server to bind to. Defaults to 6443.

APIServer

Appears in:

- ClusterConfiguration

APIServer holds settings necessary for API server deployments in the cluster

Field	Description
`ControlPlaneComponent` **[Required]** ControlPlaneComponent	(Members of `ControlPlaneComponent` are embedded into this type.) No description provided.
`certSANs` `[]string`	`certSANs` sets extra Subject Alternative Names (SANs) for the API Server signing certificate.

Arg

Appears in:

- ControlPlaneComponent
- LocalEtcd
- NodeRegistrationOptions

Arg represents an argument with a name and a value.

Field	Description
`name` **[Required]** `string`	The name of the argument.
`value` **[Required]** `string`	The value of the argument.

BootstrapTokenDiscovery

Appears in:

- Discovery

BootstrapTokenDiscovery is used to set the options for bootstrap token based discovery.

Field	Description
`token` **[Required]** `string`	`token` is a token used to validate cluster information fetched from the control-plane.
`apiServerEndpoint` `string`	`apiServerEndpoint` is an IP or domain name to the API server from which information will be fetched.
`caCertHashes` `[]string`	`caCertHashes` specifies a set of public key pins to verify when token-based discovery is used. The root CA found during discovery must match one of these values. Specifying an empty set disables root CA pinning, which can be unsafe. Each hash is specified as `<type>:<value>`, where the only currently supported type is "sha256". This is a hex-encoded SHA-256 hash of the Subject Public Key Info (SPKI) object in DER-encoded ASN.1. These hashes can be // calculated using, for example, OpenSSL.
`unsafeSkipCAVerification` `bool`	`unsafeSkipCAVerification` allows token-based discovery without CA verification via `caCertHashes`. This can weaken the security of kubeadm since other nodes can impersonate the control-plane.

ControlPlaneComponent

Appears in:

- ClusterConfiguration
- APIServer

ControlPlaneComponent holds settings common to control plane component of the cluster

Field	Description
`extraArgs` `[]Arg`	`extraArgs` is an extra set of flags to pass to the control plane component. An argument name in this list is the flag name as it appears on the command line except without leading dash(es). Extra arguments will override existing default arguments. Duplicate extra arguments are allowed.
`extraVolumes` `[]HostPathMount`	`extraVolumes` is an extra set of host volumes, mounted to the control plane component.
`extraEnvs` `[]EnvVar`	`extraEnvs` is an extra set of environment variables to pass to the control plane component. Environment variables passed using `extraEnvs` will override any existing environment variables, or `*_proxy` environment variables that kubeadm adds by default.

DNS

Appears in:

- ClusterConfiguration

DNS defines the DNS addon that should be used in the cluster

Field	Description
`ImageMeta` **[Required]** ImageMeta	(Members of `ImageMeta` are embedded into this type.) `imageMeta` allows to customize the image used for the DNS addon.
`disabled` **[Required]** `bool`	`disabled` specifies whether to disable this addon in the cluster.

Discovery

Appears in:

- JoinConfiguration

Discovery specifies the options for the kubelet to use during the TLS Bootstrap process

Field	Description
`bootstrapToken` BootstrapTokenDiscovery	`bootstrapToken` is used to set the options for bootstrap token based discovery. `bootstrapToken` and `file` are mutually exclusive.
`file` FileDiscovery	`file` is used to specify a file or URL to a kubeconfig file from which to load cluster information. `bootstrapToken` and `file` are mutually exclusive.
`tlsBootstrapToken` `string`	`tlsBootstrapToken` is a token used for TLS bootstrapping. If `bootstrapToken` is set, this field is defaulted to `bootstrapToken.token`, but can be overridden. If `file` is set, this field **must be set** in case the KubeConfigFile does not contain any other authentication information.

EncryptionAlgorithmType

(Alias of `string`)

Appears in:

- ClusterConfiguration

EncryptionAlgorithmType can define an asymmetric encryption algorithm type.

EnvVar

Appears in:

- ControlPlaneComponent
- LocalEtcd

EnvVar represents an environment variable present in a Container.

Field	Description
`EnvVar` **[Required]** core/v1.EnvVar	(Members of `EnvVar` are embedded into this type.) No description provided.

Etcd

Appears in:

- ClusterConfiguration

Etcd contains elements describing Etcd configuration.

Field	Description
`local` LocalEtcd	`local` provides configuration knobs for configuring the local etcd instance. `local` and `external` are mutually exclusive.
`external` ExternalEtcd	`external` describes how to connect to an external etcd cluster. `local` and `external` are mutually exclusive.

ExternalEtcd

Appears in:

- Etcd

ExternalEtcd describes an external etcd cluster. Kubeadm has no knowledge of where certificate files live and they must be supplied.

Field	Description
`endpoints` **[Required]** `[]string`	`endpoints` contains the list of etcd members.
`caFile` **[Required]** `string`	`caFile` is an SSL Certificate Authority (CA) file used to secure etcd communication. Required if using a TLS connection.
`certFile` **[Required]** `string`	`certFile` is an SSL certification file used to secure etcd communication. Required if using a TLS connection.
`keyFile` **[Required]** `string`	`keyFile` is an SSL key file used to secure etcd communication. Required if using a TLS connection.

FileDiscovery

Appears in:

- Discovery

FileDiscovery is used to specify a file or URL to a kubeconfig file from which to load cluster information.

Field	Description
`kubeConfigPath` **[Required]** `string`	`kubeConfigPath` is used to specify the actual file path or URL to the kubeconfig file from which to load cluster information.

HostPathMount

Appears in:

- ControlPlaneComponent

HostPathMount contains elements describing volumes that are mounted from the host.

Field	Description
`name` **[Required]** `string`	`name` is the name of the volume inside the Pod template.
`hostPath` **[Required]** `string`	`hostPath` is the path in the host that will be mounted inside the Pod.
`mountPath` **[Required]** `string`	`mountPath` is the path inside the Pod where `hostPath` will be mounted.
`readOnly` `bool`	`readOnly` controls write access to the volume.
`pathType` `core/v1.HostPathType`	`pathType` is the type of the `hostPath`.

ImageMeta

Appears in:

- DNS
- LocalEtcd

ImageMeta allows to customize the image used for components that are not originated from the Kubernetes/Kubernetes release process

Field	Description
`imageRepository` `string`	`imageRepository` sets the container registry to pull images from. if not set, the `imageRepository` defined in ClusterConfiguration will be used instead.
`imageTag` `string`	`imageTag` allows to specify a tag for the image. In case this value is set, kubeadm does not change automatically the version of the above components during upgrades.

JoinControlPlane

Appears in:

- JoinConfiguration

JoinControlPlane contains elements describing an additional control plane instance to be deployed on the joining node.

Field	Description
`localAPIEndpoint` APIEndpoint	`localAPIEndpoint` represents the endpoint of the API server instance to be deployed on this node.
`certificateKey` `string`	`certificateKey` is the key that is used for decryption of certificates after they are downloaded from the Secret upon joining a new control plane node. The corresponding encryption key is in the InitConfiguration. The certificate key is a hex encoded string that is an AES key of size 32 bytes.

LocalEtcd

Appears in:

- Etcd

LocalEtcd describes that kubeadm should run an etcd cluster locally.

Field	Description
`ImageMeta` **[Required]** ImageMeta	(Members of `ImageMeta` are embedded into this type.) ImageMeta allows to customize the container used for etcd
`dataDir` **[Required]** `string`	`dataDir` is the directory etcd will place its data. Defaults to "/var/lib/etcd".
`extraArgs` **[Required]** []Arg	`extraArgs` are extra arguments provided to the etcd binary when run inside a static Pod. An argument name in this list is the flag name as it appears on the command line except without leading dash(es). Extra arguments will override existing default arguments. Duplicate extra arguments are allowed.
`extraEnvs` []EnvVar	`extraEnvs` is an extra set of environment variables to pass to the control plane component. Environment variables passed using `extraEnvs` will override any existing environment variables, or `*_proxy` environment variables that kubeadm adds by default.
`serverCertSANs` `[]string`	`serverCertSANs` sets extra Subject Alternative Names (SANs) for the etcd server signing certificate.
`peerCertSANs` `[]string`	`peerCertSANs` sets extra Subject Alternative Names (SANs) for the etcd peer signing certificate.

Networking

Appears in:

- ClusterConfiguration

Networking contains elements describing cluster's networking configuration.

Field	Description
`serviceSubnet` `string`	`serviceSubnet` is the subnet used by Kubernetes Services. Defaults to "10.96.0.0/12".
`podSubnet` `string`	`podSubnet` is the subnet used by Pods.
`dnsDomain` `string`	`dnsDomain` is the dns domain used by Kubernetes Services. Defaults to "cluster.local".

NodeRegistrationOptions

Appears in:

- InitConfiguration
- JoinConfiguration

NodeRegistrationOptions holds fields that relate to registering a new control-plane or node to the cluster, either via `kubeadm init` or `kubeadm join`.

Field	Description
`name` `string`	`name` is the `.Metadata.Name` field of the Node API object that will be created in this `kubeadm init` or `kubeadm join` operation. This field is also used in the `CommonName` field of the kubelet's

Field	Description
	client certificate to the API server. Defaults to the hostname of the node if not provided.
`criSocket` `string`	`criSocket` is used to retrieve container runtime info. This information will be annotated to the Node API object, for later re-use.
`taints` **[Required]** `[]core/v1.Taint`	`taints` specifies the taints the Node API object should be registered with. If this field is unset, i.e. nil, it will be defaulted with a control-plane taint for control-plane nodes. If you don't want to taint your control-plane node, set this field to an empty list, i.e. `taints: []` in the YAML file. This field is solely used for Node registration.
`kubeletExtraArgs` `[]Arg`	`kubeletExtraArgs` passes through extra arguments to the kubelet. The arguments here are passed to the kubelet command line via the environment file kubeadm writes at runtime for the kubelet to source. This overrides the generic base-level configuration in the `kubelet-config` ConfigMap. Flags have higher priority when parsing. These values are local and specific to the node kubeadm is executing on. An argument name in this list is the flag name as it appears on the command line except without leading dash(es). Extra arguments will override existing default arguments. Duplicate extra arguments are allowed.
`ignorePreflightErrors` `[]string`	`ignorePreflightErrors` provides a slice of pre-flight errors to be ignored when the current node is registered, e.g. 'IsPrivilegedUser,Swap'. Value 'all' ignores errors from all checks.
`imagePullPolicy` `core/v1.PullPolicy`	`imagePullPolicy` specifies the policy for image pulling during kubeadm `init` and `join` operations. The value of this field must be one of "Always", "IfNotPresent" or "Never". If this field is unset kubeadm will default it to "IfNotPresent", or pull the required images if not present on the host.
`imagePullSerial` `bool`	`imagePullSerial` specifies if image pulling performed by kubeadm must be done serially or in parallel. Default: true

`Patches`

Appears in:

- InitConfiguration
- JoinConfiguration
- UpgradeApplyConfiguration
- UpgradeNodeConfiguration

Patches contains options related to applying patches to components deployed by kubeadm.

Field	Description
`directory` `string`	`directory` is a path to a directory that contains files named "target[suffix][+patchtype].extension". For example, "kube-apiserver0+merge.yaml" or just "etcd.json". "target" can be one of "kube-apiserver", "kube-controller-manager", "kube-scheduler", "etcd", "kubeletconfiguration", "corednsdeployment". "patchtype" can be one of "strategic", "merge" or "json" and they match the patch formats supported by kubectl. The default "patchtype" is "strategic". "extension" must be either "json" or "yaml". "suffix" is an optional string that can be used to determine which patches are applied first alpha-numerically.

`Proxy`

Appears in:

- ClusterConfiguration

Proxy defines the proxy addon that should be used in the cluster.

Field	Description
`disabled` **[Required]** `bool`	`disabled` specifies whether to disable this addon in the cluster.

`Timeouts`

Appears in:

- InitConfiguration
- JoinConfiguration
- ResetConfiguration

- UpgradeConfiguration

Timeouts holds various timeouts that apply to kubeadm commands.

Field	Description
`controlPlaneComponentHealthCheck` meta/v1.Duration	`controlPlaneComponentHealthCheck` is the amount of time to wait for a control plane component, such as the API server, to be healthy during `kubeadm init` and `kubeadm join`. Default: 4m
`kubeletHealthCheck` meta/v1.Duration	`kubeletHealthCheck` is the amount of time to wait for the kubelet to be healthy during `kubeadm init` and `kubeadm join`. Default: 4m
`kubernetesAPICall` meta/v1.Duration	`kubernetesAPICall` is the amount of time to wait for the kubeadm client to complete a request to the API server. This applies to all types of methods (GET, POST, etc). Default: 1m
`etcdAPICall` meta/v1.Duration	`etcdAPICall` is the amount of time to wait for the kubeadm etcd client to complete a request to the etcd cluster. Default: 2m
`tlsBootstrap` meta/v1.Duration	`tlsBootstrap` is the amount of time to wait for the kubelet to complete TLS bootstrap for a joining node. Default: 5m
`discovery` meta/v1.Duration	`discovery` is the amount of time to wait for kubeadm to validate the API server identity for a joining node. Default: 5m
`upgradeManifests` **[Required]** meta/v1.Duration	`upgradeManifests` is the timeout for upgrading static Pod manifests Default: 5m

UpgradeApplyConfiguration

Appears in:

- UpgradeConfiguration

UpgradeApplyConfiguration contains a list of configurable options which are specific to the "kubeadm upgrade apply" command.

Field	Description
`kubernetesVersion` string	`kubernetesVersion` is the target version of the control plane.
`allowExperimentalUpgrades` bool	`allowExperimentalUpgrades` instructs kubeadm to show unstable versions of Kubernetes as an upgrade alternative and allows upgrading to an alpha/beta/release candidate version of Kubernetes. Default: false
`allowRCUpgrades` bool	Enable `allowRCUpgrades` will show release candidate versions of Kubernetes as an upgrade alternative and allows upgrading to a release candidate version of Kubernetes.
`certificateRenewal` bool	`certificateRenewal` instructs kubeadm to execute certificate renewal during upgrades. Defaults to true.
`dryRun` bool	`dryRun` tells if the dry run mode is enabled, don't apply any change if it is and just output what would be done.
`etcdUpgrade` bool	`etcdUpgrade` instructs kubeadm to execute etcd upgrade during upgrades. Defaults to true.
`forceUpgrade` bool	`forceUpgrade` flag instructs kubeadm to upgrade the cluster without prompting for confirmation.
`ignorePreflightErrors` `[]`string	`ignorePreflightErrors` provides a slice of pre-flight errors to be ignored during the upgrade process, e.g. `IsPrivilegedUser,Swap`. Value `all` ignores errors from all checks.
`patches` Patches	`patches` contains options related to applying patches to components deployed by kubeadm during `kubeadm upgrade`.
`printConfig` bool	`printConfig` specifies whether the configuration file that will be used in the upgrade should be printed or not.
`skipPhases` **[Required]** `[]`string	`skipPhases` is a list of phases to skip during command execution. NOTE: This field is currently ignored for `kubeadm upgrade apply`, but in the future it will be supported.
`imagePullPolicy` core/v1.PullPolicy	`imagePullPolicy` specifies the policy for image pulling during `kubeadm upgrade apply` operations. The value of this field must be one of "Always", "IfNotPresent" or "Never". If this field is unset kubeadm will default it to "IfNotPresent", or pull the required images if not present on the host.
`imagePullSerial` bool	`imagePullSerial` specifies if image pulling performed by kubeadm must be done serially or in parallel. Default: true

UpgradeDiffConfiguration

Appears in:

- UpgradeConfiguration

UpgradeDiffConfiguration contains a list of configurable options which are specific to the `kubeadm upgrade diff` command.

Field	Description
`kubernetesVersion` `string`	`kubernetesVersion` is the target version of the control plane.
`contextLines` `int`	`diffContextLines` is the number of lines of context in the diff.

`UpgradeNodeConfiguration`

Appears in:

- UpgradeConfiguration

UpgradeNodeConfiguration contains a list of configurable options which are specific to the "kubeadm upgrade node" command.

Field	Description
`certificateRenewal` `bool`	`certificateRenewal` instructs kubeadm to execute certificate renewal during upgrades. Defaults to true.
`dryRun` `bool`	`dryRun` tells if the dry run mode is enabled, don't apply any change if it is and just output what would be done.
`etcdUpgrade` `bool`	`etcdUpgrade` instructs kubeadm to execute etcd upgrade during upgrades. Defaults to true.
`ignorePreflightErrors` `[]string`	`ignorePreflightErrors` provides a slice of pre-flight errors to be ignored during the upgrade process, e.g. 'IsPrivilegedUser,Swap'. Value 'all' ignores errors from all checks.
`skipPhases` `[]string`	`skipPhases` is a list of phases to skip during command execution. The list of phases can be obtained with the `kubeadm upgrade node phase --help` command.
`patches` `Patches`	`patches` contains options related to applying patches to components deployed by kubeadm during `kubeadm upgrade`.
`imagePullPolicy` `core/v1.PullPolicy`	`imagePullPolicy` specifies the policy for image pulling during `kubeadm upgrade node` operations. The value of this field must be one of "Always", "IfNotPresent" or "Never". If this field is unset kubeadm will default it to "IfNotPresent", or pull the required images if not present on the host.
`imagePullSerial` `bool`	`imagePullSerial` specifies if image pulling performed by kubeadm must be done serially or in parallel. Default: true

`UpgradePlanConfiguration`

Appears in:

- UpgradeConfiguration

UpgradePlanConfiguration contains a list of configurable options which are specific to the "kubeadm upgrade plan" command.

Field	Description
`kubernetesVersion` **[Required]** `string`	`kubernetesVersion` is the target version of the control plane.
`allowExperimentalUpgrades` `bool`	`allowExperimentalUpgrades` instructs kubeadm to show unstable versions of Kubernetes as an upgrade alternative and allows upgrading to an alpha/beta/release candidate version of Kubernetes. Default: false
`allowRCUpgrades` `bool`	Enable `allowRCUpgrades` will show release candidate versions of Kubernetes as an upgrade alternative and allows upgrading to a release candidate version of Kubernetes.
`dryRun` `bool`	`dryRun` tells if the dry run mode is enabled, don't apply any change if it is and just output what would be done.
`ignorePreflightErrors` `[]string`	`ignorePreflightErrors` provides a slice of pre-flight errors to be ignored during the upgrade process, e.g. 'IsPrivilegedUser,Swap'. Value 'all' ignores errors from all checks.
`printConfig` `bool`	`printConfig` specifies whether the configuration file that will be used in the upgrade should be printed or not.

---

kube-apiserver Configuration (v1beta1)

Package v1beta1 is the v1beta1 version of the API.

Resource Types

- AuthenticationConfiguration
- AuthorizationConfiguration
- EgressSelectorConfiguration

- TracingConfiguration

TracingConfiguration

Appears in:

- KubeletConfiguration
- TracingConfiguration
- TracingConfiguration

TracingConfiguration provides versioned configuration for OpenTelemetry tracing clients.

Field	Description
`endpoint` string	Endpoint of the collector this component will report traces to. The connection is insecure, and does not currently support TLS. Recommended is unset, and endpoint is the otlp grpc default, localhost:4317.
`samplingRatePerMillion` int32	SamplingRatePerMillion is the number of samples to collect per million spans. Recommended is unset. If unset, sampler respects its parent span's sampling rate, but otherwise never samples.

AuthenticationConfiguration

AuthenticationConfiguration provides versioned configuration for authentication.

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1beta1`
`kind` string	`AuthenticationConfiguration`
`jwt` **[Required]** []JWTAuthenticator	jwt is a list of authenticator to authenticate Kubernetes users using JWT compliant tokens. The authenticator will attempt to parse a raw ID token, verify it's been signed by the configured issuer. The public key to verify the signature is discovered from the issuer's public endpoint using OIDC discovery. For an incoming token, each JWT authenticator will be attempted in the order in which it is specified in this list. Note however that other authenticators may run before or after the JWT authenticators. The specific position of JWT authenticators in relation to other authenticators is neither defined nor stable across releases. Since each JWT authenticator must have a unique issuer URL, at most one JWT authenticator will attempt to cryptographically validate the token. The minimum valid JWT payload must contain the following claims: { "iss": "https://issuer.example.com", "aud": ["audience"], "exp": 1234567890, "": "username" }
`anonymous` **[Required]** AnonymousAuthConfig	If present --anonymous-auth must not be set

AuthorizationConfiguration

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1beta1`
`kind` string	`AuthorizationConfiguration`
`authorizers` **[Required]** []AuthorizerConfiguration	Authorizers is an ordered list of authorizers to authorize requests against. This is similar to the --authorization-modes kube-apiserver flag Must be at least one.

EgressSelectorConfiguration

EgressSelectorConfiguration provides versioned configuration for egress selector clients.

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1beta1`
`kind` string	`EgressSelectorConfiguration`
`egressSelections` **[Required]** []EgressSelection	connectionServices contains a list of egress selection client configurations

TracingConfiguration

TracingConfiguration provides versioned configuration for tracing clients.

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1beta1`
`kind` string	`TracingConfiguration`
`TracingConfiguration` **[Required]** `TracingConfiguration`	(Members of `TracingConfiguration` are embedded into this type.) Embed the component config tracing configuration struct

`AnonymousAuthCondition`

Appears in:

- AnonymousAuthConfig

AnonymousAuthCondition describes the condition under which anonymous auth should be enabled.

Field	Description
`path` **[Required]** `string`	Path for which anonymous auth is enabled.

`AnonymousAuthConfig`

Appears in:

- AuthenticationConfiguration

AnonymousAuthConfig provides the configuration for the anonymous authenticator.

Field	Description
`enabled` **[Required]** `bool`	No description provided.
`conditions` **[Required]** `[]AnonymousAuthCondition`	If set, anonymous auth is only allowed if the request meets one of the conditions.

`AudienceMatchPolicyType`

(Alias of `string`)

Appears in:

- Issuer

AudienceMatchPolicyType is a set of valid values for issuer.audienceMatchPolicy

`AuthorizerConfiguration`

Appears in:

- AuthorizationConfiguration

Field	Description
`type` **[Required]** `string`	Type refers to the type of the authorizer "Webhook" is supported in the generic API server Other API servers may support additional authorizer types like Node, RBAC, ABAC, etc.
`name` **[Required]** `string`	Name used to describe the webhook This is explicitly used in monitoring machinery for metrics Note: Names must be DNS1123 labels like `myauthorizername` or subdomains like `myauthorizer.example.domain` Required, with no default
`webhook` **[Required]** `WebhookConfiguration`	Webhook defines the configuration for a Webhook authorizer Must be defined when Type=Webhook Must not be defined when Type!=Webhook

`ClaimMappings`

Appears in:

- JWTAuthenticator

ClaimMappings provides the configuration for claim mapping

Field	Description
`username` **[Required]** [PrefixedClaimOrExpression](#)	username represents an option for the username attribute. The claim's value must be a singular string. Same as the --oidc-username-claim and --oidc-username-prefix flags. If username.expression is set, the expression must produce a string value. If username.expression uses 'claims.email', then 'claims.email_verified' must be used in username.expression or extra[*].valueExpression or claimValidationRules[*].expression. An example claim validation rule expression that matches the validation automatically applied when username.claim is set to 'email' is 'claims.?email_verified.orValue(true) == true'. By explicitly comparing the value to true, we let type-checking see the result will be a boolean, and to make sure a non-boolean email_verified claim will be caught at runtime. In the flag based approach, the --oidc-username-claim and --oidc-username-prefix are optional. If --oidc-username-claim is not set, the default value is "sub". For the authentication config, there is no defaulting for claim or prefix. The claim and prefix must be set explicitly. For claim, if --oidc-username-claim was not set with legacy flag approach, configure username.claim="sub" in the authentication config. For prefix: (1) --oidc-username-prefix="-", no prefix was added to the username. For the same behavior using authentication config, set username.prefix="" (2) --oidc-username-prefix="" and --oidc-username-claim != "email", prefix was "<value of --oidc-issuer-url>#". For the same behavior using authentication config, set username.prefix="#" (3) --oidc-username-prefix="". For the same behavior using authentication config, set username.prefix=""
`groups` [PrefixedClaimOrExpression](#)	groups represents an option for the groups attribute. The claim's value must be a string or string array claim. If groups.claim is set, the prefix must be specified (and can be the empty string). If groups.expression is set, the expression must produce a string or string array value. "", [], and null values are treated as the group mapping not being present.
`uid` [ClaimOrExpression](#)	uid represents an option for the uid attribute. Claim must be a singular string claim. If uid.expression is set, the expression must produce a string value.
`extra` [][ExtraMapping](#)	extra represents an option for the extra attribute. expression must produce a string or string array value. If the value is empty, the extra mapping will not be present. hard-coded extra key/value • key: "foo" valueExpression: "'bar'" This will result in an extra attribute - foo: ["bar"] hard-coded key, value copying claim value • key: "foo" valueExpression: "claims.some_claim" This will result in an extra attribute - foo: [value of some_claim] hard-coded key, value derived from claim value • key: "admin" valueExpression: '(has(claims.is_admin) && claims.is_admin) ? "true":""' This will result in: • if is_admin claim is present and true, extra attribute - admin: ["true"] • if is_admin claim is present and false or is_admin claim is not present, no extra attribute will be added

`ClaimOrExpression`

Appears in:

- ClaimMappings

ClaimOrExpression provides the configuration for a single claim or expression.

Field	Description
`claim` `string`	claim is the JWT claim to use. Either claim or expression must be set. Mutually exclusive with expression.
`expression` `string`	expression represents the expression which will be evaluated by CEL. CEL expressions have access to the contents of the token claims, organized into CEL variable: • 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim.

`ClaimValidationRule`

Appears in:

- JWTAuthenticator

ClaimValidationRule provides the configuration for a single claim validation rule.

Field	Description
`claim` `string`	claim is the name of a required claim. Same as --oidc-required-claim flag. Only string claim keys are supported. Mutually exclusive with expression and message.
`requiredValue` `string`	requiredValue is the value of a required claim. Same as --oidc-required-claim flag. Only string claim values are supported. If claim is set and requiredValue is not set, the claim must be present with a value set to the empty string. Mutually exclusive with expression and message.
`expression` `string`	expression represents the expression which will be evaluated by CEL. Must produce a boolean. CEL expressions have access to the contents of the token claims, organized into CEL variable: - 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Must return true for the validation to pass. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim and requiredValue.
`message` `string`	message customizes the returned error message when expression returns false. message is a literal string. Mutually exclusive with claim and requiredValue.

`Connection`

Appears in:

- EgressSelection

Connection provides the configuration for a single egress selection client.

Field	Description
`proxyProtocol` **[Required]** `ProtocolType`	Protocol is the protocol used to connect from client to the konnectivity server.
`transport` `Transport`	Transport defines the transport configurations we use to dial to the konnectivity server. This is required if ProxyProtocol is HTTPConnect or GRPC.

`EgressSelection`

Appears in:

- EgressSelectorConfiguration

EgressSelection provides the configuration for a single egress selection client.

Field	Description
`name` **[Required]** `string`	name is the name of the egress selection. Currently supported values are "controlplane", "master", "etcd" and "cluster" The "master" egress selector is deprecated in favor of "controlplane"
`connection` **[Required]** `Connection`	connection is the exact information used to configure the egress selection

`EgressSelectorType`

(Alias of `string`)

Appears in:

- Issuer

EgressSelectorType is an indicator of which egress selection should be used for sending traffic.

`ExtraMapping`

Appears in:

- ClaimMappings

ExtraMapping provides the configuration for a single extra mapping.

Field	Description
`key` **[Required]** `string`	key is a string to use as the extra attribute key. key must be a domain-prefix path (e.g. example.org/foo). All characters before the first "/" must be a valid subdomain as defined by RFC 1123. All characters trailing the first "/" must be valid HTTP Path characters as defined by RFC 3986. key must be lowercase. Required to be unique.
`valueExpression` **[Required]** `string`	valueExpression is a CEL expression to extract extra attribute value. valueExpression must produce a string or string array value. "", [], and null values are treated as the extra mapping not being present. Empty string values contained within a string array are filtered out. CEL expressions have access to the contents of the token claims, organized into CEL variable: • 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/

`Issuer`

Appears in:

- JWTAuthenticator

Issuer provides the configuration for an external provider's specific settings.

Field	Description
`url` **[Required]** `string`	url points to the issuer URL in a format https://url or https://url/path. This must match the "iss" claim in the presented JWT, and the issuer returned from discovery. Same value as the --oidc-issuer-url flag. Discovery information is fetched from "{url}/.well-known/openid-configuration" unless overridden by discoveryURL. Required to be unique across all JWT authenticators. Note that egress selection configuration is not used for this network connection. discoveryURL, if specified, overrides the URL used to fetch discovery information instead of using "{url}/.well-known/openid-configuration". The exact value specified is used, so "/.well-known/openid-configuration" must be included in discoveryURL if needed. The "issuer" field in the fetched discovery information must match the "issuer.url" field in the AuthenticationConfiguration and will be used to validate the "iss" claim in the presented JWT. This is for scenarios where the well-known and jwks endpoints are hosted at a different location than the issuer (such as locally in the cluster).
`discoveryURL` `string`	Example: A discovery url that is exposed using kubernetes service 'oidc' in namespace 'oidc-namespace' and discovery information is available at '/.well-known/openid-configuration'. discoveryURL: "https://oidc.oidc-namespace/.well-known/openid-configuration" certificateAuthority is used to verify the TLS connection and the hostname on the leaf certificate must be set to 'oidc.oidc-namespace'. curl https://oidc.oidc-namespace/.well-known/openid-configuration (.discoveryURL field) { issuer: "https://oidc.example.com" (.url field) } discoveryURL must be different from url. Required to be unique across all JWT authenticators. Note that egress selection configuration is not used for this network connection.
`certificateAuthority` `string`	certificateAuthority contains PEM-encoded certificate authority certificates used to validate the connection when fetching discovery information. If unset, the system verifier is used. Same value as the content of the file referenced by the --oidc-ca-file flag.
`audiences` **[Required]** `[]string`	audiences is the set of acceptable audiences the JWT must be issued to. At least one of the entries must match the "aud" claim in presented JWTs. Same value as the --oidc-client-id flag (though this field supports an array). Required to be non-empty.
`audienceMatchPolicy` `AudienceMatchPolicyType`	audienceMatchPolicy defines how the "audiences" field is used to match the "aud" claim in the presented JWT. Allowed values are: 1. "MatchAny" when multiple audiences are specified and 2. empty (or unset) or "MatchAny" when a single audience is specified. • MatchAny: the "aud" claim in the presented JWT must match at least one of the entries in the "audiences" field. For example, if "audiences" is ["foo", "bar"], the "aud" claim in the presented JWT must contain either "foo" or "bar" (and may contain both). • "": The match policy can be empty (or unset) when a single audience is specified in the "audiences" field. The "aud" claim in the presented JWT must contain the single audience

Field	Description
	(and may contain others). For more nuanced audience validation, use claimValidationRules. example: claimValidationRule[].expression: 'sets.equivalent(claims.aud, ["bar", "foo", "baz"])' to require an exact match.
`egressSelectorType` [EgressSelectorType](#)	egressSelectorType is an indicator of which egress selection should be used for sending all traffic related to this issuer (discovery, JWKS, distributed claims, etc). If unspecified, no custom dialer is used. When specified, the valid choices are "controlplane" and "cluster". These correspond to the associated values in the --egress-selector-config-file. • controlplane: for traffic intended to go to the control plane. • cluster: for traffic intended to go to the system being managed by Kubernetes.

`JWTAuthenticator`

Appears in:

- AuthenticationConfiguration

JWTAuthenticator provides the configuration for a single JWT authenticator.

Field	Description
`issuer` **[Required]** [Issuer](#)	issuer contains the basic OIDC provider connection options.
`claimValidationRules` [][ClaimValidationRule](#)	claimValidationRules are rules that are applied to validate token claims to authenticate users.
`claimMappings` **[Required]** [ClaimMappings](#)	claimMappings points claims of a token to be treated as user attributes.
`userValidationRules` [][UserValidationRule](#)	userValidationRules are rules that are applied to final user before completing authentication. These allow invariants to be applied to incoming identities such as preventing the use of the system: prefix that is commonly used by Kubernetes components. The validation rules are logically ANDed together and must all return true for the validation to pass.

`PrefixedClaimOrExpression`

Appears in:

- ClaimMappings

PrefixedClaimOrExpression provides the configuration for a single prefixed claim or expression.

Field	Description
`claim` `string`	claim is the JWT claim to use. Mutually exclusive with expression.
`prefix` `string`	prefix is prepended to claim's value to prevent clashes with existing names. prefix needs to be set if claim is set and can be the empty string. Mutually exclusive with expression.
`expression` `string`	expression represents the expression which will be evaluated by CEL. CEL expressions have access to the contents of the token claims, organized into CEL variable: • 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim and prefix.

`ProtocolType`

(Alias of `string`)

Appears in:

- Connection

ProtocolType is a set of valid values for Connection.ProtocolType

TCPTransport

Appears in:

- Transport

TCPTransport provides the information to connect to konnectivity server via TCP

Field	Description
`url` **[Required]** `string`	URL is the location of the konnectivity server to connect to. As an example it might be "https://127.0.0.1:8131"
`tlsConfig` `TLSConfig`	TLSConfig is the config needed to use TLS when connecting to konnectivity server

TLSConfig

Appears in:

- TCPTransport

TLSConfig provides the authentication information to connect to konnectivity server Only used with TCPTransport

Field	Description
`caBundle` `string`	caBundle is the file location of the CA to be used to determine trust with the konnectivity server. Must be absent/empty if TCPTransport.URL is prefixed with http:// If absent while TCPTransport.URL is prefixed with https://, default to system trust roots.
`clientKey` `string`	clientKey is the file location of the client key to be used in mtls handshakes with the konnectivity server. Must be absent/empty if TCPTransport.URL is prefixed with http:// Must be configured if TCPTransport.URL is prefixed with https://
`clientCert` `string`	clientCert is the file location of the client certificate to be used in mtls handshakes with the konnectivity server. Must be absent/empty if TCPTransport.URL is prefixed with http:// Must be configured if TCPTransport.URL is prefixed with https://

Transport

Appears in:

- Connection

Transport defines the transport configurations we use to dial to the konnectivity server

Field	Description
`tcp` `TCPTransport`	TCP is the TCP configuration for communicating with the konnectivity server via TCP ProxyProtocol of GRPC is not supported with TCP transport at the moment Requires at least one of TCP or UDS to be set
`uds` `UDSTransport`	UDS is the UDS configuration for communicating with the konnectivity server via UDS Requires at least one of TCP or UDS to be set

UDSTransport

Appears in:

- Transport

UDSTransport provides the information to connect to konnectivity server via UDS

Field	Description
`udsName` **[Required]** `string`	UDSName is the name of the unix domain socket to connect to konnectivity server This does not use a unix:// prefix. (Eg: /etc/srv/kubernetes/konnectivity-server/konnectivity-server.socket)

UserValidationRule

Appears in:

- JWTAuthenticator

UserValidationRule provides the configuration for a single user info validation rule.

Field	Description
`expression` **[Required]** `string`	expression represents the expression which will be evaluated by CEL. Must return true for the validation to pass. CEL expressions have access to the contents of UserInfo, organized into CEL variable: - 'user' - authentication.k8s.io/v1, Kind=UserInfo object Refer to https://github.com/kubernetes/api/blob/release-1.28/authentication/v1/types.go#L105-L122 for the definition. API documentation: https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.28/#userinfo-v1-authentication-k8s-io Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/
`message` `string`	message customizes the returned error message when rule returns false. message is a literal string.

`WebhookConfiguration`

Appears in:

- AuthorizerConfiguration

Field	Description
`authorizedTTL` **[Required]** `meta/v1.Duration`	The duration to cache 'authorized' responses from the webhook authorizer. Same as setting `--authorization-webhook-cache-authorized-ttl` flag Default: 5m0s
`cacheAuthorizedRequests` `bool`	CacheAuthorizedRequests specifies whether authorized requests should be cached. If set to true, the TTL for cached decisions can be configured via the AuthorizedTTL field. Default: true
`unauthorizedTTL` **[Required]** `meta/v1.Duration`	The duration to cache 'unauthorized' responses from the webhook authorizer. Same as setting `--authorization-webhook-cache-unauthorized-ttl` flag Default: 30s
`cacheUnauthorizedRequests` `bool`	CacheUnauthorizedRequests specifies whether unauthorized requests should be cached. If set to true, the TTL for cached decisions can be configured via the UnauthorizedTTL field. Default: true
`timeout` **[Required]** `meta/v1.Duration`	Timeout for the webhook request Maximum allowed value is 30s. Required, no default value.
`subjectAccessReviewVersion` **[Required]** `string`	The API version of the authorization.k8s.io SubjectAccessReview to send to and expect from the webhook. Same as setting `--authorization-webhook-version` flag Valid values: v1beta1, v1 Required, no default value
`matchConditionSubjectAccessReviewVersion` **[Required]** `string`	MatchConditionSubjectAccessReviewVersion specifies the SubjectAccessReview version the CEL expressions are evaluated against Valid values: v1 Required, no default value
`failurePolicy` **[Required]** `string`	Controls the authorization decision when a webhook request fails to complete or returns a malformed response or errors evaluating matchConditions. Valid values: - NoOpinion: continue to subsequent authorizers to see if one of them allows the request - Deny: reject the request without consulting subsequent authorizers Required, with no default.
`connectionInfo` **[Required]** `WebhookConnectionInfo`	ConnectionInfo defines how we talk to the webhook

`matchConditions` **[Required]**
`[]WebhookMatchCondition`
	matchConditions is a list of conditions that must be met for a request to be sent to this webhook. An empty list of matchConditions matches all requests. There are a maximum of 64 match conditions allowed.
	
	The exact matching logic is (in order):
	
	1. If at least one matchCondition evaluates to FALSE, then the webhook is skipped.
	2. If ALL matchConditions evaluate to TRUE, then the webhook is called.
	3. If at least one matchCondition evaluates to an error (but none are FALSE):
	    - If failurePolicy=Deny, then the webhook rejects the request
	    - If failurePolicy=NoOpinion, then the error is ignored and the webhook is skipped

`WebhookConnectionInfo`

Appears in:

- WebhookConfiguration

Field	Description
`type` **[Required]** `string`	Controls how the webhook should communicate with the server. Valid values: • KubeConfigFile: use the file specified in kubeConfigFile to locate the server. • InClusterConfig: use the in-cluster configuration to call the SubjectAccessReview API hosted by kube-apiserver. This mode is not allowed for kube-apiserver.
`kubeConfigFile` **[Required]** `string`	Path to KubeConfigFile for connection info Required, if connectionInfo.Type is KubeConfig

`WebhookMatchCondition`

Appears in:

- WebhookConfiguration

Field	Description
`expression` **[Required]** `string`	expression represents the expression which will be evaluated by CEL. Must evaluate to bool. CEL expressions have access to the contents of the SubjectAccessReview in v1 version. If version specified by subjectAccessReviewVersion in the request variable is v1beta1, the contents would be converted to the v1 version before evaluating the CEL expression. • 'resourceAttributes' describes information for a resource access request and is unset for non-resource requests. e.g. has(request.resourceAttributes) && request.resourceAttributes.namespace == 'default' • 'nonResourceAttributes' describes information for a non-resource access request and is unset for resource requests. e.g. has(request.nonResourceAttributes) && request.nonResourceAttributes.path == '/healthz'. • 'user' is the user to test for. e.g. request.user == 'alice' • 'groups' is the groups to test for. e.g. ('group1' in request.groups) • 'extra' corresponds to the user.Info.GetExtra() method from the authenticator. • 'uid' is the information about the requesting user. e.g. request.uid == '1' Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/

Configuration APIs

Client Authentication (v1)

Client Authentication (v1beta1)

Event Rate Limit Configuration (v1alpha1)

Image Policy API (v1alpha1)

kube-apiserver Admission (v1)

kube-apiserver Audit Configuration (v1)

kube-apiserver Configuration (v1)

kube-apiserver Configuration (v1alpha1)

kube-apiserver Configuration (v1beta1)

kube-controller-manager Configuration (v1alpha1)

kube-proxy Configuration (v1alpha1)

kube-scheduler Configuration (v1)

kubeadm Configuration (v1beta3)

kubeadm Configuration (v1beta4)

Feature Gates

This page contains an overview of the various feature gates an administrator can specify on different Kubernetes components.

See feature stages for an explanation of the stages for a feature.

Overview

Feature gates are a set of key=value pairs that describe Kubernetes features. You can turn these features on or off using the `--feature-gates` command line flag on each Kubernetes component.

Each Kubernetes component lets you enable or disable a set of feature gates that are relevant to that component. Use `-h` flag to see a full set of feature gates for all components. To set feature gates for a component, such as kubelet, use the `--feature-gates` flag assigned to a list of feature pairs:

```
--feature-gates=...,GracefulNodeShutdown=true
```

The following tables are a summary of the feature gates that you can set on different Kubernetes components.

- The "Since" column contains the Kubernetes release when a feature is introduced or its release stage is changed.
- The "Until" column, if not empty, contains the last Kubernetes release in which you can still use a feature gate.
- If a feature is in the Alpha or Beta state, you can find the feature listed in the Alpha/Beta feature gate table.
- If a feature is stable you can find all stages for that feature listed in the Graduated/Deprecated feature gate table.
- The Graduated/Deprecated feature gate table also lists deprecated and withdrawn features.

Note:

For a reference to old feature gates that are removed, please refer to feature gates removed.

Feature gates for Alpha or Beta features

Feature	Default	Stage	Since	Until
`AllowParsingUserUIDFromCertAuth`	`false`	Alpha	1.33	1.33
`AllowParsingUserUIDFromCertAuth`	`true`	Beta	1.34	–
`AllowUnsafeMalformedObjectDeletion`	`false`	Alpha	1.32	–
`APIResponseCompression`	`false`	Alpha	1.7	1.15
`APIResponseCompression`	`true`	Beta	1.16	–
`APIServerIdentity`	`false`	Alpha	1.20	1.25
`APIServerIdentity`	`true`	Beta	1.26	–
`APIServingWithRoutine`	`false`	Alpha	1.30	–
`CBORServingAndStorage`	`false`	Alpha	1.32	–
`ClearingNominatedNodeNameAfterBinding`	`false`	Alpha	1.34	1.34
`CloudControllerManagerWebhook`	`false`	Alpha	1.27	–
`ClusterTrustBundle`	`false`	Alpha	1.27	1.32
`ClusterTrustBundle`	`false`	Beta	1.33	–
`ClusterTrustBundleProjection`	`false`	Alpha	1.29	1.32
`ClusterTrustBundleProjection`	`false`	Beta	1.33	–
`ComponentFlagz`	`false`	Alpha	1.32	–
`ComponentStatusz`	`false`	Alpha	1.32	–

Feature	Default	Stage	Since	Until
ConcurrentWatchObjectDecode	false	Beta	1.31	–
ContainerCheckpoint	false	Alpha	1.25	1.29
ContainerCheckpoint	true	Beta	1.30	–
ContainerRestartRules	false	Alpha	1.34	–
ContainerStopSignals	false	Alpha	1.33	–
ContextualLogging	false	Alpha	1.24	–
ContextualLogging	true	Beta	1.30	–
CoordinatedLeaderElection	false	Alpha	1.31	1.32
CoordinatedLeaderElection	false	Beta	1.33	–
CPUManagerPolicyAlphaOptions	false	Alpha	1.23	–
CPUManagerPolicyBetaOptions	true	Beta	1.23	–
CrossNamespaceVolumeDataSource	false	Alpha	1.26	–
CSIVolumeHealth	false	Alpha	1.21	–
CustomCPUCFSQuotaPeriod	false	Alpha	1.12	–
DeclarativeValidation	true	Beta	1.33	–
DeclarativeValidationTakeover	false	Beta	1.33	–
DeploymentReplicaSetTerminatingReplicas	false	Alpha	1.33	–
DetectCacheInconsistency	true	Beta	1.34	–
DRAAdminAccess	false	Alpha	1.32	1.33
DRAAdminAccess	true	Beta	1.34	–
DRAConsumableCapacity	false	Alpha	1.34	–
DRADeviceBindingConditions	false	Alpha	1.34	–
DRADeviceTaints	false	Alpha	1.33	–
DRAExtendedResource	false	Alpha	1.34	–
DRAPartitionableDevices	false	Alpha	1.33	–
DRAPrioritizedList	false	Alpha	1.33	1.33
DRAPrioritizedList	true	Beta	1.34	–
DRAResourceClaimDeviceStatus	false	Alpha	1.32	1.32
DRAResourceClaimDeviceStatus	true	Beta	1.33	–
DRASchedulerFilterTimeout	false	Alpha	1.34	–
EnvFiles	false	Alpha	1.34	–
EventedPLEG	false	Alpha	1.26	–
ExternalServiceAccountTokenSigner	false	Alpha	1.32	1.33
ExternalServiceAccountTokenSigner	true	Beta	1.34	–
GracefulNodeShutdown	false	Alpha	1.20	1.20
GracefulNodeShutdown	true	Beta	1.21	–
GracefulNodeShutdownBasedOnPodPriority	false	Alpha	1.23	1.23
GracefulNodeShutdownBasedOnPodPriority	true	Beta	1.24	–
HostnameOverride	false	Alpha	1.34	–
HPAConfigurableTolerance	false	Alpha	1.33	–
HPAScaleToZero	false	Alpha	1.16	–
ImageMaximumGCAge	false	Alpha	1.29	1.29
ImageMaximumGCAge	true	Beta	1.30	–
ImageVolume	false	Alpha	1.31	1.32
ImageVolume	false	Beta	1.33	–
InformerResourceVersion	false	Alpha	1.30	–
InOrderInformers	true	Alpha	1.33	1.33
InOrderInformers	true	Beta	1.34	–
InPlacePodVerticalScaling	false	Alpha	1.27	1.32
InPlacePodVerticalScaling	true	Beta	1.33	–
InPlacePodVerticalScalingExclusiveCPUs	false	Alpha	1.32	–
InPlacePodVerticalScalingExclusiveMemory	false	Alpha	1.34	–
InTreePluginPortworxUnregister	false	Alpha	1.23	–
JobManagedBy	false	Alpha	1.30	1.31
JobManagedBy	true	Beta	1.32	–
KubeletCrashLoopBackOffMax	false	Alpha	1.32	–
KubeletEnsureSecretPulledImages	false	Alpha	1.33	–
KubeletFineGrainedAuthz	false	Alpha	1.32	1.32

Feature	Default	Stage	Since	Until
KubeletFineGrainedAuthz	true	Beta	1.33	–
KubeletInUserNamespace	false	Alpha	1.22	–
KubeletPodResourcesDynamicResources	false	Alpha	1.27	1.33
KubeletPodResourcesDynamicResources	true	Beta	1.34	–
KubeletPodResourcesGet	false	Alpha	1.27	1.33
KubeletPodResourcesGet	true	Beta	1.34	–
KubeletPSI	false	Alpha	1.33	1.33
KubeletPSI	true	Beta	1.34	–
KubeletSeparateDiskGC	false	Alpha	1.29	1.30
KubeletSeparateDiskGC	true	Beta	1.31	–
KubeletServiceAccountTokenForCredentialProviders	false	Alpha	1.33	1.33
KubeletServiceAccountTokenForCredentialProviders	true	Beta	1.34	–
ListFromCacheSnapshot	false	Alpha	1.33	1.33
ListFromCacheSnapshot	true	Beta	1.34	–
LocalStorageCapacityIsolationFSQuotaMonitoring	false	Alpha	1.15	1.30
LocalStorageCapacityIsolationFSQuotaMonitoring	false	Beta	1.31	–
LoggingAlphaOptions	false	Alpha	1.24	–
LoggingBetaOptions	true	Beta	1.24	–
MatchLabelKeysInPodTopologySpread	false	Alpha	1.25	1.26
MatchLabelKeysInPodTopologySpread	true	Beta	1.27	–
MatchLabelKeysInPodTopologySpreadSelectorMerge	true	Beta	1.34	–
MaxUnavailableStatefulSet	false	Alpha	1.24	–
MemoryQoS	false	Alpha	1.22	–
MutableCSINodeAllocatableCount	false	Alpha	1.33	1.33
MutableCSINodeAllocatableCount	false	Beta	1.34	–
MutatingAdmissionPolicy	false	Alpha	1.30	1.33
MutatingAdmissionPolicy	false	Beta	1.34	–
NodeLogQuery	false	Alpha	1.27	1.29
NodeLogQuery	false	Beta	1.30	–
NominatedNodeNameForExpectation	false	Alpha	1.34	–
OpenAPIEnums	false	Alpha	1.23	1.23
OpenAPIEnums	true	Beta	1.24	–
PodAndContainerStatsFromCRI	false	Alpha	1.23	–
PodCertificateRequest	false	Alpha	1.34	–
PodDeletionCost	false	Alpha	1.21	1.21
PodDeletionCost	true	Beta	1.22	–
PodLevelResources	false	Alpha	1.32	1.33
PodLevelResources	true	Beta	1.34	–
PodLogsQuerySplitStreams	false	Alpha	1.32	–
PodObservedGenerationTracking	false	Alpha	1.33	1.33
PodObservedGenerationTracking	true	Beta	1.34	–
PodReadyToStartContainersCondition	false	Alpha	1.28	1.28
PodReadyToStartContainersCondition	true	Beta	1.29	–
PodTopologyLabelsAdmission	false	Alpha	1.33	–
PortForwardWebsockets	false	Alpha	1.30	1.30
PortForwardWebsockets	true	Beta	1.31	–
PreferSameTrafficDistribution	false	Alpha	1.33	1.33
PreferSameTrafficDistribution	true	Beta	1.34	–
PreventStaticPodAPIReferences	true	Beta	1.34	–
ProcMountType	false	Alpha	1.12	1.30
ProcMountType	false	Beta	1.31	1.32
ProcMountType	true	Beta	1.33	–
QOSReserved	false	Alpha	1.11	–
ReduceDefaultCrashLoopBackOffDecay	false	Alpha	1.33	–
RelaxedServiceNameValidation	false	Alpha	1.34	–
ReloadKubeletServerCertificateFile	true	Beta	1.31	–
RemoteRequestHeaderUID	false	Alpha	1.32	–
ResourceHealthStatus	false	Alpha	1.31	–

Feature	Default	Stage	Since	Until
RotateKubeletServerCertificate	false	Alpha	1.7	1.11
RotateKubeletServerCertificate	true	Beta	1.12	–
RuntimeClassInImageCriApi	false	Alpha	1.29	–
SchedulerAsyncAPICalls	true	Beta	1.34	–
SchedulerAsyncPreemption	false	Alpha	1.32	1.32
SchedulerAsyncPreemption	true	Beta	1.33	–
SchedulerPopFromBackoffQ	true	Beta	1.33	–
SELinuxChangePolicy	false	Alpha	1.32	1.32
SELinuxChangePolicy	true	Beta	1.33	–
SELinuxMount	false	Alpha	1.30	1.32
SELinuxMount	false	Beta	1.33	–
SELinuxMountReadWriteOncePod	false	Alpha	1.25	1.26
SELinuxMountReadWriteOncePod	false	Beta	1.27	1.27
SELinuxMountReadWriteOncePod	true	Beta	1.28	–
ServiceAcccountNodeAudienceRestriction	false	Beta	1.32	1.32
ServiceAcccountNodeAudienceRestriction	true	Beta	1.33	–
SizeBasedListCostEstimate	true	Beta	1.24	–
StorageCapacityScoring	false	Alpha	1.33	–
StorageVersionAPI	false	Alpha	1.20	–
StorageVersionHash	false	Alpha	1.14	1.14
StorageVersionHash	true	Beta	1.15	–
StorageVersionMigrator	false	Alpha	1.30	–
StrictIPCIDRValidation	false	Alpha	1.33	–
StructuredAuthenticationConfigurationEgressSelector	true	Beta	1.34	–
SupplementalGroupsPolicy	false	Alpha	1.31	1.32
SupplementalGroupsPolicy	true	Beta	1.33	–
SystemdWatchdog	true	Beta	1.32	–
TokenRequestServiceAccountUIDValidation	true	Beta	1.34	–
TopologyManagerPolicyAlphaOptions	false	Alpha	1.26	–
TopologyManagerPolicyBetaOptions	false	Beta	1.26	1.27
TopologyManagerPolicyBetaOptions	true	Beta	1.28	–
TranslateStreamCloseWebsocketRequests	false	Alpha	1.29	1.29
TranslateStreamCloseWebsocketRequests	true	Beta	1.30	–
UnauthenticatedHTTP2DOSMitigation	false	Beta	1.28	1.28
UnauthenticatedHTTP2DOSMitigation	true	Beta	1.29	–
UnknownVersionInteroperabilityProxy	false	Alpha	1.28	–
UserNamespacesPodSecurityStandards	false	Alpha	1.29	–
UserNamespacesSupport	false	Alpha	1.28	1.29
UserNamespacesSupport	false	Beta	1.30	1.32
UserNamespacesSupport	true	Beta	1.33	–
WatchCacheInitializationPostStartHook	false	Beta	1.31	–
WatchList	false	Alpha	1.27	1.31
WatchList	true	Beta	1.32	1.32
WatchList	false	Beta	1.33	1.33
WatchList	true	Beta	1.34	–
WatchListClient	false	Beta	1.30	–
WindowsCPUAndMemoryAffinity	false	Alpha	1.32	–
WindowsGracefulNodeShutdown	false	Alpha	1.32	1.33
WindowsGracefulNodeShutdown	true	Beta	1.34	–

Feature gates for graduated or deprecated features

Feature	Default	Stage	Since	Until
AllowDNSOnlyNodeCSR	false	Deprecated	1.31	–
AllowInsecureKubeletCertificateSigningRequests	false	Deprecated	1.31	–
AllowServiceLBStatusOnNonLB	false	Deprecated	1.29	–
AnonymousAuthConfigurableEndpoints	false	Alpha	1.31	1.31
AnonymousAuthConfigurableEndpoints	true	Beta	1.32	1.33

Feature	Default	Stage	Since	Until
AnonymousAuthConfigurableEndpoints	true	GA	1.34	–
AnyVolumeDataSource	false	Alpha	1.18	1.23
AnyVolumeDataSource	true	Beta	1.24	1.32
AnyVolumeDataSource	true	GA	1.33	–
APIServerTracing	false	Alpha	1.22	1.26
APIServerTracing	true	Beta	1.27	1.33
APIServerTracing	true	GA	1.34	–
AuthorizeNodeWithSelectors	false	Alpha	1.31	1.31
AuthorizeNodeWithSelectors	true	Beta	1.32	1.33
AuthorizeNodeWithSelectors	true	GA	1.34	–
AuthorizeWithSelectors	false	Alpha	1.31	1.31
AuthorizeWithSelectors	true	Beta	1.32	1.33
AuthorizeWithSelectors	true	GA	1.34	–
BtreeWatchCache	true	Beta	1.32	1.32
BtreeWatchCache	true	GA	1.33	–
ComponentSLIs	false	Alpha	1.26	1.26
ComponentSLIs	true	Beta	1.27	1.31
ComponentSLIs	true	GA	1.32	–
ConsistentListFromCache	false	Alpha	1.28	1.30
ConsistentListFromCache	true	Beta	1.31	1.33
ConsistentListFromCache	true	GA	1.34	–
CPUManagerPolicyOptions	false	Alpha	1.22	1.22
CPUManagerPolicyOptions	true	Beta	1.23	1.32
CPUManagerPolicyOptions	true	GA	1.33	–
CRDValidationRatcheting	false	Alpha	1.28	1.29
CRDValidationRatcheting	true	Beta	1.30	1.32
CRDValidationRatcheting	true	GA	1.33	–
CronJobsScheduledAnnotation	true	Beta	1.28	1.31
CronJobsScheduledAnnotation	true	GA	1.32	–
CSIMigrationPortworx	false	Alpha	1.23	1.24
CSIMigrationPortworx	false	Beta	1.25	1.30
CSIMigrationPortworx	true	Beta	1.31	1.32
CSIMigrationPortworx	true	GA	1.33	–
CustomResourceFieldSelectors	false	Alpha	1.30	1.30
CustomResourceFieldSelectors	true	Beta	1.31	1.31
CustomResourceFieldSelectors	true	GA	1.32	–
DisableAllocatorDualWrite	false	Alpha	1.31	1.32
DisableAllocatorDualWrite	false	Beta	1.33	1.33
DisableAllocatorDualWrite	true	GA	1.34	–
DisableNodeKubeProxyVersion	false	Alpha	1.29	1.30
DisableNodeKubeProxyVersion	true	Beta	1.31.0	1.31.0
DisableNodeKubeProxyVersion	false	Deprecated	1.31.1	–
DisableNodeKubeProxyVersion	false	Deprecated	1.32	1.32
DisableNodeKubeProxyVersion	true	Deprecated	1.33	–
DynamicResourceAllocation	false	Alpha	1.30	1.31
DynamicResourceAllocation	false	Beta	1.32	1.33
DynamicResourceAllocation	true	GA	1.34	–
ElasticIndexedJob	true	Beta	1.27	1.30
ElasticIndexedJob	true	GA	1.31	–
ExecProbeTimeout	true	GA	1.20	–
GitRepoVolumeDriver	false	Deprecated	1.33	–
HonorPVReclaimPolicy	false	Alpha	1.23	1.30
HonorPVReclaimPolicy	true	Beta	1.31	1.32
HonorPVReclaimPolicy	true	GA	1.33	–
InPlacePodVerticalScalingAllocatedStatus	false	Alpha	1.32	1.32
InPlacePodVerticalScalingAllocatedStatus	false	Deprecated	1.33	–
JobBackoffLimitPerIndex	false	Alpha	1.28	1.28
JobBackoffLimitPerIndex	true	Beta	1.29	1.32

Feature	Default	Stage	Since	Until
JobBackoffLimitPerIndex	true	GA	1.33	–
JobPodReplacementPolicy	false	Alpha	1.28	1.28
JobPodReplacementPolicy	true	Beta	1.29	1.33
JobPodReplacementPolicy	true	GA	1.34	–
JobSuccessPolicy	false	Alpha	1.30	1.30
JobSuccessPolicy	true	Beta	1.31	1.32
JobSuccessPolicy	true	GA	1.33	–
KMSv1	true	Deprecated	1.28	1.28
KMSv1	false	Deprecated	1.29	–
KubeletCgroupDriverFromCRI	false	Alpha	1.28	1.30
KubeletCgroupDriverFromCRI	true	Beta	1.31	–
KubeletCgroupDriverFromCRI	true	GA	1.34	–
KubeletTracing	false	Alpha	1.25	1.26
KubeletTracing	true	Beta	1.27	1.33
KubeletTracing	true	GA	1.34	–
LoadBalancerIPMode	false	Alpha	1.29	1.30
LoadBalancerIPMode	true	Beta	1.30	1.31
LoadBalancerIPMode	true	GA	1.32	–
LogarithmicScaleDown	false	Alpha	1.21	1.21
LogarithmicScaleDown	true	Beta	1.22	1.30
LogarithmicScaleDown	true	GA	1.31	–
MatchLabelKeysInPodAffinity	false	Alpha	1.29	1.30
MatchLabelKeysInPodAffinity	true	Beta	1.31	1.32
MatchLabelKeysInPodAffinity	true	GA	1.33	–
MemoryManager	false	Alpha	1.21	1.21
MemoryManager	true	Beta	1.22	1.31
MemoryManager	true	GA	1.32	–
MultiCIDRServiceAllocator	false	Alpha	1.27	1.30
MultiCIDRServiceAllocator	false	Beta	1.31	1.32
MultiCIDRServiceAllocator	true	GA	1.33	–
NFTablesProxyMode	false	Alpha	1.29	1.30
NFTablesProxyMode	true	Beta	1.31	1.32
NFTablesProxyMode	true	GA	1.33	–
NodeInclusionPolicyInPodTopologySpread	false	Alpha	1.25	1.25
NodeInclusionPolicyInPodTopologySpread	true	Beta	1.26	1.32
NodeInclusionPolicyInPodTopologySpread	true	GA	1.33	–
NodeSwap	false	Alpha	1.22	1.27
NodeSwap	false	Beta	1.28	1.29
NodeSwap	true	Beta	1.30	1.33
NodeSwap	true	GA	1.34	–
OrderedNamespaceDeletion	false	Beta	1.30	1.32
OrderedNamespaceDeletion	true	Beta	1.33	1.33
OrderedNamespaceDeletion	true	GA	1.34	–
PodIndexLabel	true	Beta	1.28	1.31
PodIndexLabel	true	GA	1.32	–
PodLifecycleSleepAction	false	Alpha	1.29	1.29
PodLifecycleSleepAction	true	Beta	1.30	1.33
PodLifecycleSleepAction	true	GA	1.34	–
PodLifecycleSleepActionAllowZero	false	Alpha	1.32	1.32
PodLifecycleSleepActionAllowZero	true	Beta	1.33	1.33
PodLifecycleSleepActionAllowZero	true	GA	1.34	–
PodSchedulingReadiness	false	Alpha	1.26	1.26
PodSchedulingReadiness	true	Beta	1.27	1.29
PodSchedulingReadiness	true	GA	1.30	–
RecoverVolumeExpansionFailure	false	Alpha	1.23	1.31
RecoverVolumeExpansionFailure	true	Beta	1.32	1.33
RecoverVolumeExpansionFailure	true	GA	1.34	–
RecursiveReadOnlyMounts	false	Alpha	1.30	1.30

Feature	Default	Stage	Since	Until
RecursiveReadOnlyMounts	true	Beta	1.31	1.32
RecursiveReadOnlyMounts	true	GA	1.33	–
RelaxedDNSSearchValidation	false	Alpha	1.32	1.32
RelaxedDNSSearchValidation	true	Beta	1.33	1.33
RelaxedDNSSearchValidation	true	GA	1.34	–
RelaxedEnvironmentVariableValidation	false	Alpha	1.30	1.31
RelaxedEnvironmentVariableValidation	true	Beta	1.32	1.33
RelaxedEnvironmentVariableValidation	true	GA	1.34	–
ResilientWatchCacheInitialization	true	Beta	1.31	1.33
ResilientWatchCacheInitialization	true	GA	1.34	–
RetryGenerateName	false	Alpha	1.30	1.30
RetryGenerateName	true	Beta	1.31	1.31
RetryGenerateName	true	GA	1.32	–
SchedulerQueueingHints	true	Beta	1.28	1.28
SchedulerQueueingHints	false	Beta	1.29	1.31
SchedulerQueueingHints	true	Beta	1.32	1.33
SchedulerQueueingHints	true	GA	1.34	–
SeparateCacheWatchRPC	true	Beta	1.28	1.32
SeparateCacheWatchRPC	false	Deprecated	1.33	–
SeparateTaintEvictionController	true	Beta	1.29	1.33
SeparateTaintEvictionController	true	GA	1.34	–
ServiceAccountTokenJTI	false	Alpha	1.29	1.29
ServiceAccountTokenJTI	true	Beta	1.30	1.31
ServiceAccountTokenJTI	true	GA	1.32	–
ServiceAccountTokenNodeBinding	false	Alpha	1.29	1.30
ServiceAccountTokenNodeBinding	true	Beta	1.31	1.32
ServiceAccountTokenNodeBinding	true	GA	1.33	–
ServiceAccountTokenNodeBindingValidation	false	Alpha	1.29	1.29
ServiceAccountTokenNodeBindingValidation	true	Beta	1.30	1.31
ServiceAccountTokenNodeBindingValidation	true	GA	1.32	–
ServiceAccountTokenPodNodeInfo	false	Alpha	1.29	1.29
ServiceAccountTokenPodNodeInfo	true	Beta	1.30	1.31
ServiceAccountTokenPodNodeInfo	true	GA	1.32	–
ServiceTrafficDistribution	false	Alpha	1.30	1.30
ServiceTrafficDistribution	true	Beta	1.31	1.32
ServiceTrafficDistribution	true	GA	1.33	–
SidecarContainers	false	Alpha	1.28	1.28
SidecarContainers	true	Beta	1.29	1.32
SidecarContainers	true	GA	1.33	–
SizeMemoryBackedVolumes	false	Alpha	1.20	1.21
SizeMemoryBackedVolumes	true	Beta	1.22	1.31
SizeMemoryBackedVolumes	true	GA	1.32	–
StatefulSetAutoDeletePVC	false	Alpha	1.23	1.26
StatefulSetAutoDeletePVC	true	Beta	1.27	1.31
StatefulSetAutoDeletePVC	true	GA	1.32	–
StatefulSetStartOrdinal	false	Alpha	1.26	1.26
StatefulSetStartOrdinal	true	Beta	1.27	1.30
StatefulSetStartOrdinal	true	GA	1.31	–
StorageNamespaceIndex	true	Beta	1.30	1.32
StorageNamespaceIndex	true	Deprecated	1.33	–
StreamingCollectionEncodingToJSON	true	Beta	1.33	1.33
StreamingCollectionEncodingToJSON	true	GA	1.34	–
StreamingCollectionEncodingToProtobuf	true	Alpha	1.33	1.33
StreamingCollectionEncodingToProtobuf	true	GA	1.34	–
StrictCostEnforcementForVAP	false	Beta	1.30	1.31
StrictCostEnforcementForVAP	true	GA	1.32	–
StrictCostEnforcementForWebhooks	false	Beta	1.31	1.31
StrictCostEnforcementForWebhooks	true	GA	1.32	–

Feature	Default	Stage	Since	Until
`StructuredAuthenticationConfiguration`	`false`	Alpha	1.29	1.29
`StructuredAuthenticationConfiguration`	`true`	Beta	1.30	1.33
`StructuredAuthenticationConfiguration`	`true`	GA	1.34	–
`StructuredAuthorizationConfiguration`	`false`	Alpha	1.29	1.29
`StructuredAuthorizationConfiguration`	`true`	Beta	1.30	1.31
`StructuredAuthorizationConfiguration`	`true`	GA	1.32	–
`TopologyAwareHints`	`false`	Alpha	1.21	1.22
`TopologyAwareHints`	`false`	Beta	1.23	1.23
`TopologyAwareHints`	`true`	Beta	1.24	1.32
`TopologyAwareHints`	`true`	GA	1.33	–
`TopologyManagerPolicyOptions`	`false`	Alpha	1.26	1.27
`TopologyManagerPolicyOptions`	`true`	Beta	1.28	1.31
`TopologyManagerPolicyOptions`	`true`	GA	1.32	–
`VolumeAttributesClass`	`false`	Alpha	1.29	1.30
`VolumeAttributesClass`	`false`	Beta	1.31	1.33
`VolumeAttributesClass`	`true`	GA	1.34	–
`WatchFromStorageWithoutResourceVersion`	`false`	Beta	1.30	1.32
`WatchFromStorageWithoutResourceVersion`	`false`	Deprecated	1.33	–
`WindowsHostNetwork`	`true`	Alpha	1.26	1.32
`WindowsHostNetwork`	`false`	Deprecated	1.33	–
`WinDSR`	`false`	Alpha	1.14	1.32
`WinDSR`	`true`	Beta	1.33	1.33
`WinDSR`	`true`	GA	1.34	–
`WinOverlay`	`false`	Alpha	1.14	1.19
`WinOverlay`	`true`	Beta	1.20	1.33
`WinOverlay`	`true`	GA	1.34	–

Using a feature

Feature stages

A feature can be in *Alpha*, *Beta* or *GA* stage. An *Alpha* feature means:

- Disabled by default.
- Might be buggy. Enabling the feature may expose bugs.
- Support for feature may be dropped at any time without notice.
- The API may change in incompatible ways in a later software release without notice.
- Recommended for use only in short-lived testing clusters, due to increased risk of bugs and lack of long-term support.

A *Beta* feature means:

- Usually enabled by default. Beta API groups are disabled by default.
- The feature is well tested. Enabling the feature is considered safe.
- Support for the overall feature will not be dropped, though details may change.
- The schema and/or semantics of objects may change in incompatible ways in a subsequent beta or stable release. When this happens, we will provide instructions for migrating to the next version. This may require deleting, editing, and re-creating API objects. The editing process may require some thought. This may require downtime for applications that rely on the feature.
- Recommended for only non-business-critical uses because of potential for incompatible changes in subsequent releases. If you have multiple clusters that can be upgraded independently, you may be able to relax this restriction.

Note:

Please do try *Beta* features and give feedback on them! After they exit beta, it may not be practical for us to make more changes.

A *General Availability* (GA) feature is also referred to as a *stable* feature. It means:

- The feature is always enabled; you cannot disable it.
- The corresponding feature gate is no longer needed.
- Stable versions of features will appear in released software for many subsequent versions.

List of feature gates

Each feature gate is designed for enabling/disabling a specific feature.

`AllowDNSOnlyNodeCSR`

Allow kubelet to request a certificate without any Node IP available, only with DNS names.

`AllowInsecureKubeletCertificateSigningRequests`

Disable node admission validation of CertificateSigningRequests for kubelet signers. Unless you disable this feature gate, Kubernetes enforces that new kubelet certificates have a `commonName` matching `system:node:$nodeName`.

`AllowParsingUserUIDFromCertAuth`

When this feature is enabled, the subject name attribute `1.3.6.1.4.1.57683.2` in an X.509 certificate will be parsed as the user UID during certificate authentication.

`AllowServiceLBStatusOnNonLB`

Enables `.status.ingress.loadBalancer` to be set on Services of types other than `LoadBalancer`.

`AllowUnsafeMalformedObjectDeletion`

Enables the cluster operator to identify corrupt resource(s) using the **list** operation, and introduces an option `ignoreStoreReadErrorWithClusterBreakingPotential` that the operator can set to perform unsafe and force **delete** operation of such corrupt resource(s) using the Kubernetes API.

`AnonymousAuthConfigurableEndpoints`

Enable configurable endpoints for anonymous auth for the API server.

`AnyVolumeDataSource`

Enable use of any custom resource as the `DataSource` of a PVC.

`APIResponseCompression`

Compress the API responses for `LIST` or `GET` requests.

`APIServerIdentity`

Assign each API server an ID in a cluster, using a Lease.

`APIServerTracing`

Add support for distributed tracing in the API server. See Traces for Kubernetes System Components for more details.

`APIServingWithRoutine`

This feature gate enables an API server performance improvement: the API server can use separate goroutines (lightweight threads managed by the Go runtime) to serve **watch** requests.

`AuthorizeNodeWithSelectors`

Make the Node authorizer use fine-grained selector authorization.

`AuthorizeWithSelectors`

Allows authorization to use field and label selectors. Enables `fieldSelector` and `labelSelector` fields in the SubjectAccessReview API, passes field and label selector information to authorization webhooks, enables `fieldSelector` and `labelSelector` functions in the authorizer CEL library, and enables checking `fieldSelector` and `labelSelector` fields in authorization webhook `matchConditions`.

`BtreeWatchCache`

When enabled, the API server will replace the legacy HashMap-based *watch cache* with a BTree-based implementation. This replacement may bring performance improvements.

`CBORServingAndStorage`

Enables CBOR as a supported encoding for requests and responses, and as the preferred storage encoding for custom resources.

`ClearingNominatedNodeNameAfterBinding`

Enable clearing `.status.nominatedNodeName` whenever Pods are bound to nodes.

`CloudControllerManagerWebhook`

Enable webhooks in cloud controller manager.

`ClusterTrustBundle`

Enable ClusterTrustBundle objects and kubelet integration.

`ClusterTrustBundleProjection`

`clusterTrustBundle` projected volume sources.

`ComponentFlagz`

Enables the component's flagz endpoint. See zpages for more information.

`ComponentSLIs`

Enable the `/metrics/slis` endpoint on Kubernetes components like kubelet, kube-scheduler, kube-proxy, kube-controller-manager, cloud-controller-manager allowing you to scrape health check metrics.

`ComponentStatusz`

Enables the component's statusz endpoint. See zpages for more information.

`ConcurrentWatchObjectDecode`

Enable concurrent watch object decoding. This is to avoid starving the API server's watch cache when a conversion webhook is installed.

`ConsistentListFromCache`

Enhance Kubernetes API server performance by serving consistent **list** requests directly from its watch cache, improving scalability and response times. To consistent list from cache Kubernetes requires a newer etcd version (v3.4.31+ or v3.5.13+), that includes fixes to watch progress request feature. If older etcd version is provided Kubernetes will automatically detect it and fallback to serving consistent reads from etcd. Progress notifications ensure watch cache is consistent with etcd while reducing the need for resource-intensive quorum reads from etcd.

See the Kubernetes documentation on Semantics for **get** and **list** for more details.

`ContainerCheckpoint`

Enables the kubelet `checkpoint` API. See Kubelet Checkpoint API for more details.

`ContainerRestartRules`

Enables the ability to configure container-level restart policy and restart rules. See Container Restart Policy and Rules for more details.

`ContainerStopSignals`

Enables usage of the StopSignal lifecycle for containers for configuring custom stop signals using which the containers would be stopped.

`ContextualLogging`

Enables extra details in log output of Kubernetes components that support contextual logging.

`CoordinatedLeaderElection`

Enables the behaviors supporting the LeaseCandidate API, and also enables coordinated leader election for the Kubernetes control plane, deterministically.

`CPUManagerPolicyAlphaOptions`

This allows fine-tuning of CPUManager policies, experimental, Alpha-quality options This feature gate guards *a group* of CPUManager options whose quality level is alpha. This feature gate will never graduate to beta or stable.

`CPUManagerPolicyBetaOptions`

This allows fine-tuning of CPUManager policies, experimental, Beta-quality options This feature gate guards *a group* of CPUManager options whose quality level is beta. This feature gate will never graduate to stable.

`CPUManagerPolicyOptions`

Allow fine-tuning of CPUManager policies.

`CRDValidationRatcheting`

Enable updates to custom resources to contain violations of their OpenAPI schema if the offending portions of the resource update did not change. See Validation Ratcheting for more details.

`CronJobsScheduledAnnotation`

Set the scheduled job time as an annotation on Jobs that were created on behalf of a CronJob.

`CrossNamespaceVolumeDataSource`

Enable the usage of cross namespace volume data source to allow you to specify a source namespace in the `dataSourceRef` field of a PersistentVolumeClaim.

`CSIMigrationPortworx`

Enables shims and translation logic to route volume operations from the Portworx in-tree plugin to Portworx CSI plugin. Requires Portworx CSI driver to be installed and configured in the cluster.

`CSIVolumeHealth`

Enable support for CSI volume health monitoring on node.

`CustomCPUCFSQuotaPeriod`

Enable nodes to change `cpuCFSQuotaPeriod` in kubelet config.

`CustomResourceFieldSelectors`

Enable `selectableFields` in the CustomResourceDefinition API to allow filtering of custom resource **list**, **watch** and **deletecollection** requests.

`DeclarativeValidation`

Enables declarative validation of in-tree Kubernetes APIs. When enabled, APIs with declarative validation rules (defined using IDL tags in the Go code) will have both the generated declarative validation code and the original hand-written validation code executed. The results are compared, and any discrepancies are reported via the `declarative_validation_mismatch_total` metric. Only the hand-written validation result is returned to the user (eg: actually validates in the request path). The original hand-written validation are still the authoritative validations when this is enabled but this can be changed if the DeclarativeValidationTakeover feature gate is enabled in addition to this gate. This feature gate only operates on the kube-apiserver.

`DeclarativeValidationTakeover`

When enabled, along with the DeclarativeValidation feature gate, declarative validation errors are returned directly to the caller, replacing hand-written validation errors for rules that have declarative implementations. When disabled (and `DeclarativeValidation` is enabled), hand-written validation errors are always returned, effectively putting declarative validation in a **mismatch validation mode** that monitors but does not affect API responses. This **mismatch validation mode** allows for the monitoring of the `declarative_validation_mismatch_total` and `declarative_validation_panic_total` metrics which are implementation details for a safer rollout, average user shouldn't need to interact with it directly. This feature gate only operates on the kube-apiserver. Note: Although declarative validation aims for functional equivalence with hand-written validation, the exact description of error messages may differ between the two approaches.

`DeploymentReplicaSetTerminatingReplicas`

Enables a new status field `.status.terminatingReplicas` in Deployments and ReplicaSets to allow tracking of terminating pods.

`DetectCacheInconsistency`

Enable cache inconsistency detection in the API server.

`DisableAllocatorDualWrite`

You can enable the `MultiCIDRServiceAllocator` feature gate. The API server supports migration from the old bitmap ClusterIP allocators to the new IPAddress allocators.

The API server performs a dual-write on both allocators. This feature gate disables the dual write on the new Cluster IP allocators; you can enable this feature gate if you have completed the relevant stage of the migration.

`DisableNodeKubeProxyVersion`

Disable setting the `kubeProxyVersion` field of the Node.

`DRAAdminAccess`

Enables support for requesting admin access in a ResourceClaim or a ResourceClaimTemplate. Admin access grants access to in-use devices and may enable additional permissions when making the device available in a container. Starting with Kubernetes v1.33, only users authorized to create ResourceClaim or ResourceClaimTemplate objects in namespaces labeled with `resource.kubernetes.io/admin-access: "true"` (case-sensitive) can use the `adminAccess` field. This ensures that non-admin users cannot misuse the feature. Starting with Kubernetes v1.34, this label has been updated to `resource.kubernetes.io/admin-access: "true"`.

This feature gate has no effect unless you also enable the `DynamicResourceAllocation` feature gate.

`DRAConsumableCapacity`

Enables device sharing across multiple ResourceClaims or requests.

Additionally, if a device supports sharing, its resource (capacity) can be managed through a defined sharing policy.

`DRADeviceBindingConditions`

Enables support for DeviceBindingConditions in the DRA related fields. This allows for thorough device readiness checks and attachment processes before Bind phase.

`DRADeviceTaints`

Enables support for tainting devices and selectively tolerating those taints when using dynamic resource allocation to manage devices.

This feature gate has no effect unless you also enable the `DynamicResourceAllocation` feature gate.

`DRAExtendedResource`

Enables support for the Extended Resource allocation by DRA feature. It makes it possible to specify an extended resource name in a DeviceClass.

This feature gate has no effect unless the `DynamicResourceAllocation` feature gate is enabled.

`DRAPartitionableDevices`

Enables support for requesting Partitionable Devices for DRA. This lets drivers advertise multiple devices that maps to the same resources of a physical device.

This feature gate has no effect unless you also enable the `DynamicResourceAllocation` feature gate.

`DRAPrioritizedList`

Enables support for the Prioritized List feature. It makes it possible to specify a prioritized list of subrequests for requests in a ResourceClaim.

This feature gate has no effect unless you also enable the `DynamicResourceAllocation` feature gate.

`DRAResourceClaimDeviceStatus`

Enables support the ResourceClaim.status.devices field and for setting this status from DRA drivers. It requires the `DynamicResourceAllocation` feature gate to be enabled.

`DRASchedulerFilterTimeout`

Enables aborting the per-node filter operation in the scheduler after a certain time (10 seconds by default, configurable in the DynamicResources scheduler plugin configuration).

`DynamicResourceAllocation`

Enables support for resources with custom parameters and a lifecycle that is independent of a Pod. Allocation of resources is handled by the Kubernetes scheduler based on "structured parameters".

`ElasticIndexedJob`

Enables Indexed Jobs to be scaled up or down by mutating both `spec.completions` and `spec.parallelism` together such that `spec.completions == spec.parallelism`. See docs on elastic Indexed Jobs for more details.

`EnvFiles`

Support defining container's Environment Variable Values via File. See Define Environment Variable Values Using An Init Container for more details.

`EventedPLEG`

Enable support for the kubelet to receive container life cycle events from the container runtime via an extension to CRI. (PLEG is an abbreviation for “Pod lifecycle event generator”). For this feature to be useful, you also need to enable support for container lifecycle events in each container runtime running in your cluster. If the container runtime does not announce support for container lifecycle events then the kubelet automatically switches to the legacy generic PLEG mechanism, even if you have this feature gate enabled.

`ExecProbeTimeout`

Ensure kubelet respects exec probe timeouts. This feature gate exists in case any of your existing workloads depend on a now-corrected fault where Kubernetes ignored exec probe timeouts. See readiness probes.

`ExternalServiceAccountTokenSigner`

Enable setting `--service-account-signing-endpoint` to make the kube-apiserver use external signer for token signing and token verifying key management.

`GitRepoVolumeDriver`

: This controls if the `gitRepo` volume plugin is supported or not. The `gitRepo` volume plugin is disabled by default starting v1.33 release. This provides a way for users to enable it.

`GracefulNodeShutdown`

: Enables support for graceful shutdown in kubelet. During a system shutdown, kubelet will attempt to detect the shutdown event and gracefully terminate pods running on the node. See Graceful Node Shutdown for more details.

`GracefulNodeShutdownBasedOnPodPriority`

: Enables the kubelet to check Pod priorities when shutting down a node gracefully.

`HonorPVReclaimPolicy`

: Honor persistent volume reclaim policy when it is `Delete` irrespective of PV-PVC deletion ordering. For more details, check the PersistentVolume deletion protection finalizer documentation.

`HostnameOverride`

: Allows setting any FQDN as the pod's hostname.

`HPAConfigurableTolerance`

: Enables setting a tolerance threshold for HorizontalPodAutoscaler metrics.

`HPAScaleToZero`

: Enables setting `minReplicas` to 0 for `HorizontalPodAutoscaler` resources when using custom or external metrics.

`ImageMaximumGCAge`

: Enables the kubelet configuration field `imageMaximumGCAge`, allowing an administrator to specify the age after which an image will be garbage collected.

`ImageVolume`

: Allow using the `image` volume source in a Pod. This volume source lets you mount a container image as a read-only volume.

`InformerResourceVersion`

: Enables the check over the last synced resource version using the informer.

`InOrderInformers`

: Force the informers to deliver watch stream events in order instead of out of order.

`InPlacePodVerticalScaling`

: Enables in-place Pod vertical scaling.

`InPlacePodVerticalScalingAllocatedStatus`

: Enables the `allocatedResources` field in the container status. This feature requires the `InPlacePodVerticalScaling` gate be enabled as well.

`InPlacePodVerticalScalingExclusiveCPUs`

: Enable resource resizing for containers in Guaranteed pods with integer CPU requests. It applies only in nodes with `InPlacePodVerticalScaling` and `CPUManager` features enabled, and the CPUManager policy set to `static`.

`InPlacePodVerticalScalingExclusiveMemory`

: Allow resource resize for containers in Guaranteed Pods when the memory manager policy is set to `"Static"`. Applies only to nodes with `InPlacePodVerticalScaling` and memory manager features enabled.

`InTreePluginPortworxUnregister`

: Stops registering the Portworx in-tree plugin in kubelet and volume controllers.

`JobBackoffLimitPerIndex`

: Allows specifying the maximal number of pod retries per index in Indexed jobs.

`JobManagedBy`

: Allows to delegate reconciliation of a Job object to an external controller.

`JobPodReplacementPolicy`

Allows you to specify pod replacement for terminating pods in a Job

`JobSuccessPolicy`

Allow users to specify when a Job can be declared as succeeded based on the set of succeeded pods.

`KMSv1`

Enables KMS v1 API for encryption at rest. See Using a KMS Provider for data encryption for more details.

`KubeletCgroupDriverFromCRI`

Enable detection of the kubelet cgroup driver configuration option from the CRI. This feature gate is now on for all clusters. However, it only works on nodes where there is a CRI container runtime that supports the `RuntimeConfig` CRI call. If the CRI supports this feature, the kubelet ignores the `cgroupDriver` configuration setting (or deprecated `--cgroup-driver` command line argument). If the container runtime doesn't support it, the kubelet falls back to using the driver configured using the `cgroupDriver` configuration setting. The kubelet will stop falling back to this configuration in Kubernetes 1.36. Thus, users must upgrade their CRI container runtime to a version that supports the `RuntimeConfig` CRI call by then. Admins can use the metric `kubelet_cri_losing_support` to see if there are any nodes in their cluster that will lose support in 1.36. The following CRI versions support this CRI call:

- containerd: Support was added in v2.0.0
- CRI-O: Support was added in v1.28.0

See Configuring a cgroup driver for more details.

`KubeletCrashLoopBackOffMax`

Enables support for configurable per-node backoff maximums for restarting containers in the `CrashLoopBackOff` state. For more details, check the `crashLoopBackOff.maxContainerRestartPeriod` field in the kubelet config file.

`KubeletEnsureSecretPulledImages`

Ensure that pods requesting an image are authorized to access the image with the provided credentials when the image is already present on the node. See Ensure Image Pull Credential Verification.

`KubeletFineGrainedAuthz`

Enable fine-grained authorization for the kubelet's HTTP(s) API.

`KubeletInUserNamespace`

Enables support for running kubelet in a user namespace. See Running Kubernetes Node Components as a Non-root User.

`KubeletPodResourcesDynamicResources`

Extend the kubelet's pod resources monitoring gRPC API endpoints List and Get to include resources allocated in ResourceClaims via Dynamic Resource Allocation.

`KubeletPodResourcesGet`

Enable the `Get` gRPC endpoint on kubelet's for Pod resources. This API augments the resource allocation reporting.

`KubeletPSI`

Enable kubelet to surface Pressure Stall Information (PSI) metrics in the Summary API and Prometheus metrics.

`KubeletSeparateDiskGC`

The split image filesystem feature enables kubelet to perform garbage collection of images (read-only layers) and/or containers (writeable layers) deployed on separate filesystems.

`KubeletServiceAccountTokenForCredentialProviders`

Enable kubelet to send the service account token bound to the pod for which the image is being pulled to the credential provider plugin.

`KubeletTracing`

Add support for distributed tracing in the kubelet. When enabled, kubelet CRI interface and authenticated http servers are instrumented to generate OpenTelemetry trace spans. See Traces for Kubernetes System Components for more details.

`ListFromCacheSnapshot`

Enables the API server to generate snapshots for the watch cache store and using them to serve LIST requests.

`LoadBalancerIPMode`

Allows setting `ipMode` for Services where `type` is set to `LoadBalancer`. See Specifying IPMode of load balancer status for more information.

`LocalStorageCapacityIsolationFSQuotaMonitoring`

When `LocalStorageCapacityIsolation` is enabled for local ephemeral storage, the backing filesystem for emptyDir volumes supports project quotas, and `UserNamespacesSupport` is enabled, project quotas are used to monitor `emptyDir` volume storage consumption rather than using filesystem walk, ensuring better performance and accuracy.

`LogarithmicScaleDown`

Enable semi-random selection of pods to evict on controller scaledown based on logarithmic bucketing of pod timestamps.

`LoggingAlphaOptions`

Allow fine-tuning of experimental, alpha-quality logging options.

`LoggingBetaOptions`

Allow fine-tuning of experimental, beta-quality logging options.

`MatchLabelKeysInPodAffinity`

Enable the `matchLabelKeys` and `mismatchLabelKeys` fields for pod (anti)affinity.

`MatchLabelKeysInPodTopologySpread`

Enable the `matchLabelKeys` field for Pod topology spread constraints.

`MatchLabelKeysInPodTopologySpreadSelectorMerge`

Enable merging of selectors built from `matchLabelKeys` into `labelSelector` of Pod topology spread constraints. This feature gate can be enabled when `matchLabelKeys` feature is enabled with the `MatchLabelKeysInPodTopologySpread` feature flag.

`MaxUnavailableStatefulSet`

Enables setting the `maxUnavailable` field for the rolling update strategy of a StatefulSet. The field specifies the maximum number of Pods that can be unavailable during the update.

`MemoryManager`

Allows setting memory affinity for a container based on NUMA topology.

`MemoryQoS`

Enable memory protection and usage throttle on pod / container using cgroup v2 memory controller.

`MultiCIDRServiceAllocator`

Track IP address allocations for Service cluster IPs using IPAddress objects.

`MutableCSINodeAllocatableCount`

When this feature gate is enabled, the `.spec.drivers[*].allocatable.count` field of a CSINode becomes mutable, and a new field, `nodeAllocatableUpdatePeriodSeconds`, is available in the CSIDriver object. This allows periodic updates to a node's reported allocatable volume capacity, preventing stateful pods from becoming stuck due to outdated information that the kube-scheduler relies on.

`MutatingAdmissionPolicy`

Enable MutatingAdmissionPolicy support, which allows CEL mutations to be applied during admission control.

For Kubernetes v1.30 and v1.31, this feature gate existed but had no effect.

`NFTablesProxyMode`

Allow running kube-proxy in nftables mode.

`NodeInclusionPolicyInPodTopologySpread`

Enable using `nodeAffinityPolicy` and `nodeTaintsPolicy` in Pod topology spread constraints when calculating pod topology spread skew.

`NodeLogQuery`

Enables querying logs of node services using the `/logs` endpoint.

`NodeSwap`

: Enable the kubelet to allocate swap memory for Kubernetes workloads on a node. Must be used with `KubeletConfiguration.failSwapOn` set to false. For more details, please see swap memory

`NominatedNodeNameForExpectation`

: When enabled, the kube-scheduler uses `.status.nominatedNodeName` to express where a Pod is going to be bound. External components can also write to `.status.nominatedNodeName` for a Pod to provide a suggested placement.

`OpenAPIEnums`

: Enables populating "enum" fields of OpenAPI schemas in the spec returned from the API server.

`OrderedNamespaceDeletion`

: While deleting namespace, the pods resources is going to be deleted before the rest of resources.

`PodAndContainerStatsFromCRI`

: Configure the kubelet to gather container and pod stats from the CRI container runtime rather than gathering them from cAdvisor. As of 1.26, this also includes gathering metrics from CRI and emitting them over `/metrics/cadvisor` (rather than having cAdvisor emit them directly).

`PodCertificateRequest`

: Enable PodCertificateRequest objects and podCertificate projected volume sources.

`PodDeletionCost`

: Enable the Pod Deletion Cost feature which allows users to influence ReplicaSet downscaling order.

`PodIndexLabel`

: Enables the Job controller and StatefulSet controller to add the pod index as a label when creating new pods. See Job completion mode docs and StatefulSet pod index label docs for more details.

`PodLevelResources`

: Enable *Pod level resources*: the ability to specify resource requests and limits at the Pod level, rather than only for specific containers.

`PodLifecycleSleepAction`

: Enables the `sleep` action in Container lifecycle hooks (`preStop` and `postStart`).

`PodLifecycleSleepActionAllowZero`

: Enables setting zero value for the `sleep` action in container lifecycle hooks.

`PodLogsQuerySplitStreams`

: Enable fetching specific log streams (either stdout or stderr) from a container's log streams, using the Pod API.

`PodObservedGenerationTracking`

: Enables the kubelet to set `observedGeneration` in the Pod `.status`, and enables other components to set `observedGeneration` in pod conditions. This feature allows reflecting the `.metadata.generation` of the Pod at the time that the overall status, or some specific condition, was being recorded. Storing it helps avoid risks associated with *lost updates*.

`PodReadyToStartContainersCondition`

: Enable the kubelet to mark the PodReadyToStartContainers condition on pods.

  This feature gate was previously known as `PodHasNetworkCondition`, and the associated condition was named `PodHasNetwork`.

`PodSchedulingReadiness`

: Enable setting `schedulingGates` field to control a Pod's scheduling readiness.

`PodTopologyLabelsAdmission`

: Enables the `PodTopologyLabels` admission plugin. See Pod Topology Labels for details.

`PortForwardWebsockets`

: Allow WebSocket streaming of the portforward sub-protocol (`port-forward`) from clients requesting version v2 (`v2.portforward.k8s.io`) of the sub-protocol.

`PreferSameTrafficDistribution`

Allows usage of the values `PreferSameZone` and `PreferSameNode` in the Service `trafficDistribution` field.

`PreventStaticPodAPIReferences`

Denies Pod admission if static Pods reference other API objects.

`ProcMountType`

Enables control over the type proc mounts for containers by setting the `procMount` field of a Pod's `securityContext`.

`QOSReserved`

Allows resource reservations at the QoS level preventing pods at lower QoS levels from bursting into resources requested at higher QoS levels (memory only for now).

`RecoverVolumeExpansionFailure`

Enables users to edit their PVCs to smaller sizes so as they can recover from previously issued volume expansion failures. See Recovering from Failure when Expanding Volumes for more details.

`RecursiveReadOnlyMounts`

Enables support for recursive read-only mounts. For more details, see read-only mounts.

`ReduceDefaultCrashLoopBackOffDecay`

Enabled reduction of both the initial delay and the maximum delay accrued between container restarts for a node for containers in `CrashLoopBackOff` across the cluster to `1s` initial delay and `60s` maximum delay.

`RelaxedDNSSearchValidation`

Relax the server side validation for the DNS search string (`.spec.dnsConfig.searches`) for containers. For example, with this gate enabled, it is okay to include the _ character in the DNS name search string.

`RelaxedEnvironmentVariableValidation`

Allow almost all printable ASCII characters in environment variables.

`RelaxedServiceNameValidation`

Enables relaxed validation for Service object names, allowing the use of RFC 1123 label names instead of RFC 1035 label names.

This feature allows Service object names to start with a digit.

`ReloadKubeletServerCertificateFile`

Enable the kubelet TLS server to update its certificate if the specified certificate file are changed.

This feature is useful when specifying `tlsCertFile` and `tlsPrivateKeyFile` in kubelet configuration. The feature gate has no effect for other cases such as using TLS bootstrap.

`RemoteRequestHeaderUID`

Enable the API server to accept UIDs (user IDs) via request header authentication. This will also make the `kube-apiserver`'s API aggregator add UIDs via standard headers when forwarding requests to the servers serving the aggregated API.

`ResilientWatchCacheInitialization`

Enables resilient watchcache initialization to avoid controlplane overload.

`ResourceHealthStatus`

Enable the `allocatedResourcesStatus` field within the `.status` for a Pod. The field reports additional details for each container in the Pod, with the health information for each device assigned to the Pod.

This feature applies to devices managed by both Device Plugins and Dynamic Resource Allocation. See Device plugin and unhealthy devices for more details.

`RetryGenerateName`

Enables retrying of object creation when the API server is expected to generate a name.

When this feature is enabled, requests using `generateName` are retried automatically in case the control plane detects a name conflict with an existing object, up to a limit of 8 total attempts.

`RotateKubeletServerCertificate`

Enable the rotation of the server TLS certificate on the kubelet. See kubelet configuration for more details.

`RuntimeClassInImageCriApi`

: Enables images to be pulled based on the runtime class of the pods that reference them.

`SchedulerAsyncAPICalls`

: Change the kube-scheduler to make the entire scheduling cycle free of blocking requests to the Kubernetes API server. Instead, interact with the Kubernetes API using asynchronous code.

`SchedulerAsyncPreemption`

: Enable running some expensive operations within the scheduler, associated with preemption, asynchronously. Asynchronous processing of preemption improves overall Pod scheduling latency.

`SchedulerPopFromBackoffQ`

: Improves scheduling queue behavior by popping pods from the backoffQ when the activeQ is empty. This allows to process potentially schedulable pods ASAP, eliminating a penalty effect of the backoff queue.

`SchedulerQueueingHints`

: Enables scheduler queueing hints, which benefits to reduce the useless requeuing. The scheduler retries scheduling pods if something changes in the cluster that could make the pod scheduled. Queueing hints are internal signals that allow the scheduler to filter the changes in the cluster that are relevant to the unscheduled pod, based on previous scheduling attempts.

`SELinuxChangePolicy`

: Enables `spec.securityContext.seLinuxChangePolicy` field. This field can be used to opt-out from applying the SELinux label to the pod volumes using mount options. This is required when a single volume that supports mounting with SELinux mount option is shared between Pods that have different SELinux labels, such as a privileged and unprivileged Pods.

  Enabling the `SELinuxChangePolicy` feature gate requires the feature gate `SELinuxMountReadWriteOncePod` to be enabled.

`SELinuxMount`

: Speeds up container startup by allowing kubelet to mount volumes for a Pod directly with the correct SELinux label instead of changing each file on the volumes recursively. It widens the performance improvements behind the `SELinuxMountReadWriteOncePod` feature gate by extending the implementation to all volumes.

  Enabling the `SELinuxMount` feature gate requires the feature gates `SELinuxMountReadWriteOncePod` and `SELinuxChangePolicy` to be enabled.

`SELinuxMountReadWriteOncePod`

: Speeds up container startup by allowing kubelet to mount volumes for a Pod directly with the correct SELinux label instead of changing each file on the volumes recursively. The initial implementation focused on ReadWriteOncePod volumes.

`SeparateCacheWatchRPC`

: Allows the API server watch cache to create a watch on a dedicated RPC. This prevents watch cache from being starved by other watches.

`SeparateTaintEvictionController`

: Enables running the *taint based eviction* controller, that performs Taint-based Evictions, as a standalone controller (separate from the *node lifecycle* controller).

`ServiceAcccountNodeAudienceRestriction`

: This gate is used to restrict the audience for which the kubelet can request a service account token for.

`ServiceAccountTokenJTI`

: Controls whether JTIs (UUIDs) are embedded into generated service account tokens, and whether these JTIs are recorded into the Kubernetes audit log for future requests made by these tokens.

`ServiceAccountTokenNodeBinding`

: Controls whether the API server allows binding service account tokens to Node objects.

`ServiceAccountTokenNodeBindingValidation`

: Controls whether the apiserver will validate a Node reference in service account tokens.

`ServiceAccountTokenPodNodeInfo`

: Controls whether the apiserver embeds the node name and uid for the associated node when issuing service account tokens bound to Pod objects.

`ServiceTrafficDistribution`

: Allows usage of the optional `spec.trafficDistribution` field in Services. The field offers a way to express preferences for how traffic is distributed to Service endpoints.

`SidecarContainers`

: Allow setting the `restartPolicy` of an init container to `Always` so that the container becomes a sidecar container (restartable init containers). See Sidecar containers and restartPolicy for more details.

`SizeBasedListCostEstimate`

: Enables APF to use size of objects for estimating request cost.

`SizeMemoryBackedVolumes`

: Enable kubelets to determine the size limit for memory-backed volumes (mainly `emptyDir` volumes).

`StatefulSetAutoDeletePVC`

: Allows the use of the optional `.spec.persistentVolumeClaimRetentionPolicy` field, providing control over the deletion of PVCs in a StatefulSet's lifecycle. See PersistentVolumeClaim retention for more details.

`StatefulSetStartOrdinal`

: Allow configuration of the start ordinal in a StatefulSet. See Start ordinal for more details.

`StorageCapacityScoring`

: The feature gate `VolumeCapacityPriority` was used in v1.32 to support storage that are statically provisioned. Starting from v1.33, the new feature gate `StorageCapacityScoring` replaces the old `VolumeCapacityPriority` gate with added support to dynamically provisioned storage. When `StorageCapacityScoring` is enabled, the VolumeBinding plugin in the kube-scheduler is extended to score Nodes based on the storage capacity on each of them. This feature is applicable to CSI volumes that supported Storage Capacity, including local storage backed by a CSI driver.

`StorageNamespaceIndex`

: Enables a namespace indexer for namespace scoped resources in API server cache to accelerate list operations.

`StorageVersionAPI`

: Enable the storage version API.

`StorageVersionHash`

: Allow API servers to expose the storage version hash in the discovery.

`StorageVersionMigrator`

: Enables storage version migration. See Migrate Kubernetes Objects Using Storage Version Migration for more details.

`StreamingCollectionEncodingToJSON`

: Allow the API server JSON encoder to encode collections item by item, instead of all at once.

`StreamingCollectionEncodingToProtobuf`

: Allow the API server Protobuf encoder to encode collections item by item, instead of all at once.

`StrictCostEnforcementForVAP`

: Apply strict CEL cost validation for ValidatingAdmissionPolicies.

`StrictCostEnforcementForWebhooks`

: Apply strict CEL cost validation for `matchConditions` within admission webhooks.

`StrictIPCIDRValidation`

: Use stricter validation for fields containing IP addresses and CIDR values.

  In particular, with this feature gate enabled, octets within IPv4 addresses are not allowed to have any leading `0`s, and IPv4-mapped IPv6 values (e.g. `::ffff:192.168.0.1`) are forbidden. These sorts of values can potentially cause security problems when different components interpret the same string as referring to different IP addresses (as in CVE-2021-29923).

  This tightening applies only to fields in build-in API kinds, and not to custom resource kinds, values in Kubernetes configuration files, or command-line arguments.

`StructuredAuthenticationConfiguration`

Enable structured authentication configuration for the API server.

`StructuredAuthenticationConfigurationEgressSelector`

Enables Egress Selector in Structured Authentication Configuration.

`StructuredAuthorizationConfiguration`

Enable structured authorization configuration, so that cluster administrators can specify more than one authorization webhook in the API server handler chain.

`SupplementalGroupsPolicy`

Enables support for fine-grained SupplementalGroups control. For more details, see Configure fine-grained SupplementalGroups control for a Pod.

`SystemdWatchdog`

Allow using systemd watchdog to monitor the health status of kubelet. See Kubelet Systemd Watchdog for more details.

`TokenRequestServiceAccountUIDValidation`

This is used to ensure that the UID provided in the TokenRequest matches the UID of the ServiceAccount for which the token is being requested. It helps prevent misuse of the TokenRequest API by ensuring that tokens are only issued for the correct ServiceAccount.

`TopologyAwareHints`

Enables topology aware routing based on topology hints in EndpointSlices. See Topology Aware Hints for more details.

`TopologyManagerPolicyAlphaOptions`

Allow fine-tuning of topology manager policies, experimental, Alpha-quality options. This feature gate guards *a group* of topology manager options whose quality level is alpha. This feature gate will never graduate to beta or stable.

`TopologyManagerPolicyBetaOptions`

Allow fine-tuning of topology manager policies, experimental, Beta-quality options. This feature gate guards *a group* of topology manager options whose quality level is beta. This feature gate will never graduate to stable.

`TopologyManagerPolicyOptions`

Enable fine-tuning of topology manager policies.

`TranslateStreamCloseWebsocketRequests`

Allow WebSocket streaming of the remote command sub-protocol (`exec`, `cp`, `attach`) from clients requesting version 5 (v5) of the sub-protocol.

`UnauthenticatedHTTP2DOSMitigation`

Enables HTTP/2 Denial of Service (DoS) mitigations for unauthenticated clients. Kubernetes v1.28.0 through v1.28.2 do not include this feature gate.

`UnknownVersionInteroperabilityProxy`

Proxy resource requests to the correct peer kube-apiserver when multiple kube-apiservers exist at varied versions. See Mixed version proxy for more information.

`UserNamespacesPodSecurityStandards`

Enable Pod Security Standards policies relaxation for pods that run with namespaces. You must set the value of this feature gate consistently across all nodes in your cluster, and you must also enable `UserNamespacesSupport` to use this feature.

`UserNamespacesSupport`

Enable user namespace support for Pods.

`VolumeAttributesClass`

Enable support for VolumeAttributesClasses. See Volume Attributes Classes for more information.

`WatchCacheInitializationPostStartHook`

Enables post-start-hook for watchcache initialization to be part of readyz (with timeout).

`WatchFromStorageWithoutResourceVersion`

Enables watches without `resourceVersion` to be served from storage.

`WatchList`

Enable support for streaming initial state of objects in watch requests.

`WatchListClient`

Allows an API client to request a stream of data rather than fetching a full list. This functionality is available in `client-go` and requires the WatchList feature to be enabled on the server. If the `WatchList` is not supported on the server, the client will seamlessly fall back to a standard list request.

`WindowsCPUAndMemoryAffinity`

Add CPU and Memory Affinity support to Windows nodes with CPUManager, MemoryManager and topology manager.

`WindowsGracefulNodeShutdown`

Enables support for windows node graceful shutdown in kubelet. During a system shutdown, kubelet will attempt to detect the shutdown event and gracefully terminate pods running on the node. See Graceful Node Shutdown for more details.

`WindowsHostNetwork`

Enables support for joining Windows containers to a hosts' network namespace.

`WinDSR`

Allows kube-proxy to create DSR loadbalancers for Windows.

`WinOverlay`

Allows kube-proxy to run in overlay mode for Windows.

What's next

- The deprecation policy for Kubernetes explains the project's approach to removing features and components.
- Since Kubernetes 1.24, new beta APIs are not enabled by default. When enabling a beta feature, you will also need to enable any associated API resources. For example, to enable a particular resource like `storage.k8s.io/v1beta1/csistoragecapacities`, set `--runtime-config=storage.k8s.io/v1beta1/csistoragecapacities`. See API Versioning for more details on the command line flags.

---

kube-apiserver

Synopsis

The Kubernetes API server validates and configures data for the api objects which include pods, services, replicationcontrollers, and others. The API Server services REST operations and provides the frontend to the cluster's shared state through which all other components interact.

```
kube-apiserver [flags]
```

Options

--admission-control-config-file string

File with admission control configuration.

--advertise-address string

The IP address on which to advertise the apiserver to members of the cluster. This address must be reachable by the rest of the cluster. If blank, the --bind-address will be used. If --bind-address is unspecified, the host's default interface will be used.

--aggregator-reject-forwarding-redirect Default: true

Aggregator reject forwarding redirect response back to client.

--allow-metric-labels stringToString Default: []

The map from metric-label to value allow-list of this label. The key's format is <MetricName>,<LabelName>. The value's format is <allowed_value>,<allowed_value>...e.g. metric1,label1='v1,v2,v3', metric1,label2='v1,v2,v3' metric2,label1='v1,v2,v3'.

--allow-metric-labels-manifest string

The path to the manifest file that contains the allow-list mapping. The format of the file is the same as the flag --allow-metric-labels. Note that the flag --allow-metric-labels will override the manifest file.

--allow-privileged

If true, allow privileged containers. [default=false]

--anonymous-auth Default: true

Enables anonymous requests to the secure port of the API server. Requests that are not rejected by another authentication method are treated as anonymous requests. Anonymous requests have a username of system:anonymous, and a group name of system:unauthenticated.

--api-audiences strings

Identifiers of the API. The service account token authenticator will validate that tokens used against the API are bound to at least one of these audiences. If the --service-account-issuer flag is configured and this flag is not, this field defaults to a single element list containing the issuer URL.

--audit-log-batch-buffer-size int Default: 10000

The size of the buffer to store events before batching and writing. Only used in batch mode.

--audit-log-batch-max-size int Default: 1

The maximum size of a batch. Only used in batch mode.

--audit-log-batch-max-wait duration

The amount of time to wait before force writing the batch that hadn't reached the max size. Only used in batch mode.

--audit-log-batch-throttle-burst int

Maximum number of requests sent at the same moment if ThrottleQPS was not utilized before. Only used in batch mode.

--audit-log-batch-throttle-enable

Whether batching throttling is enabled. Only used in batch mode.

--audit-log-batch-throttle-qps float

Maximum average number of batches per second. Only used in batch mode.

--audit-log-compress

If set, the rotated log files will be compressed using gzip.

--audit-log-format string Default: "json"

Format of saved audits. "legacy" indicates 1-line text format for each event. "json" indicates structured json format. Known formats are legacy,json.

--audit-log-maxage int

The maximum number of days to retain old audit log files based on the timestamp encoded in their filename.

--audit-log-maxbackup int

The maximum number of old audit log files to retain. Setting a value of 0 will mean there's no restriction on the number of files.

--audit-log-maxsize int

The maximum size in megabytes of the audit log file before it gets rotated.

--audit-log-mode string Default: "blocking"

Strategy for sending audit events. Blocking indicates sending events should block server responses. Batch causes the backend to buffer and write events asynchronously. Known modes are batch,blocking,blocking-strict.

--audit-log-path string

If set, all requests coming to the apiserver will be logged to this file. '-' means standard out.

--audit-log-truncate-enabled

Whether event and batch truncating is enabled.

--audit-log-truncate-max-batch-size int Default: 10485760

Maximum size of the batch sent to the underlying backend. Actual serialized size can be several hundreds of bytes greater. If a batch exceeds this limit, it is split into several batches of smaller size.

--audit-log-truncate-max-event-size int Default: 102400

Maximum size of the audit event sent to the underlying backend. If the size of an event is greater than this number, first request and response are removed, and if this doesn't reduce the size enough, event is discarded.

--audit-log-version string Default: "audit.k8s.io/v1"

API group and version used for serializing audit events written to log.

--audit-policy-file string

Path to the file that defines the audit policy configuration.

--audit-webhook-batch-buffer-size int Default: 10000

The size of the buffer to store events before batching and writing. Only used in batch mode.

--audit-webhook-batch-max-size int Default: 400

The maximum size of a batch. Only used in batch mode.

--audit-webhook-batch-max-wait duration Default: 30s

The amount of time to wait before force writing the batch that hadn't reached the max size. Only used in batch mode.

--audit-webhook-batch-throttle-burst int Default: 15

Maximum number of requests sent at the same moment if ThrottleQPS was not utilized before. Only used in batch mode.

--audit-webhook-batch-throttle-enable Default: true

Whether batching throttling is enabled. Only used in batch mode.

--audit-webhook-batch-throttle-qps float Default: 10

Maximum average number of batches per second. Only used in batch mode.

--audit-webhook-config-file string

Path to a kubeconfig formatted file that defines the audit webhook configuration.

--audit-webhook-initial-backoff duration Default: 10s

The amount of time to wait before retrying the first failed request.

--audit-webhook-mode string Default: "batch"

Strategy for sending audit events. Blocking indicates sending events should block server responses. Batch causes the backend to buffer and write events asynchronously. Known modes are batch,blocking,blocking-strict.

--audit-webhook-truncate-enabled

Whether event and batch truncating is enabled.

--audit-webhook-truncate-max-batch-size int Default: 10485760

Maximum size of the batch sent to the underlying backend. Actual serialized size can be several hundreds of bytes greater. If a batch exceeds this limit, it is split into several batches of smaller size.

--audit-webhook-truncate-max-event-size int Default: 102400

Maximum size of the audit event sent to the underlying backend. If the size of an event is greater than this number, first request and response are removed, and if this doesn't reduce the size enough, event is discarded.

--audit-webhook-version string Default: "audit.k8s.io/v1"

API group and version used for serializing audit events written to webhook.

--authentication-config string

File with Authentication Configuration to configure the JWT Token authenticator or the anonymous authenticator. Requires the StructuredAuthenticationConfiguration feature gate. This flag is mutually exclusive with the --oidc-* flags if the file configures the JWT Token authenticator. This flag is mutually exclusive with --anonymous-auth if the file configures the Anonymous authenticator.

--authentication-token-webhook-cache-ttl duration Default: 2m0s

The duration to cache responses from the webhook token authenticator.

--authentication-token-webhook-config-file string

File with webhook configuration for token authentication in kubeconfig format. The API server will query the remote service to determine authentication for bearer tokens.

--authentication-token-webhook-version string Default: "v1beta1"

The API version of the authentication.k8s.io TokenReview to send to and expect from the webhook.

--authorization-config string

File with Authorization Configuration to configure the authorizer chain. Requires feature gate StructuredAuthorizationConfiguration. This flag is mutually exclusive with the other --authorization-mode and --authorization-webhook-* flags.

--authorization-mode strings

Ordered list of plug-ins to do authorization on secure port. Defaults to AlwaysAllow if --authorization-config is not used. Comma-delimited list of: AlwaysAllow,AlwaysDeny,ABAC,Webhook,RBAC,Node.

--authorization-policy-file string

File with authorization policy in json line by line format, used with --authorization-mode=ABAC, on the secure port.

--authorization-webhook-cache-authorized-ttl duration Default: 5m0s

The duration to cache 'authorized' responses from the webhook authorizer.

--authorization-webhook-cache-unauthorized-ttl duration Default: 30s

The duration to cache 'unauthorized' responses from the webhook authorizer.

--authorization-webhook-config-file string

File with webhook configuration in kubeconfig format, used with --authorization-mode=Webhook. The API server will query the remote service to determine access on the API server's secure port.

--authorization-webhook-version string Default: "v1beta1"

The API version of the authorization.k8s.io SubjectAccessReview to send to and expect from the webhook.

--bind-address string Default: 0.0.0.0

The IP address on which to listen for the --secure-port port. The associated interface(s) must be reachable by the rest of the cluster, and by CLI/web clients. If blank or an unspecified address (0.0.0.0 or ::), all interfaces and IP address families will be used.

--cert-dir string Default: "/var/run/kubernetes"

The directory where the TLS certs are located. If --tls-cert-file and --tls-private-key-file are provided, this flag will be ignored.

--client-ca-file string

If set, any request presenting a client certificate signed by one of the authorities in the client-ca-file is authenticated with an identity corresponding to the CommonName of the client certificate.

--contention-profiling

Enable block profiling, if profiling is enabled

--coordinated-leadership-lease-duration duration Default: 15s

The duration of the lease used for Coordinated Leader Election.

--coordinated-leadership-renew-deadline duration Default: 10s

The deadline for renewing a coordinated leader election lease.

--coordinated-leadership-retry-period duration Default: 2s

The period for retrying to renew a coordinated leader election lease.

--cors-allowed-origins strings

List of allowed origins for CORS, comma separated. An allowed origin can be a regular expression to support subdomain matching. If this list is empty CORS will not be enabled. Please ensure each expression matches the entire hostname by anchoring to the start with '^' or including the '//' prefix, and by anchoring to the end with '$' or including the ':' port separator suffix. Examples of valid expressions are '//example.com(:|$)' and '^https://example.com(:|$)'

--debug-socket-path string

Use an unprotected (no authn/authz) unix-domain socket for profiling with the given path

--default-not-ready-toleration-seconds int Default: 300

Indicates the tolerationSeconds of the toleration for notReady:NoExecute that is added by default to every pod that does not already have such a toleration.

--default-unreachable-toleration-seconds int Default: 300

Indicates the tolerationSeconds of the toleration for unreachable:NoExecute that is added by default to every pod that does not already have such a toleration.

--delete-collection-workers int Default: 1

Number of workers spawned for DeleteCollection call. These are used to speed up namespace cleanup.

--disable-admission-plugins strings

admission plugins that should be disabled although they are in the default enabled plugins list (NamespaceLifecycle, LimitRanger, ServiceAccount, TaintNodesByCondition, PodSecurity, Priority, DefaultTolerationSeconds, DefaultStorageClass, StorageObjectInUseProtection, PersistentVolumeClaimResize, RuntimeClass, CertificateApproval, CertificateSigning, ClusterTrustBundleAttest, CertificateSubjectRestriction, DefaultIngressClass, PodTopologyLabels, MutatingAdmissionPolicy, MutatingAdmissionWebhook, ValidatingAdmissionPolicy, ValidatingAdmissionWebhook, ResourceQuota). Comma-delimited list of admission plugins: AlwaysAdmit, AlwaysDeny, AlwaysPullImages, CertificateApproval, CertificateSigning, CertificateSubjectRestriction, ClusterTrustBundleAttest, DefaultIngressClass, DefaultStorageClass, DefaultTolerationSeconds, DenyServiceExternalIPs, EventRateLimit, ExtendedResourceToleration, ImagePolicyWebhook, LimitPodHardAntiAffinityTopology, LimitRanger, MutatingAdmissionPolicy, MutatingAdmissionWebhook, NamespaceAutoProvision, NamespaceExists, NamespaceLifecycle, NodeRestriction, OwnerReferencesPermissionEnforcement, PersistentVolumeClaimResize, PodNodeSelector, PodSecurity, PodTolerationRestriction, PodTopologyLabels, Priority, ResourceQuota, RuntimeClass, ServiceAccount, StorageObjectInUseProtection, TaintNodesByCondition, ValidatingAdmissionPolicy, ValidatingAdmissionWebhook. The order of plugins in this flag does not matter.

--disable-http2-serving

If true, HTTP2 serving will be disabled [default=false]

--disabled-metrics strings

This flag provides an escape hatch for misbehaving metrics. You must provide the fully qualified metric name in order to disable it. Disclaimer: disabling metrics is higher in precedence than showing hidden metrics.

--egress-selector-config-file string

File with apiserver egress selector configuration.

--emulated-version strings

The versions different components emulate their capabilities (APIs, features, ...) of.
If set, the component will emulate the behavior of this version instead of the underlying binary version.
Version format could only be major.minor, for example: '--emulated-version=wardle=1.2,kube=1.31'.
Options are: kube=1.31..1.34(default:1.34)
If the component is not specified, defaults to "kube"

--emulation-forward-compatible

If true, for any beta+ APIs enabled by default or by --runtime-config at the emulation version, their future versions with higher priority/stability will be auto enabled even if they introduced after the emulation version. Can only be set to true if the emulation version is lower than the binary version.

--enable-admission-plugins strings

admission plugins that should be enabled in addition to default enabled ones (NamespaceLifecycle, LimitRanger, ServiceAccount, TaintNodesByCondition, PodSecurity, Priority, DefaultTolerationSeconds, DefaultStorageClass, StorageObjectInUseProtection, PersistentVolumeClaimResize, RuntimeClass, CertificateApproval, CertificateSigning, ClusterTrustBundleAttest,

CertificateSubjectRestriction, DefaultIngressClass, PodTopologyLabels, MutatingAdmissionPolicy, MutatingAdmissionWebhook, ValidatingAdmissionPolicy, ValidatingAdmissionWebhook, ResourceQuota). Comma-delimited list of admission plugins: AlwaysAdmit, AlwaysDeny, AlwaysPullImages, CertificateApproval, CertificateSigning, CertificateSubjectRestriction, ClusterTrustBundleAttest, DefaultIngressClass, DefaultStorageClass, DefaultTolerationSeconds, DenyServiceExternalIPs, EventRateLimit, ExtendedResourceToleration, ImagePolicyWebhook, LimitPodHardAntiAffinityTopology, LimitRanger, MutatingAdmissionPolicy, MutatingAdmissionWebhook, NamespaceAutoProvision, NamespaceExists, NamespaceLifecycle, NodeRestriction, OwnerReferencesPermissionEnforcement, PersistentVolumeClaimResize, PodNodeSelector, PodSecurity, PodTolerationRestriction, PodTopologyLabels, Priority, ResourceQuota, RuntimeClass, ServiceAccount, StorageObjectInUseProtection, TaintNodesByCondition, ValidatingAdmissionPolicy, ValidatingAdmissionWebhook. The order of plugins in this flag does not matter.

--enable-aggregator-routing

Turns on aggregator routing requests to endpoints IP rather than cluster IP.

--enable-bootstrap-token-auth

Enable to allow secrets of type 'bootstrap.kubernetes.io/token' in the 'kube-system' namespace to be used for TLS bootstrapping authentication.

--enable-garbage-collector Default: true

Enables the generic garbage collector. MUST be synced with the corresponding flag of the kube-controller-manager.

--enable-priority-and-fairness Default: true

If true, replace the max-in-flight handler with an enhanced one that queues and dispatches with priority and fairness

--encryption-provider-config string

The file containing configuration for encryption providers to be used for storing secrets in etcd

--encryption-provider-config-automatic-reload

Determines if the file set by --encryption-provider-config should be automatically reloaded if the disk contents change. Setting this to true disables the ability to uniquely identify distinct KMS plugins via the API server healthz endpoints.

--endpoint-reconciler-type string Default: "lease"

Use an endpoint reconciler (master-count, lease, none) master-count is deprecated, and will be removed in a future version.

--etcd-cafile string

SSL Certificate Authority file used to secure etcd communication.

--etcd-certfile string

SSL certification file used to secure etcd communication.

--etcd-compaction-interval duration Default: 5m0s

The interval of compaction requests. If 0, the compaction request from apiserver is disabled.

--etcd-count-metric-poll-period duration Default: 1m0s

Frequency of polling etcd for number of resources per type. 0 disables the metric collection.

--etcd-db-metric-poll-interval duration Default: 30s

The interval of requests to poll etcd and update metric. 0 disables the metric collection

--etcd-healthcheck-timeout duration Default: 2s

The timeout to use when checking etcd health.

--etcd-keyfile string

SSL key file used to secure etcd communication.

--etcd-prefix string Default: "/registry"

The prefix to prepend to all resource paths in etcd.

--etcd-readycheck-timeout duration Default: 2s

The timeout to use when checking etcd readiness

--etcd-servers strings

List of etcd servers to connect with (scheme://ip:port), comma separated.

--etcd-servers-overrides strings

Per-resource etcd servers overrides, comma separated. The individual override format: group/resource#servers, where servers are URLs, semicolon separated. Note that this applies only to resources compiled into this server binary. e.g. "/pods#http://etcd4:2379;http://etcd5:2379,/events#http://etcd6:2379"

--event-ttl duration Default: 1h0m0s

Amount of time to retain events.

--external-hostname string

The hostname to use when generating externalized URLs for this master (e.g. Swagger API Docs or OpenID Discovery).

--feature-gates colonSeparatedMultimapStringString

Comma-separated list of component:key=value pairs that describe feature gates for alpha/experimental features of different components. If the component is not specified, defaults to "kube". This flag can be repeatedly invoked. For example: --feature-gates 'wardle:featureA=true,wardle:featureB=false' --feature-gates 'kube:featureC=true'Options are:

kube:APIResponseCompression=true|false (BETA - default=true)
kube:APIServerIdentity=true|false (BETA - default=true)
kube:APIServingWithRoutine=true|false (ALPHA - default=false)
kube:AllAlpha=true|false (ALPHA - default=false)
kube:AllBeta=true|false (BETA - default=false)
kube:AllowParsingUserUIDFromCertAuth=true|false (BETA - default=true)
kube:AllowUnsafeMalformedObjectDeletion=true|false (ALPHA - default=false)
kube:CBORServingAndStorage=true|false (ALPHA - default=false)
kube:CPUManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
kube:CPUManagerPolicyBetaOptions=true|false (BETA - default=true)
kube:CSIVolumeHealth=true|false (ALPHA - default=false)
kube:ClearingNominatedNodeNameAfterBinding=true|false (ALPHA - default=false)
kube:ClientsAllowCBOR=true|false (ALPHA - default=false)
kube:ClientsPreferCBOR=true|false (ALPHA - default=false)
kube:CloudControllerManagerWebhook=true|false (ALPHA - default=false)
kube:ClusterTrustBundle=true|false (BETA - default=false)
kube:ClusterTrustBundleProjection=true|false (BETA - default=false)
kube:ComponentFlagz=true|false (ALPHA - default=false)
kube:ComponentStatusz=true|false (ALPHA - default=false)
kube:ConcurrentWatchObjectDecode=true|false (BETA - default=false)
kube:ContainerCheckpoint=true|false (BETA - default=true)
kube:ContainerRestartRules=true|false (ALPHA - default=false)
kube:ContainerStopSignals=true|false (ALPHA - default=false)
kube:ContextualLogging=true|false (BETA - default=true)
kube:CoordinatedLeaderElection=true|false (BETA - default=false)
kube:CrossNamespaceVolumeDataSource=true|false (ALPHA - default=false)
kube:CustomCPUCFSQuotaPeriod=true|false (ALPHA - default=false)
kube:DRAAdminAccess=true|false (BETA - default=true)
kube:DRAConsumableCapacity=true|false (ALPHA - default=false)
kube:DRADeviceBindingConditions=true|false (ALPHA - default=false)
kube:DRADeviceTaints=true|false (ALPHA - default=false)
kube:DRAExtendedResource=true|false (ALPHA - default=false)
kube:DRAPartitionableDevices=true|false (ALPHA - default=false)
kube:DRAPrioritizedList=true|false (BETA - default=true)
kube:DRAResourceClaimDeviceStatus=true|false (BETA - default=true)
kube:DRASchedulerFilterTimeout=true|false (BETA - default=true)
kube:DeclarativeValidation=true|false (BETA - default=true)
kube:DeclarativeValidationTakeover=true|false (BETA - default=false)
kube:DeploymentReplicaSetTerminatingReplicas=true|false (ALPHA - default=false)
kube:DetectCacheInconsistency=true|false (BETA - default=true)
kube:DisableCPUQuotaWithExclusiveCPUs=true|false (BETA - default=true)
kube:EnvFiles=true|false (ALPHA - default=false)
kube:EventedPLEG=true|false (ALPHA - default=false)
kube:ExternalServiceAccountTokenSigner=true|false (BETA - default=true)
kube:GracefulNodeShutdown=true|false (BETA - default=true)
kube:GracefulNodeShutdownBasedOnPodPriority=true|false (BETA - default=true)
kube:HPAConfigurableTolerance=true|false (ALPHA - default=false)
kube:HPAScaleToZero=true|false (ALPHA - default=false)
kube:HostnameOverride=true|false (ALPHA - default=false)
kube:ImageMaximumGCAge=true|false (BETA - default=true)
kube:ImageVolume=true|false (BETA - default=false)
kube:InOrderInformers=true|false (BETA - default=true)
kube:InPlacePodVerticalScaling=true|false (BETA - default=true)
kube:InPlacePodVerticalScalingExclusiveCPUs=true|false (ALPHA - default=false)
kube:InPlacePodVerticalScalingExclusiveMemory=true|false (ALPHA - default=false)
kube:InTreePluginPortworxUnregister=true|false (ALPHA - default=false)
kube:InformerResourceVersion=true|false (ALPHA - default=false)
kube:JobManagedBy=true|false (BETA - default=true)
kube:KubeletCrashLoopBackOffMax=true|false (ALPHA - default=false)
kube:KubeletEnsureSecretPulledImages=true|false (ALPHA - default=false)
kube:KubeletFineGrainedAuthz=true|false (BETA - default=true)
kube:KubeletInUserNamespace=true|false (ALPHA - default=false)

kube:KubeletPSI=true|false (BETA - default=true)
kube:KubeletPodResourcesDynamicResources=true|false (BETA - default=true)
kube:KubeletPodResourcesGet=true|false (BETA - default=true)
kube:KubeletSeparateDiskGC=true|false (BETA - default=true)
kube:KubeletServiceAccountTokenForCredentialProviders=true|false (BETA - default=true)
kube:ListFromCacheSnapshot=true|false (BETA - default=true)
kube:LocalStorageCapacityIsolationFSQuotaMonitoring=true|false (BETA - default=false)
kube:LoggingAlphaOptions=true|false (ALPHA - default=false)
kube:LoggingBetaOptions=true|false (BETA - default=true)
kube:MatchLabelKeysInPodTopologySpread=true|false (BETA - default=true)
kube:MatchLabelKeysInPodTopologySpreadSelectorMerge=true|false (BETA - default=true)
kube:MaxUnavailableStatefulSet=true|false (ALPHA - default=false)
kube:MemoryQoS=true|false (ALPHA - default=false)
kube:MutableCSINodeAllocatableCount=true|false (BETA - default=false)
kube:MutatingAdmissionPolicy=true|false (BETA - default=false)
kube:NodeLogQuery=true|false (BETA - default=false)
kube:NominatedNodeNameForExpectation=true|false (ALPHA - default=false)
kube:OpenAPIEnums=true|false (BETA - default=true)
kube:PodAndContainerStatsFromCRI=true|false (ALPHA - default=false)
kube:PodCertificateRequest=true|false (ALPHA - default=false)
kube:PodDeletionCost=true|false (BETA - default=true)
kube:PodLevelResources=true|false (BETA - default=true)
kube:PodLogsQuerySplitStreams=true|false (ALPHA - default=false)
kube:PodObservedGenerationTracking=true|false (BETA - default=true)
kube:PodReadyToStartContainersCondition=true|false (BETA - default=true)
kube:PodTopologyLabelsAdmission=true|false (ALPHA - default=false)
kube:PortForwardWebsockets=true|false (BETA - default=true)
kube:PreferSameTrafficDistribution=true|false (BETA - default=true)
kube:PreventStaticPodAPIReferences=true|false (BETA - default=true)
kube:ProcMountType=true|false (BETA - default=true)
kube:QOSReserved=true|false (ALPHA - default=false)
kube:ReduceDefaultCrashLoopBackOffDecay=true|false (ALPHA - default=false)
kube:RelaxedServiceNameValidation=true|false (ALPHA - default=false)
kube:ReloadKubeletServerCertificateFile=true|false (BETA - default=true)
kube:RemoteRequestHeaderUID=true|false (BETA - default=true)
kube:ResourceHealthStatus=true|false (ALPHA - default=false)
kube:RotateKubeletServerCertificate=true|false (BETA - default=true)
kube:RuntimeClassInImageCriApi=true|false (ALPHA - default=false)
kube:SELinuxChangePolicy=true|false (BETA - default=true)
kube:SELinuxMount=true|false (BETA - default=false)
kube:SELinuxMountReadWriteOncePod=true|false (BETA - default=true)
kube:SchedulerAsyncAPICalls=true|false (BETA - default=true)
kube:SchedulerAsyncPreemption=true|false (BETA - default=true)
kube:SchedulerPopFromBackoffQ=true|false (BETA - default=true)
kube:ServiceAccountNodeAudienceRestriction=true|false (BETA - default=true)
kube:SizeBasedListCostEstimate=true|false (BETA - default=true)
kube:StorageCapacityScoring=true|false (ALPHA - default=false)
kube:StorageVersionAPI=true|false (ALPHA - default=false)
kube:StorageVersionHash=true|false (BETA - default=true)
kube:StorageVersionMigrator=true|false (ALPHA - default=false)
kube:StrictIPCIDRValidation=true|false (ALPHA - default=false)
kube:StructuredAuthenticationConfigurationEgressSelector=true|false (BETA - default=true)
kube:SupplementalGroupsPolicy=true|false (BETA - default=true)
kube:SystemdWatchdog=true|false (BETA - default=true)
kube:TokenRequestServiceAccountUIDValidation=true|false (BETA - default=true)
kube:TopologyManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
kube:TopologyManagerPolicyBetaOptions=true|false (BETA - default=true)
kube:TranslateStreamCloseWebsocketRequests=true|false (BETA - default=true)
kube:UnauthenticatedHTTP2DOSMitigation=true|false (BETA - default=true)
kube:UnknownVersionInteroperabilityProxy=true|false (ALPHA - default=false)
kube:UserNamespacesPodSecurityStandards=true|false (ALPHA - default=false)
kube:UserNamespacesSupport=true|false (BETA - default=true)

kube:WatchCacheInitializationPostStartHook=true|false (BETA - default=false)
kube:WatchList=true|false (BETA - default=true)
kube:WatchListClient=true|false (BETA - default=false)
kube:WindowsCPUAndMemoryAffinity=true|false (ALPHA - default=false)
kube:WindowsGracefulNodeShutdown=true|false (BETA - default=true)

--goaway-chance float

To prevent HTTP/2 clients from getting stuck on a single apiserver, randomly close a connection (GOAWAY). The client's other in-flight requests won't be affected, and the client will reconnect, likely landing on a different apiserver after going through the load balancer again. This argument sets the fraction of requests that will be sent a GOAWAY. Clusters with single apiservers, or which don't use a load balancer, should NOT enable this. Min is 0 (off), Max is .02 (1/50 requests); .001 (1/1000) is a recommended starting point.

-h, --help

help for kube-apiserver

--http2-max-streams-per-connection int

The limit that the server gives to clients for the maximum number of streams in an HTTP/2 connection. Zero means to use golang's default.

--kubelet-certificate-authority string

Path to a cert file for the certificate authority.

--kubelet-client-certificate string

Path to a client cert file for TLS.

--kubelet-client-key string

Path to a client key file for TLS.

--kubelet-preferred-address-types strings Default: "Hostname,InternalDNS,InternalIP,ExternalDNS,ExternalIP"

List of the preferred NodeAddressTypes to use for kubelet connections.

--kubelet-timeout duration Default: 5s

Timeout for kubelet operations.

--kubernetes-service-node-port int

If non-zero, the Kubernetes master service (which apiserver creates/maintains) will be of type NodePort, using this as the value of the port. If zero, the Kubernetes master service will be of type ClusterIP.

--lease-reuse-duration-seconds int Default: 60

The time in seconds that each lease is reused. A lower value could avoid large number of objects reusing the same lease. Notice that a too small value may cause performance problems at storage layer.

--livez-grace-period duration

This option represents the maximum amount of time it should take for apiserver to complete its startup sequence and become live. From apiserver's start time to when this amount of time has elapsed, /livez will assume that unfinished post-start hooks will complete successfully and therefore return true.

--log-flush-frequency duration Default: 5s

Maximum number of seconds between log flushes

--log-text-info-buffer-size quantity

[Alpha] In text format with split output streams, the info messages can be buffered for a while to increase performance. The default value of zero bytes disables buffering. The size can be specified as number of bytes (512), multiples of 1000 (1K), multiples of 1024 (2Ki), or powers of those (3M, 4G, 5Mi, 6Gi). Enable the LoggingAlphaOptions feature gate to use this.

--log-text-split-stream

[Alpha] In text format, write error messages to stderr and info messages to stdout. The default is to write a single stream to stdout. Enable the LoggingAlphaOptions feature gate to use this.

--logging-format string Default: "text"

Sets the log format. Permitted formats: "text".

--max-connection-bytes-per-sec int

If non-zero, throttle each user connection to this number of bytes/sec. Currently only applies to long-running requests.

--max-mutating-requests-inflight int Default: 200

This and --max-requests-inflight are summed to determine the server's total concurrency limit (which must be positive) if --enable-priority-and-fairness is true. Otherwise, this flag limits the maximum number of mutating requests in flight, or a zero value disables the limit completely.

--max-requests-inflight int Default: 400

This and --max-mutating-requests-inflight are summed to determine the server's total concurrency limit (which must be positive) if --enable-priority-and-fairness is true. Otherwise, this flag limits the maximum number of non-mutating requests in flight, or a zero value disables the limit completely.

--min-request-timeout int Default: 1800

An optional field indicating the minimum number of seconds a handler must keep a request open before timing it out. Currently only honored by the watch request handler, which picks a randomized value above this number as the connection timeout, to spread out load.

--oidc-ca-file string

If set, the OpenID server's certificate will be verified by one of the authorities in the oidc-ca-file, otherwise the host's root CA set will be used.

--oidc-client-id string

The client ID for the OpenID Connect client, must be set if oidc-issuer-url is set.

--oidc-groups-claim string

If provided, the name of a custom OpenID Connect claim for specifying user groups. The claim value is expected to be a string or array of strings. This flag is experimental, please see the authentication documentation for further details.

--oidc-groups-prefix string

If provided, all groups will be prefixed with this value to prevent conflicts with other authentication strategies.

--oidc-issuer-url string

The URL of the OpenID issuer, only HTTPS scheme will be accepted. If set, it will be used to verify the OIDC JSON Web Token (JWT).

--oidc-required-claim <comma-separated 'key=value' pairs>

A key=value pair that describes a required claim in the ID Token. If set, the claim is verified to be present in the ID Token with a matching value. Repeat this flag to specify multiple claims.

--oidc-signing-algs strings Default: "RS256"

Comma-separated list of allowed JOSE asymmetric signing algorithms. JWTs with a supported 'alg' header values are: RS256, RS384, RS512, ES256, ES384, ES512, PS256, PS384, PS512. Values are defined by RFC 7518 https://tools.ietf.org/html/rfc7518#section-3.1.

--oidc-username-claim string Default: "sub"

The OpenID claim to use as the user name. Note that claims other than the default ('sub') is not guaranteed to be unique and immutable. This flag is experimental, please see the authentication documentation for further details.

--oidc-username-prefix string

If provided, all usernames will be prefixed with this value. If not provided, username claims other than 'email' are prefixed by the issuer URL to avoid clashes. To skip any prefixing, provide the value '-'.

--peer-advertise-ip string

If set and the UnknownVersionInteroperabilityProxy feature gate is enabled, this IP will be used by peer kube-apiservers to proxy requests to this kube-apiserver when the request cannot be handled by the peer due to version skew between the kube-apiservers. This flag is only used in clusters configured with multiple kube-apiservers for high availability.

--peer-advertise-port string

If set and the UnknownVersionInteroperabilityProxy feature gate is enabled, this port will be used by peer kube-apiservers to proxy requests to this kube-apiserver when the request cannot be handled by the peer due to version skew between the kube-apiservers. This flag is only used in clusters configured with multiple kube-apiservers for high availability.

--peer-ca-file string

If set and the UnknownVersionInteroperabilityProxy feature gate is enabled, this file will be used to verify serving certificates of peer kube-apiservers. This flag is only used in clusters configured with multiple kube-apiservers for high availability.

--permit-address-sharing

If true, SO_REUSEADDR will be used when binding the port. This allows binding to wildcard IPs like 0.0.0.0 and specific IPs in parallel, and it avoids waiting for the kernel to release sockets in TIME_WAIT state. [default=false]

--permit-port-sharing

If true, SO_REUSEPORT will be used when binding the port, which allows more than one instance to bind on the same address and port. [default=false]

--profiling Default: true

Enable profiling via web interface host:port/debug/pprof/

--proxy-client-cert-file string

Client certificate used to prove the identity of the aggregator or kube-apiserver when it must call out during a request. This includes proxying requests to a user api-server and calling out to webhook admission plugins. It is expected that this cert includes a signature from the CA in the --requestheader-client-ca-file flag. That CA is published in the 'extension-apiserver-authentication' configmap in the kube-system namespace. Components receiving calls from kube-aggregator should use that CA to perform their half of the mutual TLS verification.

--proxy-client-key-file string

Private key for the client certificate used to prove the identity of the aggregator or kube-apiserver when it must call out during a request. This includes proxying requests to a user api-server and calling out to webhook admission plugins.

--request-timeout duration Default: 1m0s

An optional field indicating the duration a handler must keep a request open before timing it out. This is the default request timeout for requests but may be overridden by flags such as --min-request-timeout for specific types of requests.

--requestheader-allowed-names strings

List of client certificate common names to allow to provide usernames in headers specified by --requestheader-username-headers. If empty, any client certificate validated by the authorities in --requestheader-client-ca-file is allowed.

--requestheader-client-ca-file string

Root certificate bundle to use to verify client certificates on incoming requests before trusting usernames in headers specified by --requestheader-username-headers. WARNING: generally do not depend on authorization being already done for incoming requests.

--requestheader-extra-headers-prefix strings

List of request header prefixes to inspect. X-Remote-Extra- is suggested.

--requestheader-group-headers strings

List of request headers to inspect for groups. X-Remote-Group is suggested.

--requestheader-uid-headers strings

List of request headers to inspect for UIDs. X-Remote-Uid is suggested. Requires the RemoteRequestHeaderUID feature to be enabled.

--requestheader-username-headers strings

List of request headers to inspect for usernames. X-Remote-User is common.

--runtime-config <comma-separated 'key=value' pairs>

A set of key=value pairs that enable or disable built-in APIs. Supported options are:
v1=true|false for the core API group
<group>/<version>=true|false for a specific API group and version (e.g. apps/v1=true)
api/all=true|false controls all API versions
api/ga=true|false controls all API versions of the form v[0-9]+
api/beta=true|false controls all API versions of the form v[0-9]+beta[0-9]+
api/alpha=true|false controls all API versions of the form v[0-9]+alpha[0-9]+
api/legacy is deprecated, and will be removed in a future version

--runtime-config-emulation-forward-compatible

If true, APIs identified by group/version that are enabled in the --runtime-config flag will be installed even if it is introduced after the emulation version. If false, server would fail to start if any APIs identified by group/version that are enabled in the --runtime-config flag are introduced after the emulation version. Can only be set to true if the emulation version is lower than the binary version.

--secure-port int Default: 6443

The port on which to serve HTTPS with authentication and authorization. It cannot be switched off with 0.

--service-account-extend-token-expiration Default: true

Turns on projected service account expiration extension during token generation, which helps safe transition from legacy token to bound service account token feature. If this flag is enabled, admission injected tokens would be extended up to 1 year to prevent unexpected failure during transition, ignoring value of service-account-max-token-expiration.

--service-account-issuer strings

Identifier of the service account token issuer. The issuer will assert this identifier in "iss" claim of issued tokens. This value is a string or URI. If this option is not a valid URI per the OpenID Discovery 1.0 spec, the ServiceAccountIssuerDiscovery feature will remain disabled, even if the feature gate is set to true. It is highly recommended that this value comply with the OpenID spec: https://openid.net/specs/openid-connect-discovery-1_0.html. In practice, this means that service-account-issuer must be an https URL. It is also highly recommended that this URL be capable of serving OpenID discovery documents at {service-account-issuer}/.well-known/openid-configuration. When this flag is specified multiple times, the first is used to generate tokens and all are used to determine which issuers are accepted.

--service-account-jwks-uri string

Overrides the URI for the JSON Web Key Set in the discovery doc served at /.well-known/openid-configuration. This flag is useful if the discovery doc and key set are served to relying parties from a URL other than the API server's external (as auto-detected or overridden with external-hostname).

--service-account-key-file strings

File containing PEM-encoded x509 RSA or ECDSA private or public keys, used to verify ServiceAccount tokens. The specified file can contain multiple keys, and the flag can be specified multiple times with different files. If unspecified, --tls-private-key-file is used. Must be specified when --service-account-signing-key-file is provided

--service-account-lookup Default: true

If true, validate ServiceAccount tokens exist in etcd as part of authentication.

--service-account-max-token-expiration duration

The maximum validity duration of a token created by the service account token issuer. If an otherwise valid TokenRequest with a validity duration larger than this value is requested, a token will be issued with a validity duration of this value.

--service-account-signing-endpoint string

Path to socket where a external JWT signer is listening. This flag is mutually exclusive with --service-account-signing-key-file and --service-account-key-file. Requires enabling feature gate (ExternalServiceAccountTokenSigner)

--service-account-signing-key-file string

Path to the file that contains the current private key of the service account token issuer. The issuer will sign issued ID tokens with this private key.

--service-cluster-ip-range string

A CIDR notation IP range from which to assign service cluster IPs. This must not overlap with any IP ranges assigned to nodes or pods. Max of two dual-stack CIDRs is allowed.

--service-node-port-range <a string in the form 'N1-N2'> Default: 30000-32767

A port range to reserve for services with NodePort visibility. This must not overlap with the ephemeral port range on nodes. Example: '30000-32767'. Inclusive at both ends of the range.

--show-hidden-metrics-for-version string

The previous version for which you want to show hidden metrics. Only the previous minor version is meaningful, other values will not be allowed. The format is <major>.<minor>, e.g.: '1.16'. The purpose of this format is make sure you have the opportunity to notice if the next release hides additional metrics, rather than being surprised when they are permanently removed in the release after that.

--shutdown-delay-duration duration

Time to delay the termination. During that time the server keeps serving requests normally. The endpoints /healthz and /livez will return success, but /readyz immediately returns failure. Graceful termination starts after this delay has elapsed. This can be used to allow load balancer to stop sending traffic to this server.

--shutdown-send-retry-after

If true the HTTP Server will continue listening until all non long running request(s) in flight have been drained, during this window all incoming requests will be rejected with a status code 429 and a 'Retry-After' response header, in addition 'Connection: close' response header is set in order to tear down the TCP connection when idle.

--shutdown-watch-termination-grace-period duration

This option, if set, represents the maximum amount of grace period the apiserver will wait for active watch request(s) to drain during the graceful server shutdown window.

--storage-backend string

The storage backend for persistence. Options: 'etcd3' (default).

--storage-initialization-timeout duration Default: 1m0s

Maximum amount of time to wait for storage initialization before declaring apiserver ready. Defaults to 1m.

--storage-media-type string Default: "application/vnd.kubernetes.protobuf"

The media type to use to store objects in storage. Some resources or storage backends may only support a specific media type and will ignore this setting. Supported media types: [application/json, application/yaml, application/vnd.kubernetes.protobuf]

--strict-transport-security-directives strings

List of directives for HSTS, comma separated. If this list is empty, then HSTS directives will not be added. Example: 'max-age=31536000,includeSubDomains,preload'

--tls-cert-file string

File containing the default x509 Certificate for HTTPS. (CA cert, if any, concatenated after server cert). If HTTPS serving is enabled, and --tls-cert-file and --tls-private-key-file are not provided, a self-signed certificate and key are generated for the public address and saved to the directory specified by --cert-dir.

--tls-cipher-suites strings

Comma-separated list of cipher suites for the server. If omitted, the default Go cipher suites will be used.
Preferred values: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256.
Insecure values: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_ECDSA_WITH_RC4_128_SHA, TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_RSA_WITH_RC4_128_SHA, TLS_RSA_WITH_3DES_EDE_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_256_CBC_SHA, TLS_RSA_WITH_AES_256_GCM_SHA384, TLS_RSA_WITH_RC4_128_SHA.

--tls-min-version string

Minimum TLS version supported. Possible values: VersionTLS10, VersionTLS11, VersionTLS12, VersionTLS13

--tls-private-key-file string

File containing the default x509 private key matching --tls-cert-file.

--tls-sni-cert-key string

A pair of x509 certificate and private key file paths, optionally suffixed with a list of domain patterns which are fully qualified domain names, possibly with prefixed wildcard segments. The domain patterns also allow IP addresses, but IPs should only be used if the

apiserver has visibility to the IP address requested by a client. If no domain patterns are provided, the names of the certificate are extracted. Non-wildcard matches trump over wildcard matches, explicit domain patterns trump over extracted names. For multiple key/certificate pairs, use the --tls-sni-cert-key multiple times. Examples: "example.crt,example.key" or "foo.crt,foo.key:*.foo.com,foo.com".

--token-auth-file string

If set, the file that will be used to secure the secure port of the API server via token authentication.

--tracing-config-file string

File with apiserver tracing configuration.

-v, --v int

number for the log level verbosity

--version version[=true]

--version, --version=raw prints version information and quits; --version=vX.Y.Z... sets the reported version

--vmodule pattern=N,...

comma-separated list of pattern=N settings for file-filtered logging (only works for text log format)

--watch-cache Default: true

Enable watch caching in the apiserver

--watch-cache-sizes strings

Watch cache size settings for some resources (pods, nodes, etc.), comma separated. The individual setting format: resource[.group]#size, where resource is lowercase plural (no version), group is omitted for resources of apiVersion v1 (the legacy core API) and included for others, and size is a number. This option is only meaningful for resources built into the apiserver, not ones defined by CRDs or aggregated from external servers, and is only consulted if the watch-cache is enabled. The only meaningful size setting to supply here is zero, which means to disable watch caching for the associated resource; all non-zero values are equivalent and mean to not disable watch caching for that resource

---

kuberc (v1alpha1)

Resource Types

- Preference

`Preference`

Preference stores elements of KubeRC configuration file

Field	Description
`apiVersion` string	`kubectl.config.k8s.io/v1alpha1`
`kind` string	`Preference`
`overrides` **[Required]** `[]CommandDefaults`	overrides allows changing default flag values of commands. This is especially useful, when user doesn't want to explicitly set flags each time.

`aliases` **[Required]**
`[]AliasOverride`
	aliases allow defining command aliases for existing kubectl commands, with optional default flag values. If the alias name collides with a built-in command, built-in command always takes precedence. Flag overrides defined in the overrides section do NOT apply to aliases for the same command. kubectl [ALIAS NAME] [USER_FLAGS] [USER_EXPLICIT_ARGS] expands to kubectl [COMMAND] # built-in command alias points to [KUBERC_PREPEND_ARGS] [USER_FLAGS] [KUBERC_FLAGS] # rest of the flags that are not passed by user in [USER_FLAGS] [USER_EXPLICIT_ARGS] [KUBERC_APPEND_ARGS] e.g.
	• name: runx command: run flags:
	  ◦ name: image default: nginx appendArgs:
	  --- 	
	  ◦ custom-arg1 For example, if user invokes "kubectl runx test-pod" command, this will be expanded to "kubectl run --image=nginx test-pod -- custom-arg1"
	• name: getn command: get flags:
	  ◦ name: output default: wide prependArgs:
	  ◦ node "kubectl getn control-plane-1" expands to "kubectl get node control-plane-1 --output=wide" "kubectl getn control-plane-1 --output=json" expands to "kubectl get node --output=json control-plane-1"

`AliasOverride`

Appears in:

- Preference

AliasOverride stores the alias definitions.

Field	Description
`name` **[Required]** `string`	name is the name of alias that can only include alphabetical characters If the alias name conflicts with the built-in command, built-in command will be used.
`command` **[Required]** `string`	command is the single or set of commands to execute, such as "set env" or "create"
`prependArgs` **[Required]** `[]string`	prependArgs stores the arguments such as resource names, etc. These arguments are inserted after the alias name.
`appendArgs` **[Required]** `[]string`	appendArgs stores the arguments such as resource names, etc. These arguments are appended to the USER_ARGS.
`flags` **[Required]** `[]CommandOptionDefault`	flags is allocated to store the flag definitions of alias. flags only modifies the default value of the flag and if user explicitly passes a value, explicit one is used.

`CommandDefaults`

Appears in:

- Preference

CommandDefaults stores the commands and their associated option's default values.

Field	Description
`command` **[Required]** `string`	command refers to a command whose flag's default value is changed.
`flags` **[Required]** `[]CommandOptionDefault`	flags is a list of flags storing different default values.

`CommandOptionDefault`

Appears in:

- AliasOverride
- CommandDefaults

CommandOptionDefault stores the name and the specified default value of an option.

Field	Description
`name` **[Required]** `string`	Flag name (long form, without dashes).
`default` **[Required]** `string`	In a string format of a default value. It will be parsed by kubectl to the compatible value of the flag.

WebhookAdmission Configuration (v1)

Package v1 is the v1 version of the API.

Resource Types

- WebhookAdmission

`WebhookAdmission`

WebhookAdmission provides configuration for the webhook admission controller.

Field	Description
`apiVersion` string	`apiserver.config.k8s.io/v1`
`kind` string	`WebhookAdmission`

Field	Description
`kubeConfigFile` **[Required]** `string`	KubeConfigFile is the path to the kubeconfig file.

---

kube-apiserver Configuration (v1)

Package v1 is the v1 version of the API.

Resource Types

- AdmissionConfiguration
- AuthenticationConfiguration
- AuthorizationConfiguration
- EncryptionConfiguration
- TracingConfiguration

`TracingConfiguration`

Appears in:

- KubeletConfiguration
- TracingConfiguration
- TracingConfiguration
- TracingConfiguration

TracingConfiguration provides versioned configuration for OpenTelemetry tracing clients.

Field	Description
`endpoint` `string`	Endpoint of the collector this component will report traces to. The connection is insecure, and does not currently support TLS. Recommended is unset, and endpoint is the otlp grpc default, localhost:4317.
`samplingRatePerMillion` `int32`	SamplingRatePerMillion is the number of samples to collect per million spans. Recommended is unset. If unset, sampler respects its parent span's sampling rate, but otherwise never samples.

`AdmissionConfiguration`

AdmissionConfiguration provides versioned configuration for admission controllers.

Field	Description
`apiVersion` string	`apiserver.config.k8s.io/v1`
`kind` string	`AdmissionConfiguration`
`plugins` `[]AdmissionPluginConfiguration`	Plugins allows specifying a configuration per admission control plugin.

`AuthenticationConfiguration`

AuthenticationConfiguration provides versioned configuration for authentication.

Field	Description
`apiVersion` string	`apiserver.config.k8s.io/v1`
`kind` string	`AuthenticationConfiguration`
`jwt` **[Required]** `[]JWTAuthenticator`	jwt is a list of authenticator to authenticate Kubernetes users using JWT compliant tokens. The authenticator will attempt to parse a raw ID token, verify it's been signed by the configured issuer. The public key to verify the signature is discovered from the issuer's public endpoint using OIDC discovery. For an incoming token, each JWT authenticator will be attempted in the order in which it is specified in this list. Note however that other authenticators may run before or after the JWT authenticators. The specific position of JWT authenticators in relation to other authenticators is neither defined nor stable across releases. Since each JWT authenticator must have a unique issuer URL, at most one JWT authenticator will attempt to cryptographically validate the token.

Field	Description
	The minimum valid JWT payload must contain the following claims: { "iss": "https://issuer.example.com", "aud": ["audience"], "exp": 1234567890, "": "username" }
`anonymous` **[Required]** AnonymousAuthConfig	If present --anonymous-auth must not be set

AuthorizationConfiguration

Field	Description
`apiVersion` string	`apiserver.config.k8s.io/v1`
`kind` string	`AuthorizationConfiguration`
`authorizers` **[Required]** []AuthorizerConfiguration	Authorizers is an ordered list of authorizers to authorize requests against. This is similar to the --authorization-modes kube-apiserver flag Must be at least one.

EncryptionConfiguration

EncryptionConfiguration stores the complete configuration for encryption providers. It also allows the use of wildcards to specify the resources that should be encrypted. Use '.' *to encrypt all resources within a group or* '.' *to encrypt all resources*. '.' can be used to encrypt all resource in the core group. '.' will encrypt all resources, even custom resources that are added after API server start. Use of wildcards that overlap within the same resource list or across multiple entries are not allowed since part of the configuration would be ineffective. Resource lists are processed in order, with earlier lists taking precedence.

Example:

```
kind: EncryptionConfiguration
apiVersion: apiserver.config.k8s.io/v1
resources:
- resources:
  - events
  providers:
  - identity: {}  # do not encrypt events even though *.* is specified below
- resources:
  - secrets
  - configmaps
  - pandas.awesome.bears.example
  providers:
  - aescbc:
      keys:
      - name: key1
        secret: c2VjcmV0IGlzIHNlY3VyZQ==
- resources:
  - '*.apps'
  providers:
  - aescbc:
      keys:
      - name: key2
        secret: c2VjcmV0IGlzIHNlY3VyZSwgb3IgaXMgaXQ/Cg==
- resources:
  - '*.*'
  providers:
  - aescbc:
      keys:
      - name: key3
        secret: c2VjcmV0IGlzIHNlY3VyZSwgSSB0aGluaw==
```

Field	Description
`apiVersion` string	`apiserver.config.k8s.io/v1`
`kind` string	`EncryptionConfiguration`
`resources` **[Required]** []ResourceConfiguration	resources is a list containing resources, and their corresponding encryption providers.

TracingConfiguration

TracingConfiguration provides versioned configuration for tracing clients.

Field	Description
`apiVersion` string	`apiserver.config.k8s.io/v1`
`kind` string	`TracingConfiguration`

Field	Description
`TracingConfiguration` **[Required]** `TracingConfiguration`	(Members of `TracingConfiguration` are embedded into this type.) Embed the component config tracing configuration struct

`AESConfiguration`

Appears in:

- ProviderConfiguration

AESConfiguration contains the API configuration for an AES transformer.

Field	Description
`keys` **[Required]** `[]Key`	keys is a list of keys to be used for creating the AES transformer. Each key has to be 32 bytes long for AES-CBC and 16, 24 or 32 bytes for AES-GCM.

`AdmissionPluginConfiguration`

Appears in:

- AdmissionConfiguration

AdmissionPluginConfiguration provides the configuration for a single plug-in.

Field	Description
`name` **[Required]** `string`	Name is the name of the admission controller. It must match the registered admission plugin name.
`path` `string`	Path is the path to a configuration file that contains the plugin's configuration
`configuration` `k8s.io/apimachinery/pkg/runtime.Unknown`	Configuration is an embedded configuration object to be used as the plugin's configuration. If present, it will be used instead of the path to the configuration file.

`AnonymousAuthCondition`

Appears in:

- AnonymousAuthConfig

AnonymousAuthCondition describes the condition under which anonymous auth should be enabled.

Field	Description
`path` **[Required]** `string`	Path for which anonymous auth is enabled.

`AnonymousAuthConfig`

Appears in:

- AuthenticationConfiguration

AnonymousAuthConfig provides the configuration for the anonymous authenticator.

Field	Description
`enabled` **[Required]** `bool`	No description provided.
`conditions` **[Required]** `[]AnonymousAuthCondition`	If set, anonymous auth is only allowed if the request meets one of the conditions.

`AudienceMatchPolicyType`

(Alias of `string`)

Appears in:

- Issuer

AudienceMatchPolicyType is a set of valid values for issuer.audienceMatchPolicy

`AuthorizerConfiguration`

Appears in:

- AuthorizationConfiguration

Field	Description
`type` **[Required]** `string`	Type refers to the type of the authorizer "Webhook" is supported in the generic API server Other API servers may support additional authorizer types like Node, RBAC, ABAC, etc.
`name` **[Required]** `string`	Name used to describe the webhook This is explicitly used in monitoring machinery for metrics Note: Names must be DNS1123 labels like `myauthorizername` or subdomains like `myauthorizer.example.domain` Required, with no default
`webhook` **[Required]** `WebhookConfiguration`	Webhook defines the configuration for a Webhook authorizer Must be defined when Type=Webhook Must not be defined when Type!=Webhook

`ClaimMappings`

Appears in:

- JWTAuthenticator

ClaimMappings provides the configuration for claim mapping

Field	Description
`username` **[Required]** `PrefixedClaimOrExpression`	username represents an option for the username attribute. The claim's value must be a singular string. Same as the --oidc-username-claim and --oidc-username-prefix flags. If username.expression is set, the expression must produce a string value. If username.expression uses 'claims.email', then 'claims.email_verified' must be used in username.expression or extra[*].valueExpression or claimValidationRules[*].expression. An example claim validation rule expression that matches the validation automatically applied when username.claim is set to 'email' is 'claims.?email_verified.orValue(true) == true'. By explicitly comparing the value to true, we let type-checking see the result will be a boolean, and to make sure a non-boolean email_verified claim will be caught at runtime. In the flag based approach, the --oidc-username-claim and --oidc-username-prefix are optional. If --oidc-username-claim is not set, the default value is "sub". For the authentication config, there is no defaulting for claim or prefix. The claim and prefix must be set explicitly. For claim, if --oidc-username-claim was not set with legacy flag approach, configure username.claim="sub" in the authentication config. For prefix: (1) --oidc-username-prefix="-", no prefix was added to the username. For the same behavior using authentication config, set username.prefix="" (2) --oidc-username-prefix="" and --oidc-username-claim != "email", prefix was "<value of --oidc-issuer-url>#". For the same behavior using authentication config, set username.prefix="#" (3) --oidc-username-prefix="". For the same behavior using authentication config, set username.prefix=""
`groups` `PrefixedClaimOrExpression`	groups represents an option for the groups attribute. The claim's value must be a string or string array claim. If groups.claim is set, the prefix must be specified (and can be the empty string). If groups.expression is set, the expression must produce a string or string array value. "", [], and null values are treated as the group mapping not being present.
`uid` `ClaimOrExpression`	uid represents an option for the uid attribute. Claim must be a singular string claim. If uid.expression is set, the expression must produce a string value.
`extra` `[]ExtraMapping`	extra represents an option for the extra attribute. expression must produce a string or string array value. If the value is empty, the extra mapping will not be present. hard-coded extra key/value - key: "foo" valueExpression: "'bar'" This will result in an extra attribute - foo: ["bar"] hard-coded key, value copying claim value - key: "foo" valueExpression: "claims.some_claim" This will result in an extra attribute - foo: [value of some_claim] hard-coded key, value derived from claim value - key: "admin" valueExpression: '(has(claims.is_admin) && claims.is_admin) ? "true":""' This will result in: - if is_admin claim is present and true, extra attribute - admin: ["true"] - if is_admin claim is present and false or is_admin claim is not present, no extra attribute will be added

`ClaimOrExpression`

Appears in:

- ClaimMappings

ClaimOrExpression provides the configuration for a single claim or expression.

Field	Description
`claim` `string`	claim is the JWT claim to use. Either claim or expression must be set. Mutually exclusive with expression.
`expression` `string`	expression represents the expression which will be evaluated by CEL. CEL expressions have access to the contents of the token claims, organized into CEL variable: - 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim.

`ClaimValidationRule`

Appears in:

- JWTAuthenticator

ClaimValidationRule provides the configuration for a single claim validation rule.

Field	Description
`claim` `string`	claim is the name of a required claim. Same as --oidc-required-claim flag. Only string claim keys are supported. Mutually exclusive with expression and message.
`requiredValue` `string`	requiredValue is the value of a required claim. Same as --oidc-required-claim flag. Only string claim values are supported. If claim is set and requiredValue is not set, the claim must be present with a value set to the empty string. Mutually exclusive with expression and message.
`expression` `string`	expression represents the expression which will be evaluated by CEL. Must produce a boolean. CEL expressions have access to the contents of the token claims, organized into CEL variable: - 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Must return true for the validation to pass. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim and requiredValue.
`message` `string`	message customizes the returned error message when expression returns false. message is a literal string. Mutually exclusive with claim and requiredValue.

`EgressSelectorType`

(Alias of `string`)

Appears in:

- Issuer

EgressSelectorType is an indicator of which egress selection should be used for sending traffic.

`ExtraMapping`

Appears in:

- ClaimMappings

ExtraMapping provides the configuration for a single extra mapping.

Field	Description
`key` **[Required]** `string`	key is a string to use as the extra attribute key. key must be a domain-prefix path (e.g. example.org/foo). All characters before the first "/" must be a valid subdomain as defined by RFC 1123. All characters trailing the first "/" must be valid HTTP Path characters as defined by RFC 3986. key must be lowercase. Required to be unique.
`valueExpression` **[Required]** `string`	valueExpression is a CEL expression to extract extra attribute value. valueExpression must produce a string or string array value. "", [], and null values are treated as the extra mapping not being present. Empty string values contained within a string array are filtered out. CEL expressions have access to the contents of the token claims, organized into CEL variable: - 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/

`IdentityConfiguration`

Appears in:

- ProviderConfiguration

IdentityConfiguration is an empty struct to allow identity transformer in provider configuration.

`Issuer`

Appears in:

- JWTAuthenticator

Issuer provides the configuration for an external provider's specific settings.

Field	Description
`url` **[Required]** `string`	url points to the issuer URL in a format https://url or https://url/path. This must match the "iss" claim in the presented JWT, and the issuer returned from discovery. Same value as the --oidc-issuer-url flag. Discovery information is fetched from "{url}/.well-known/openid-configuration" unless overridden by discoveryURL. Required to be unique across all JWT authenticators. Note that egress selection configuration is not used for this network connection.
`discoveryURL` `string`	discoveryURL, if specified, overrides the URL used to fetch discovery information instead of using "{url}/.well-known/openid-configuration". The exact value specified is used, so "/.well-known/openid-configuration" must be included in discoveryURL if needed. The "issuer" field in the fetched discovery information must match the "issuer.url" field in the AuthenticationConfiguration and will be used to validate the "iss" claim in the presented JWT. This is for scenarios where the well-known and jwks endpoints are hosted at a different location than the issuer (such as locally in the cluster). Example: A discovery url that is exposed using kubernetes service 'oidc' in namespace 'oidc-namespace' and discovery information is available at '/.well-known/openid-configuration'. discoveryURL: "https://oidc.oidc-namespace/.well-known/openid-configuration" certificateAuthority is used to verify the TLS connection and the hostname on the leaf certificate must be set to 'oidc.oidc-namespace'. curl https://oidc.oidc-namespace/.well-known/openid-configuration (.discoveryURL field) { issuer: "https://oidc.example.com" (.url field) } discoveryURL must be different from url. Required to be unique across all JWT authenticators. Note that egress selection configuration is not used for this network connection.
`certificateAuthority` `string`	certificateAuthority contains PEM-encoded certificate authority certificates used to validate the connection when fetching discovery information. If unset, the system verifier is used. Same value as the content of the file referenced by the --oidc-ca-file flag.
`audiences` **[Required]** `[]string`	audiences is the set of acceptable audiences the JWT must be issued to. At least one of the entries must match the "aud" claim in presented JWTs. Same value as the --oidc-client-id flag (though this field supports an array). Required to be non-empty.
`audienceMatchPolicy` `AudienceMatchPolicyType`	audienceMatchPolicy defines how the "audiences" field is used to match the "aud" claim in the presented JWT. Allowed values are: 1. "MatchAny" when multiple audiences are specified and 2. empty (or unset) or "MatchAny" when a single audience is specified.

Field	Description
	• MatchAny: the "aud" claim in the presented JWT must match at least one of the entries in the "audiences" field. For example, if "audiences" is ["foo", "bar"], the "aud" claim in the presented JWT must contain either "foo" or "bar" (and may contain both). • "": The match policy can be empty (or unset) when a single audience is specified in the "audiences" field. The "aud" claim in the presented JWT must contain the single audience (and may contain others). For more nuanced audience validation, use claimValidationRules. example: claimValidationRule[].expression: 'sets.equivalent(claims.aud, ["bar", "foo", "baz"])' to require an exact match.
`egressSelectorType` EgressSelectorType	egressSelectorType is an indicator of which egress selection should be used for sending all traffic related to this issuer (discovery, JWKS, distributed claims, etc). If unspecified, no custom dialer is used. When specified, the valid choices are "controlplane" and "cluster". These correspond to the associated values in the --egress-selector-config-file. • controlplane: for traffic intended to go to the control plane. • cluster: for traffic intended to go to the system being managed by Kubernetes.

JWTAuthenticator

Appears in:

- AuthenticationConfiguration

JWTAuthenticator provides the configuration for a single JWT authenticator.

Field	Description
`issuer` **[Required]** Issuer	issuer contains the basic OIDC provider connection options.
`claimValidationRules` []ClaimValidationRule	claimValidationRules are rules that are applied to validate token claims to authenticate users.
`claimMappings` **[Required]** ClaimMappings	claimMappings points claims of a token to be treated as user attributes.
`userValidationRules` []UserValidationRule	userValidationRules are rules that are applied to final user before completing authentication. These allow invariants to be applied to incoming identities such as preventing the use of the system: prefix that is commonly used by Kubernetes components. The validation rules are logically ANDed together and must all return true for the validation to pass.

KMSConfiguration

Appears in:

- ProviderConfiguration

KMSConfiguration contains the name, cache size and path to configuration file for a KMS based envelope transformer.

Field	Description
`apiVersion` `string`	apiVersion of KeyManagementService
`name` **[Required]** `string`	name is the name of the KMS plugin to be used.
`cachesize` `int32`	cachesize is the maximum number of secrets which are cached in memory. The default value is 1000. Set to a negative value to disable caching. This field is only allowed for KMS v1 providers.
`endpoint` **[Required]** `string`	endpoint is the gRPC server listening address, for example "unix:///var/run/kms-provider.sock".
`timeout` meta/v1.Duration	timeout for gRPC calls to kms-plugin (ex. 5s). The default is 3 seconds.

Key

Appears in:

- AESConfiguration
- SecretboxConfiguration

Key contains name and secret of the provided key for a transformer.

Field	Description
`name` **[Required]** `string`	name is the name of the key to be used while storing data to disk.
`secret` **[Required]** `string`	secret is the actual key, encoded in base64.

`PrefixedClaimOrExpression`

Appears in:

- ClaimMappings

PrefixedClaimOrExpression provides the configuration for a single prefixed claim or expression.

Field	Description
`claim` `string`	claim is the JWT claim to use. Mutually exclusive with expression.
`prefix` `string`	prefix is prepended to claim's value to prevent clashes with existing names. prefix needs to be set if claim is set and can be the empty string. Mutually exclusive with expression.
`expression` `string`	expression represents the expression which will be evaluated by CEL. CEL expressions have access to the contents of the token claims, organized into CEL variable: - 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim and prefix.

`ProviderConfiguration`

Appears in:

- ResourceConfiguration

ProviderConfiguration stores the provided configuration for an encryption provider.

Field	Description
`aesgcm` **[Required]** `AESConfiguration`	aesgcm is the configuration for the AES-GCM transformer.
`aescbc` **[Required]** `AESConfiguration`	aescbc is the configuration for the AES-CBC transformer.
`secretbox` **[Required]** `SecretboxConfiguration`	secretbox is the configuration for the Secretbox based transformer.
`identity` **[Required]** `IdentityConfiguration`	identity is the (empty) configuration for the identity transformer.
`kms` **[Required]** `KMSConfiguration`	kms contains the name, cache size and path to configuration file for a KMS based envelope transformer.

`ResourceConfiguration`

Appears in:

- EncryptionConfiguration

ResourceConfiguration stores per resource configuration.

Field	Description
`resources` **[Required]** `[]string`	resources is a list of kubernetes resources which have to be encrypted. The resource names are derived from `resource` or `resource.group` of the group/version/resource. eg: pandas.awesome.bears.example is a custom resource with 'group': awesome.bears.example, 'resource': pandas. Use '.' to encrypt all resources and '.' *to encrypt all resources in a specific group. eg:* '.awesome.bears.example' will encrypt all resources in the group 'awesome.bears.example'. eg: '*.' will encrypt all resources in the core group (such as pods, configmaps, etc).

Field	Description
`providers` **[Required]** []ProviderConfiguration	providers is a list of transformers to be used for reading and writing the resources to disk. eg: aesgcm, aescbc, secretbox, identity, kms.

SecretboxConfiguration

Appears in:

- ProviderConfiguration

SecretboxConfiguration contains the API configuration for an Secretbox transformer.

Field	Description
`keys` **[Required]** []Key	keys is a list of keys to be used for creating the Secretbox transformer. Each key has to be 32 bytes long.

UserValidationRule

Appears in:

- JWTAuthenticator

UserValidationRule provides the configuration for a single user info validation rule.

Field	Description
`expression` **[Required]** `string`	expression represents the expression which will be evaluated by CEL. Must return true for the validation to pass. CEL expressions have access to the contents of UserInfo, organized into CEL variable: - 'user' - authentication.k8s.io/v1, Kind=UserInfo object Refer to https://github.com/kubernetes/api/blob/release-1.28/authentication/v1/types.go#L105-L122 for the definition. API documentation: https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.28/#userinfo-v1-authentication-k8s-io Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/
`message` `string`	message customizes the returned error message when rule returns false. message is a literal string.

WebhookConfiguration

Appears in:

- AuthorizerConfiguration

Field	Description
`authorizedTTL` **[Required]** meta/v1.Duration	The duration to cache 'authorized' responses from the webhook authorizer. Same as setting `--authorization-webhook-cache-authorized-ttl` flag Default: 5m0s
`cacheAuthorizedRequests` `bool`	CacheAuthorizedRequests specifies whether authorized requests should be cached. If set to true, the TTL for cached decisions can be configured via the AuthorizedTTL field. Default: true
`unauthorizedTTL` **[Required]** meta/v1.Duration	The duration to cache 'unauthorized' responses from the webhook authorizer. Same as setting `--authorization-webhook-cache-unauthorized-ttl` flag Default: 30s
`cacheUnauthorizedRequests` `bool`	CacheUnauthorizedRequests specifies whether unauthorized requests should be cached. If set to true, the TTL for cached decisions can be configured via the UnauthorizedTTL field. Default: true
`timeout` **[Required]** meta/v1.Duration	Timeout for the webhook request Maximum allowed value is 30s. Required, no default value.
`subjectAccessReviewVersion` **[Required]** `string`	The API version of the authorization.k8s.io SubjectAccessReview to send to and expect from the webhook. Same as setting `--authorization-webhook-version` flag Valid values: v1beta1, v1 Required, no default value
`matchConditionSubjectAccessReviewVersion` **[Required]** `string`	MatchConditionSubjectAccessReviewVersion specifies the SubjectAccessReview version the CEL expressions are evaluated against Valid values: v1 Required, no default value
`failurePolicy` **[Required]** `string`	Controls the authorization decision when a webhook request fails to complete or returns a malformed response or errors evaluating matchConditions. Valid values:

Field	Description
	• NoOpinion: continue to subsequent authorizers to see if one of them allows the request • Deny: reject the request without consulting subsequent authorizers Required, with no default.
`connectionInfo` **[Required]** WebhookConnectionInfo	ConnectionInfo defines how we talk to the webhook

`matchConditions` **[Required]**
[]WebhookMatchCondition
	matchConditions is a list of conditions that must be met for a request to be sent to this webhook. An empty list of matchConditions matches all requests. There are a maximum of 64 match conditions allowed.
	
	The exact matching logic is (in order):
	
	1. If at least one matchCondition evaluates to FALSE, then the webhook is skipped.
	2. If ALL matchConditions evaluate to TRUE, then the webhook is called.
	3. If at least one matchCondition evaluates to an error (but none are FALSE):
	  ◦ If failurePolicy=Deny, then the webhook rejects the request
	  ◦ If failurePolicy=NoOpinion, then the error is ignored and the webhook is skipped

WebhookConnectionInfo

Appears in:

- WebhookConfiguration

Field	Description
	Controls how the webhook should communicate with the server. Valid values:
`type` **[Required]** `string`	• KubeConfigFile: use the file specified in kubeConfigFile to locate the server. • InClusterConfig: use the in-cluster configuration to call the SubjectAccessReview API hosted by kube-apiserver. This mode is not allowed for kube-apiserver.
`kubeConfigFile` **[Required]** `string`	Path to KubeConfigFile for connection info Required, if connectionInfo.Type is KubeConfig

WebhookMatchCondition

Appears in:

- WebhookConfiguration

Field	Description
	expression represents the expression which will be evaluated by CEL. Must evaluate to bool. CEL expressions have access to the contents of the SubjectAccessReview in v1 version. If version specified by subjectAccessReviewVersion in the request variable is v1beta1, the contents would be converted to the v1 version before evaluating the CEL expression.
`expression` **[Required]** `string`	• 'resourceAttributes' describes information for a resource access request and is unset for non-resource requests. e.g. has(request.resourceAttributes) && request.resourceAttributes.namespace == 'default' • 'nonResourceAttributes' describes information for a non-resource access request and is unset for resource requests. e.g. has(request.nonResourceAttributes) && request.nonResourceAttributes.path == '/healthz'. • 'user' is the user to test for. e.g. request.user == 'alice' • 'groups' is the groups to test for. e.g. ('group1' in request.groups) • 'extra' corresponds to the user.Info.GetExtra() method from the authenticator. • 'uid' is the information about the requesting user. e.g. request.uid == '1' Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/

---

kuberc (v1beta1)

Resource Types

- Preference

Preference

Preference stores elements of KubeRC configuration file

Field	Description
`apiVersion` string	`kubectl.config.k8s.io/v1beta1`
`kind` string	`Preference`
`defaults` **[Required]** []CommandDefaults	defaults allow changing default option values of commands. This is especially useful, when user doesn't want to explicitly set options each time.

`aliases` **[Required]**
[]AliasOverride
	aliases allow defining command aliases for existing kubectl commands, with optional default option values. If the alias name collides with a built-in command, built-in command always takes precedence. Option overrides defined in the defaults section do NOT apply to aliases for the same command. kubectl [ALIAS NAME] [USER_OPTIONS] [USER_EXPLICIT_ARGS] expands to kubectl [COMMAND] # built-in command alias points to [KUBERC_PREPEND_ARGS] [USER_OPTIONS] [KUBERC_OPTIONS] # rest of the options that are not passed by user in [USER_OPTIONS] [USER_EXPLICIT_ARGS] [KUBERC_APPEND_ARGS] e.g.
	• name: runx command: run options:
	  ◦ name: image default: nginx appendArgs:
	  --- 	
	  ◦ custom-arg1 For example, if user invokes "kubectl runx test-pod" command, this will be expanded to "kubectl run --image=nginx test-pod -- custom-arg1"
	• name: getn command: get options:
	  ◦ name: output default: wide prependArgs:
	  ◦ node "kubectl getn control-plane-1" expands to "kubectl get node control-plane-1 --output=wide" "kubectl getn control-plane-1 --output=json" expands to "kubectl get node --output=json control-plane-1"

AliasOverride

Appears in:

- Preference

AliasOverride stores the alias definitions.

Field	Description
`name` **[Required]** `string`	name is the name of alias that can only include alphabetical characters If the alias name conflicts with the built-in command, built-in command will be used.
`command` **[Required]** `string`	command is the single or set of commands to execute, such as "set env" or "create"
`prependArgs` **[Required]** `[]string`	prependArgs stores the arguments such as resource names, etc. These arguments are inserted after the alias name.
`appendArgs` **[Required]** `[]string`	appendArgs stores the arguments such as resource names, etc. These arguments are appended to the USER_ARGS.
`options` **[Required]** []CommandOptionDefault	options is allocated to store the option definitions of alias. options only modify the default value of the option and if user explicitly passes a value, explicit one is used.

CommandDefaults

Appears in:

- Preference

CommandDefaults stores the commands and their associated option's default values.

Field	Description
`command` **[Required]** `string`	command refers to a command whose option's default value is changed.
`options` **[Required]** []CommandOptionDefault	options is a list of options storing different default values.

CommandOptionDefault

Appears in:

- AliasOverride
- CommandDefaults

CommandOptionDefault stores the name and the specified default value of an option.

Field	Description
`name` **[Required]** `string`	Option name (long form, without dashes).
`default` **[Required]** `string`	In a string format of a default value. It will be parsed by kubectl to the compatible value of the option.

Client Authentication (v1beta1)

Resource Types

- ExecCredential

`ExecCredential`

ExecCredential is used by exec-based plugins to communicate credentials to HTTP transports.

Field	Description
`apiVersion` string	`client.authentication.k8s.io/v1beta1`
`kind` string	`ExecCredential`
`spec` **[Required]** `ExecCredentialSpec`	Spec holds information passed to the plugin by the transport.
`status` `ExecCredentialStatus`	Status is filled in by the plugin and holds the credentials that the transport should use to contact the API.

`Cluster`

Appears in:

- ExecCredentialSpec

Cluster contains information to allow an exec plugin to communicate with the kubernetes cluster being authenticated to.

To ensure that this struct contains everything someone would need to communicate with a kubernetes cluster (just like they would via a kubeconfig), the fields should shadow "k8s.io/client-go/tools/clientcmd/api/v1".Cluster, with the exception of CertificateAuthority, since CA data will always be passed to the plugin as bytes.

Field	Description
`server` **[Required]** `string`	Server is the address of the kubernetes cluster (https://hostname:port).
`tls-server-name` `string`	TLSServerName is passed to the server for SNI and is used in the client to check server certificates against. If ServerName is empty, the hostname used to contact the server is used.
`insecure-skip-tls-verify` `bool`	InsecureSkipTLSVerify skips the validity check for the server's certificate. This will make your HTTPS connections insecure.
`certificate-authority-data` `[]byte`	CAData contains PEM-encoded certificate authority certificates. If empty, system roots should be used.
`proxy-url` `string`	ProxyURL is the URL to the proxy to be used for all requests to this cluster.
`disable-compression` `bool`	DisableCompression allows client to opt-out of response compression for all requests to the server. This is useful to speed up requests (specifically lists) when client-server network bandwidth is ample, by saving time on compression (server-side) and decompression (client-side): https://github.com/kubernetes/kubernetes/issues/112296.
`config` `k8s.io/apimachinery/pkg/runtime.RawExtension`	Config holds additional config data that is specific to the exec plugin with regards to the cluster being authenticated to. This data is sourced from the clientcmd Cluster object's extensions[client.authentication.k8s.io/exec] field:

Field	Description
	clusters: • name: my-cluster cluster: ... extensions: ◦ name: client.authentication.k8s.io/exec # reserved extension name for per cluster exec config extension: audience: 06e3fbd18de8 # arbitrary config In some environments, the user config may be exactly the same across many clusters (i.e. call this exec plugin) minus some details that are specific to each cluster such as the audience. This field allows the per cluster config to be directly specified with the cluster info. Using this field to store secret data is not recommended as one of the prime benefits of exec plugins is that no secrets need to be stored directly in the kubeconfig.

`ExecCredentialSpec`

Appears in:

- ExecCredential

ExecCredentialSpec holds request and runtime specific information provided by the transport.

Field	Description
`cluster` `Cluster`	Cluster contains information to allow an exec plugin to communicate with the kubernetes cluster being authenticated to. Note that Cluster is non-nil only when provideClusterInfo is set to true in the exec provider config (i.e., ExecConfig.ProvideClusterInfo).
`interactive` **[Required]** `bool`	Interactive declares whether stdin has been passed to this exec plugin.

`ExecCredentialStatus`

Appears in:

- ExecCredential

ExecCredentialStatus holds credentials for the transport to use.

Token and ClientKeyData are sensitive fields. This data should only be transmitted in-memory between client and exec plugin process. Exec plugin itself should at least be protected via file permissions.

Field	Description
`expirationTimestamp` `meta/v1.Time`	ExpirationTimestamp indicates a time when the provided credentials expire.
`token` **[Required]** `string`	Token is a bearer token used by the client for request authentication.
`clientCertificateData` **[Required]** `string`	PEM-encoded client TLS certificates (including intermediates, if any).
`clientKeyData` **[Required]** `string`	PEM-encoded private key for the above certificate.

Client Authentication (v1)

Resource Types

- ExecCredential

`ExecCredential`

ExecCredential is used by exec-based plugins to communicate credentials to HTTP transports.

Field	Description
`apiVersion` string	`client.authentication.k8s.io/v1`
`kind` string	`ExecCredential`

Field	Description
`spec` **[Required]** ExecCredentialSpec	Spec holds information passed to the plugin by the transport.
`status` ExecCredentialStatus	Status is filled in by the plugin and holds the credentials that the transport should use to contact the API.

Cluster

Appears in:

- ExecCredentialSpec

Cluster contains information to allow an exec plugin to communicate with the kubernetes cluster being authenticated to.

To ensure that this struct contains everything someone would need to communicate with a kubernetes cluster (just like they would via a kubeconfig), the fields should shadow "k8s.io/client-go/tools/clientcmd/api/v1".Cluster, with the exception of CertificateAuthority, since CA data will always be passed to the plugin as bytes.

Field	Description
`server` **[Required]** `string`	Server is the address of the kubernetes cluster (https://hostname:port).
`tls-server-name` `string`	TLSServerName is passed to the server for SNI and is used in the client to check server certificates against. If ServerName is empty, the hostname used to contact the server is used.
`insecure-skip-tls-verify` `bool`	InsecureSkipTLSVerify skips the validity check for the server's certificate. This will make your HTTPS connections insecure.
`certificate-authority-data` `[]byte`	CAData contains PEM-encoded certificate authority certificates. If empty, system roots should be used.
`proxy-url` `string`	ProxyURL is the URL to the proxy to be used for all requests to this cluster.
`disable-compression` `bool`	DisableCompression allows client to opt-out of response compression for all requests to the server. This is useful to speed up requests (specifically lists) when client-server network bandwidth is ample, by saving time on compression (server-side) and decompression (client-side): https://github.com/kubernetes/kubernetes/issues/112296.
`config` k8s.io/apimachinery/pkg/runtime.RawExtension	Config holds additional config data that is specific to the exec plugin with regards to the cluster being authenticated to. This data is sourced from the clientcmd Cluster object's extensions[client.authentication.k8s.io/exec] field: clusters: - name: my-cluster cluster: ... extensions: - name: client.authentication.k8s.io/exec # reserved extension name for per cluster exec config extension: audience: 06e3fbd18de8 # arbitrary config In some environments, the user config may be exactly the same across many clusters (i.e. call this exec plugin) minus some details that are specific to each cluster such as the audience. This field allows the per cluster config to be directly specified with the cluster info. Using this field to store secret data is not recommended as one of the prime benefits of exec plugins is that no secrets need to be stored directly in the kubeconfig.

ExecCredentialSpec

Appears in:

- ExecCredential

ExecCredentialSpec holds request and runtime specific information provided by the transport.

Field	Description
`cluster` Cluster	Cluster contains information to allow an exec plugin to communicate with the kubernetes cluster being authenticated to. Note that Cluster is non-nil only when provideClusterInfo is set to true in the exec provider config (i.e., ExecConfig.ProvideClusterInfo).
`interactive` **[Required]** `bool`	Interactive declares whether stdin has been passed to this exec plugin.

`ExecCredentialStatus`

Appears in:

- ExecCredential

ExecCredentialStatus holds credentials for the transport to use.

Token and ClientKeyData are sensitive fields. This data should only be transmitted in-memory between client and exec plugin process. Exec plugin itself should at least be protected via file permissions.

Field	Description
`expirationTimestamp` `meta/v1.Time`	ExpirationTimestamp indicates a time when the provided credentials expire.
`token` **[Required]** `string`	Token is a bearer token used by the client for request authentication.
`clientCertificateData` **[Required]** `string`	PEM-encoded client TLS certificates (including intermediates, if any).
`clientKeyData` **[Required]** `string`	PEM-encoded private key for the above certificate.

Kubelet Configuration (v1alpha1)

Resource Types

- CredentialProviderConfig
- ImagePullIntent
- ImagePulledRecord

`CredentialProviderConfig`

CredentialProviderConfig is the configuration containing information about each exec credential provider. Kubelet reads this configuration from disk and enables each provider as specified by the CredentialProvider type.

Field	Description
`apiVersion` string	`kubelet.config.k8s.io/v1alpha1`
`kind` string	`CredentialProviderConfig`
`providers` **[Required]** `[]CredentialProvider`	providers is a list of credential provider plugins that will be enabled by the kubelet. Multiple providers may match against a single image, in which case credentials from all providers will be returned to the kubelet. If multiple providers are called for a single image, the results are combined. If providers return overlapping auth keys, the value from the provider earlier in this list is attempted first.

`ImagePullIntent`

ImagePullIntent is a record of the kubelet attempting to pull an image.

Field	Description
`apiVersion` string	`kubelet.config.k8s.io/v1alpha1`
`kind` string	`ImagePullIntent`
`image` **[Required]** `string`	Image is the image spec from a Container's `image` field. The filename is a SHA-256 hash of this value. This is to avoid filename-unsafe characters like ':' and '/'.

`ImagePulledRecord`

ImagePullRecord is a record of an image that was pulled by the kubelet.

If there are no records in the `kubernetesSecrets` field and both `nodeWideCredentials` and `anonymous` are `false`, credentials must be re-checked the next time an image represented by this record is being requested.

Field	Description
`apiVersion` string	`kubelet.config.k8s.io/v1alpha1`
`kind` string	`ImagePulledRecord`
`lastUpdatedTime` **[Required]** meta/v1.Time	LastUpdatedTime is the time of the last update to this record
`imageRef` **[Required]** `string`	ImageRef is a reference to the image represented by this file as received from the CRI. The filename is a SHA-256 hash of this value. This is to avoid filename-unsafe characters like ':' and '/'.
`credentialMapping` **[Required]** map[string]ImagePullCredentials	CredentialMapping maps `image` to the set of credentials that it was previously pulled with. `image` in this case is the content of a pod's container `image` field that's got its tag/digest removed. Example: Container requests the `hello-world:latest@sha256:91fb4b041da273d5a3273b6d587d62d518300a6ad268b28628f74997b93171b2` image: "credentialMapping": { "hello-world": { "nodePodsAccessible": true } }

`CredentialProvider`

Appears in:

- CredentialProviderConfig

CredentialProvider represents an exec plugin to be invoked by the kubelet. The plugin is only invoked when an image being pulled matches the images handled by the plugin (see matchImages).

Field	Description
`name` **[Required]** `string`	name is the required name of the credential provider. It must match the name of the provider executable as seen by the kubelet. The executable must be in the kubelet's bin directory (set by the --image-credential-provider-bin-dir flag). Required to be unique across all providers.
`matchImages` **[Required]** `[]string`	matchImages is a required list of strings used to match against images in order to determine if this provider should be invoked. If one of the strings matches the requested image from the kubelet, the plugin will be invoked and given a chance to provide credentials. Images are expected to contain the registry domain and URL path. Each entry in matchImages is a pattern which can optionally contain a port and a path. Globs can be used in the domain, but not in the port or the path. Globs are supported as subdomains like `*.k8s.io` or `k8s.*.io`, and top-level-domains such as `k8s.*`. Matching partial subdomains like `app*.k8s.io` is also supported. Each glob can only match a single subdomain segment, so `*.io` does not match `*.k8s.io`. A match exists between an image and a matchImage when all of the below are true: • Both contain the same number of domain parts and each part matches. • The URL path of an imageMatch must be a prefix of the target image URL path. • If the imageMatch contains a port, then the port must match in the image as well. Example values of matchImages: • `123456789.dkr.ecr.us-east-1.amazonaws.com` • `*.azurecr.io` • `gcr.io` • `*.*.registry.io` • `registry.io:8080/path`
`defaultCacheDuration` **[Required]** meta/v1.Duration	defaultCacheDuration is the default duration the plugin will cache credentials in-memory if a cache duration is not provided in the plugin response. This field is required.
`apiVersion` **[Required]** `string`	Required input version of the exec CredentialProviderRequest. The returned CredentialProviderResponse MUST use the same encoding version as the input. Current supported values are: • credentialprovider.kubelet.k8s.io/v1alpha1
`args` `[]string`	Arguments to pass to the command when executing it.
`env` []ExecEnvVar	Env defines additional environment variables to expose to the process. These are unioned with the host's environment, as well as variables client-go uses to pass argument to the plugin.

`ExecEnvVar`

Appears in:

- CredentialProvider

ExecEnvVar is used for setting environment variables when executing an exec-based credential plugin.

Field	Description
`name` **[Required]** `string`	No description provided.
`value` **[Required]** `string`	No description provided.

`ImagePullCredentials`

Appears in:

- ImagePulledRecord

ImagePullCredentials describe credentials that can be used to pull an image.

Field	Description
`kubernetesSecrets` `[]ImagePullSecret`	KuberneteSecretCoordinates is an index of coordinates of all the kubernetes secrets that were used to pull the image.
`kubernetesServiceAccounts` `[]ImagePullServiceAccount`	KubernetesServiceAccounts is an index of coordinates of all the kubernetes service accounts that were used to pull the image.
`nodePodsAccessible` `bool`	NodePodsAccessible is a flag denoting the pull credentials are accessible by all the pods on the node, or that no credentials are needed for the pull. If true, it is mutually exclusive with the `kubernetesSecrets` field.

`ImagePullSecret`

Appears in:

- ImagePullCredentials

ImagePullSecret is a representation of a Kubernetes secret object coordinates along with a credential hash of the pull secret credentials this object contains.

Field	Description
`uid` **[Required]** `string`	No description provided.
`namespace` **[Required]** `string`	No description provided.
`name` **[Required]** `string`	No description provided.
`credentialHash` **[Required]** `string`	CredentialHash is a SHA-256 retrieved by hashing the image pull credentials content of the secret specified by the UID/Namespace/Name coordinates.

`ImagePullServiceAccount`

Appears in:

- ImagePullCredentials

ImagePullServiceAccount is a representation of a Kubernetes service account object coordinates for which the kubelet sent service account token to the credential provider plugin for image pull credentials.

Field	Description
`uid` **[Required]** `string`	No description provided.
`namespace` **[Required]** `string`	No description provided.
`name` **[Required]** `string`	No description provided.

---

kubeadm Configuration (v1beta3)

Overview

Package v1beta3 defines the v1beta3 version of the kubeadm configuration file format. This version improves on the v1beta2 format by fixing some minor issues and adding a few new fields.

A list of changes since v1beta2:

- The deprecated "ClusterConfiguration.useHyperKubeImage" field has been removed. Kubeadm no longer supports the hyperkube image.
- The "ClusterConfiguration.dns.type" field has been removed since CoreDNS is the only supported DNS server type by kubeadm.
- Include "datapolicy" tags on the fields that hold secrets. This would result in the field values to be omitted when API structures are printed with klog.
- Add "InitConfiguration.skipPhases", "JoinConfiguration.skipPhases" to allow skipping a list of phases during kubeadm init/join command execution.
- Add "InitConfiguration.nodeRegistration.imagePullPolicy" and "JoinConfiguration.nodeRegistration.imagePullPolicy" to allow specifying the images pull policy during kubeadm "init" and "join". The value must be one of "Always", "Never" or "IfNotPresent". "IfNotPresent" is the default, which has been the existing behavior prior to this addition.
- Add "InitConfiguration.patches.directory", "JoinConfiguration.patches.directory" to allow the user to configure a directory from which to take patches for components deployed by kubeadm.
- Move the BootstrapToken* API and related utilities out of the "kubeadm" API group to a new group "bootstraptoken". The kubeadm API version v1beta3 no longer contains the BootstrapToken* structures.

Migration from old kubeadm config versions

- kubeadm v1.15.x and newer can be used to migrate from v1beta1 to v1beta2.
- kubeadm v1.22.x and newer no longer support v1beta1 and older APIs, but can be used to migrate v1beta2 to v1beta3.
- kubeadm v1.27.x and newer no longer support v1beta2 and older APIs,

Basics

The preferred way to configure kubeadm is to pass an YAML configuration file with the `--config` option. Some of the configuration options defined in the kubeadm config file are also available as command line flags, but only the most common/simple use case are supported with this approach.

A kubeadm config file could contain multiple configuration types separated using three dashes (`---`).

kubeadm supports the following configuration types:

```
apiVersion: kubeadm.k8s.io/v1beta3
kind: InitConfiguration

apiVersion: kubeadm.k8s.io/v1beta3
kind: ClusterConfiguration

apiVersion: kubelet.config.k8s.io/v1beta1
kind: KubeletConfiguration

apiVersion: kubeproxy.config.k8s.io/v1alpha1
kind: KubeProxyConfiguration

apiVersion: kubeadm.k8s.io/v1beta3
kind: JoinConfiguration
```

To print the defaults for "init" and "join" actions use the following commands:

```
kubeadm config print init-defaults
kubeadm config print join-defaults
```

The list of configuration types that must be included in a configuration file depends by the action you are performing (`init` or `join`) and by the configuration options you are going to use (defaults or advanced customization).

If some configuration types are not provided, or provided only partially, kubeadm will use default values; defaults provided by kubeadm includes also enforcing consistency of values across components when required (e.g. `--cluster-cidr` flag on controller manager and `clusterCIDR` on kube-proxy).

Users are always allowed to override default values, with the only exception of a small subset of setting with relevance for security (e.g. enforce authorization-mode Node and RBAC on api server).

If the user provides a configuration types that is not expected for the action you are performing, kubeadm will ignore those types and print a warning.

Kubeadm init configuration types

When executing kubeadm init with the `--config` option, the following configuration types could be used: InitConfiguration, ClusterConfiguration, KubeProxyConfiguration, KubeletConfiguration, but only one between InitConfiguration and ClusterConfiguration is mandatory.

```
apiVersion: kubeadm.k8s.io/v1beta3
kind: InitConfiguration
bootstrapTokens:
  ...
nodeRegistration:
  ...
```

The InitConfiguration type should be used to configure runtime settings, that in case of kubeadm init are the configuration of the bootstrap token and all the setting which are specific to the node where kubeadm is executed, including:

- NodeRegistration, that holds fields that relate to registering the new node to the cluster; use it to customize the node name, the CRI socket to use or any other settings that should apply to this node only (e.g. the node ip).
- LocalAPIEndpoint, that represents the endpoint of the instance of the API server to be deployed on this node; use it e.g. to customize the API server advertise address.

```
apiVersion: kubeadm.k8s.io/v1beta3
kind: ClusterConfiguration
networking:
  ...
etcd:
  ...
apiServer:
  extraArgs:
    ...
  extraVolumes:
    ...
...
```

The ClusterConfiguration type should be used to configure cluster-wide settings, including settings for:

- `networking` that holds configuration for the networking topology of the cluster; use it e.g. to customize Pod subnet or services subnet.
- `etcd`: use it e.g. to customize the local etcd or to configure the API server for using an external etcd cluster.
- kube-apiserver, kube-scheduler, kube-controller-manager configurations; use it to customize control-plane components by adding customized setting or overriding kubeadm default settings.

```
apiVersion: kubeproxy.config.k8s.io/v1alpha1
kind: KubeProxyConfiguration
  ...
```

The KubeProxyConfiguration type should be used to change the configuration passed to kube-proxy instances deployed in the cluster. If this object is not provided or provided only partially, kubeadm applies defaults.

See https://kubernetes.io/docs/reference/command-line-tools-reference/kube-proxy/ or https://pkg.go.dev/k8s.io/kube-proxy/config/v1alpha1#KubeProxyConfiguration for kube-proxy official documentation.

```
apiVersion: kubelet.config.k8s.io/v1beta1
kind: KubeletConfiguration
  ...
```

The KubeletConfiguration type should be used to change the configurations that will be passed to all kubelet instances deployed in the cluster. If this object is not provided or provided only partially, kubeadm applies defaults.

See https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/ or https://pkg.go.dev/k8s.io/kubelet/config/v1beta1#KubeletConfiguration for kubelet official documentation.

Here is a fully populated example of a single YAML file containing multiple configuration types to be used during a `kubeadm init` run.

```
apiVersion: kubeadm.k8s.io/v1beta3
kind: InitConfiguration
bootstrapTokens:
  - token: "9a08jv.c0izixklcxtmnze7"
    description: "kubeadm bootstrap token"
    ttl: "24h"
  - token: "783bde.3f89s0fje9f38fhf"
    description: "another bootstrap token"
    usages:
      - authentication
      - signing
    groups:
      - system:bootstrappers:kubeadm:default-node-token
nodeRegistration:
  name: "ec2-10-100-0-1"
  criSocket: "/var/run/dockershim.sock"
  taints:
    - key: "kubeadmNode"
      value: "someValue"
      effect: "NoSchedule"
  kubeletExtraArgs:
    v: 4
```

```yaml
  ignorePreflightErrors:
    - IsPrivilegedUser
  imagePullPolicy: "IfNotPresent"
localAPIEndpoint:
  advertiseAddress: "10.100.0.1"
  bindPort: 6443
certificateKey: "e6a2eb8581237ab72a4f494f30285ec12a9694d750b9785706a83bfcbbbd2204"
skipPhases:
  - addon/kube-proxy
---
apiVersion: kubeadm.k8s.io/v1beta3
kind: ClusterConfiguration
etcd:
  # one of local or external
  local:
    imageRepository: "registry.k8s.io"
    imageTag: "3.2.24"
    dataDir: "/var/lib/etcd"
    extraArgs:
      listen-client-urls: "http://10.100.0.1:2379"
    serverCertSANs:
      -  "ec2-10-100-0-1.compute-1.amazonaws.com"
    peerCertSANs:
      - "10.100.0.1"
  # external:
    # endpoints:
    # - "10.100.0.1:2379"
    # - "10.100.0.2:2379"
    # caFile: "/etcd/kubernetes/pki/etcd/etcd-ca.crt"
    # certFile: "/etcd/kubernetes/pki/etcd/etcd.crt"
    # keyFile: "/etcd/kubernetes/pki/etcd/etcd.key"
networking:
  serviceSubnet: "10.96.0.0/16"
  podSubnet: "10.244.0.0/24"
  dnsDomain: "cluster.local"
kubernetesVersion: "v1.21.0"
controlPlaneEndpoint: "10.100.0.1:6443"
apiServer:
  extraArgs:
    authorization-mode: "Node,RBAC"
  extraVolumes:
    - name: "some-volume"
      hostPath: "/etc/some-path"
      mountPath: "/etc/some-pod-path"
      readOnly: false
      pathType: File
  certSANs:
    - "10.100.1.1"
    - "ec2-10-100-0-1.compute-1.amazonaws.com"
  timeoutForControlPlane: 4m0s
controllerManager:
  extraArgs:
    "node-cidr-mask-size": "20"
  extraVolumes:
    - name: "some-volume"
      hostPath: "/etc/some-path"
      mountPath: "/etc/some-pod-path"
      readOnly: false
      pathType: File
scheduler:
  extraArgs:
    bind-address: "10.100.0.1"
  extraVolumes:
    - name: "some-volume"
      hostPath: "/etc/some-path"
      mountPath: "/etc/some-pod-path"
      readOnly: false
      pathType: File
certificatesDir: "/etc/kubernetes/pki"
imageRepository: "registry.k8s.io"
clusterName: "example-cluster"
---
apiVersion: kubelet.config.k8s.io/v1beta1
kind: KubeletConfiguration
# kubelet specific options here
---
apiVersion: kubeproxy.config.k8s.io/v1alpha1
kind: KubeProxyConfiguration
# kube-proxy specific options here
```

Kubeadm join configuration types

When executing `kubeadm join` with the `--config` option, the JoinConfiguration type should be provided.

```
apiVersion: kubeadm.k8s.io/v1beta3
kind: JoinConfiguration
  ...
```

The JoinConfiguration type should be used to configure runtime settings, that in case of `kubeadm join` are the discovery method used for accessing the cluster info and all the setting which are specific to the node where kubeadm is executed, including:

- `nodeRegistration`, that holds fields that relate to registering the new node to the cluster; use it to customize the node name, the CRI socket to use or any other settings that should apply to this node only (e.g. the node ip).
- `apiEndpoint`, that represents the endpoint of the instance of the API server to be eventually deployed on this node.

Resource Types

`BootstrapToken`

Appears in:

- InitConfiguration

BootstrapToken describes one bootstrap token, stored as a Secret in the cluster

Field	Description
`token` **[Required]** BootstrapTokenString	`token` is used for establishing bidirectional trust between nodes and control-planes. Used for joining nodes in the cluster.
`description` `string`	`description` sets a human-friendly message why this token exists and what it's used for, so other administrators can know its purpose.
`ttl` meta/v1.Duration	`ttl` defines the time to live for this token. Defaults to `24h`. `expires` and `ttl` are mutually exclusive.
`expires` meta/v1.Time	`expires` specifies the timestamp when this token expires. Defaults to being set dynamically at runtime based on the `ttl`. `expires` and `ttl` are mutually exclusive.
`usages` `[]string`	`usages` describes the ways in which this token can be used. Can by default be used for establishing bidirectional trust, but that can be changed here.
`groups` `[]string`	`groups` specifies the extra groups that this token will authenticate as when/if used for authentication

`BootstrapTokenString`

Appears in:

- BootstrapToken

BootstrapTokenString is a token of the format `abcdef.abcdef0123456789` that is used for both validation of the practically of the API server from a joining node's point of view and as an authentication method for the node in the bootstrap phase of "kubeadm join". This token is and should be short-lived.

Field	Description
- **[Required]** `string`	No description provided.
- **[Required]** `string`	No description provided.

`ClusterConfiguration`

ClusterConfiguration contains cluster-wide configuration for a kubeadm cluster.

Field	Description
`apiVersion` string	`kubeadm.k8s.io/v1beta3`
`kind` string	`ClusterConfiguration`
`etcd` Etcd	`etcd` holds the configuration for etcd.
`networking` Networking	`networking` holds configuration for the networking topology of the cluster.

Field	Description
kubernetesVersion string	kubernetesVersion is the target version of the control plane.
controlPlaneEndpoint string	controlPlaneEndpoint sets a stable IP address or DNS name for the control plane. It can be a valid IP address or a RFC-1123 DNS subdomain, both with optional TCP port. In case the controlPlaneEndpoint is not specified, the advertiseAddress + bindPort are used; in case the controlPlaneEndpoint is specified but without a TCP port, the bindPort is used. Possible usages are: • In a cluster with more than one control plane instances, this field should be assigned the address of the external load balancer in front of the control plane instances. • In environments with enforced node recycling, the controlPlaneEndpoint could be used for assigning a stable DNS to the control plane.
apiServer APIServer	apiServer contains extra settings for the API server.
controllerManager ControlPlaneComponent	controllerManager contains extra settings for the controller manager.
scheduler ControlPlaneComponent	scheduler contains extra settings for the scheduler.
dns DNS	dns defines the options for the DNS add-on installed in the cluster.
certificatesDir string	certificatesDir specifies where to store or look for all required certificates.
imageRepository string	imageRepository sets the container registry to pull images from. If empty, registry.k8s.io will be used by default. In case of kubernetes version is a CI build (kubernetes version starts with ci/) gcr.io/k8s-staging-ci-images will be used as a default for control plane components and for kube-proxy, while registry.k8s.io will be used for all the other images.
featureGates map[string]bool	featureGates contains the feature gates enabled by the user.
clusterName string	The cluster name.

InitConfiguration

InitConfiguration contains a list of elements that is specific "kubeadm init"-only runtime information. kubeadm init-only information. These fields are solely used the first time kubeadm init runs. After that, the information in the fields IS NOT uploaded to the kubeadm-config ConfigMap that is used by kubeadm upgrade for instance. These fields must be omitempty.

Field	Description
apiVersion string	kubeadm.k8s.io/v1beta3
kind string	InitConfiguration
bootstrapTokens []BootstrapToken	bootstrapTokens is respected at kubeadm init time and describes a set of Bootstrap Tokens to create. This information IS NOT uploaded to the kubeadm cluster configmap, partly because of its sensitive nature
nodeRegistration NodeRegistrationOptions	nodeRegistration holds fields that relate to registering the new control-plane node to the cluster.
localAPIEndpoint APIEndpoint	localAPIEndpoint represents the endpoint of the API server instance that's deployed on this control plane node. In HA setups, this differs from ClusterConfiguration.controlPlaneEndpoint in the sense that controlPlaneEndpoint is the global endpoint for the cluster, which then load-balances the requests to each individual API server. This configuration object lets you customize what IP/DNS name and port the local API server advertises it's accessible on. By default, kubeadm tries to auto-detect the IP of the default interface and use that, but in case that process fails you may set the desired value here.
certificateKey string	certificateKey sets the key with which certificates and keys are encrypted prior to being uploaded in a Secret in the cluster during the uploadcerts init phase. The certificate key is a hex encoded string that is an AES key of size 32 bytes.
skipPhases []string	skipPhases is a list of phases to skip during command execution. The list of phases can be obtained with the kubeadm init --help command. The flag "--skip-phases" takes precedence over this field.
patches Patches	patches contains options related to applying patches to components deployed by kubeadm during kubeadm init.

JoinConfiguration

JoinConfiguration contains elements describing a particular node.

Field	Description
`apiVersion` string	`kubeadm.k8s.io/v1beta3`
`kind` string	`JoinConfiguration`
`nodeRegistration` `NodeRegistrationOptions`	`nodeRegistration` holds fields that relate to registering the new control-plane node to the cluster.
`caCertPath` `string`	`caCertPath` is the path to the SSL certificate authority used to secure communications between a node and the control-plane. Defaults to "/etc/kubernetes/pki/ca.crt".
`discovery` **[Required]** `Discovery`	`discovery` specifies the options for the kubelet to use during the TLS bootstrap process.
`controlPlane` `JoinControlPlane`	`controlPlane` defines the additional control plane instance to be deployed on the joining node. If nil, no additional control plane instance will be deployed.
`skipPhases` `[]string`	`skipPhases` is a list of phases to skip during command execution. The list of phases can be obtained with the `kubeadm join --help` command. The flag `--skip-phases` takes precedence over this field.
`patches` `Patches`	`patches` contains options related to applying patches to components deployed by kubeadm during `kubeadm join`.

`APIEndpoint`

Appears in:

- InitConfiguration
- JoinControlPlane

APIEndpoint struct contains elements of API server instance deployed on a node.

Field	Description
`advertiseAddress` `string`	`advertiseAddress` sets the IP address for the API server to advertise.
`bindPort` `int32`	`bindPort` sets the secure port for the API Server to bind to. Defaults to 6443.

`APIServer`

Appears in:

- ClusterConfiguration

APIServer holds settings necessary for API server deployments in the cluster

Field	Description
`ControlPlaneComponent` **[Required]** `ControlPlaneComponent`	(Members of `ControlPlaneComponent` are embedded into this type.) No description provided.
`certSANs` `[]string`	`certSANs` sets extra Subject Alternative Names (SANs) for the API Server signing certificate.
`timeoutForControlPlane` `meta/v1.Duration`	`timeoutForControlPlane` controls the timeout that we wait for API server to appear.

`BootstrapTokenDiscovery`

Appears in:

- Discovery

BootstrapTokenDiscovery is used to set the options for bootstrap token based discovery.

Field	Description
`token` **[Required]** `string`	`token` is a token used to validate cluster information fetched from the control-plane.
`apiServerEndpoint` `string`	`apiServerEndpoint` is an IP or domain name to the API server from which information will be fetched.
`caCertHashes` `[]string`	`caCertHashes` specifies a set of public key pins to verify when token-based discovery is used. The root CA found during discovery must match one of these values. Specifying an empty set disables root CA pinning, which can be unsafe. Each hash is specified as `<type>:<value>`, where the only currently supported type is "sha256". This is a hex-encoded SHA-256 hash of the

Field	Description
	Subject Public Key Info (SPKI) object in DER-encoded ASN.1. These hashes can be calculated using, for example, OpenSSL.
`unsafeSkipCAVerification` `bool`	`unsafeSkipCAVerification` allows token-based discovery without CA verification via `caCertHashes`. This can weaken the security of kubeadm since other nodes can impersonate the control-plane.

`ControlPlaneComponent`

Appears in:

- ClusterConfiguration
- APIServer

ControlPlaneComponent holds settings common to control plane component of the cluster

Field	Description
`extraArgs` `map[string]string`	`extraArgs` is an extra set of flags to pass to the control plane component. A key in this map is the flag name as it appears on the command line except without leading dash(es).
`extraVolumes` `[]HostPathMount`	`extraVolumes` is an extra set of host volumes, mounted to the control plane component.

`DNS`

Appears in:

- ClusterConfiguration

DNS defines the DNS addon that should be used in the cluster

Field	Description
`ImageMeta` **[Required]** `ImageMeta`	(Members of `ImageMeta` are embedded into this type.) `imageMeta` allows to customize the image used for the DNS component.

`Discovery`

Appears in:

- JoinConfiguration

Discovery specifies the options for the kubelet to use during the TLS Bootstrap process.

Field	Description
`bootstrapToken` `BootstrapTokenDiscovery`	`bootstrapToken` is used to set the options for bootstrap token based discovery. `bootstrapToken` and `file` are mutually exclusive.
`file` `FileDiscovery`	`file` is used to specify a file or URL to a kubeconfig file from which to load cluster information. `bootstrapToken` and `file` are mutually exclusive.
`tlsBootstrapToken` `string`	`tlsBootstrapToken` is a token used for TLS bootstrapping. If `bootstrapToken` is set, this field is defaulted to `.bootstrapToken.token`, but can be overridden. If `file` is set, this field **must be set** in case the KubeConfigFile does not contain any other authentication information
`timeout` `meta/v1.Duration`	`timeout` modifies the discovery timeout.

`Etcd`

Appears in:

- ClusterConfiguration

Etcd contains elements describing Etcd configuration.

Field	Description
`local` `LocalEtcd`	`local` provides configuration knobs for configuring the local etcd instance. `local` and `external` are mutually exclusive.
`external` `ExternalEtcd`	`external` describes how to connect to an external etcd cluster. `local` and `external` are mutually exclusive.

ExternalEtcd

Appears in:

- Etcd

ExternalEtcd describes an external etcd cluster. Kubeadm has no knowledge of where certificate files live and they must be supplied.

Field	Description
`endpoints` **[Required]** `[]string`	`endpoints` contains the list of etcd members.
`caFile` **[Required]** `string`	`caFile` is an SSL Certificate Authority (CA) file used to secure etcd communication. Required if using a TLS connection.
`certFile` **[Required]** `string`	`certFile` is an SSL certification file used to secure etcd communication. Required if using a TLS connection.
`keyFile` **[Required]** `string`	`keyFile` is an SSL key file used to secure etcd communication. Required if using a TLS connection.

FileDiscovery

Appears in:

- Discovery

FileDiscovery is used to specify a file or URL to a kubeconfig file from which to load cluster information.

Field	Description
`kubeConfigPath` **[Required]** `string`	`kubeConfigPath` is used to specify the actual file path or URL to the kubeconfig file from which to load cluster information.

HostPathMount

Appears in:

- ControlPlaneComponent

HostPathMount contains elements describing volumes that are mounted from the host.

Field	Description
`name` **[Required]** `string`	`name` is the name of the volume inside the Pod template.
`hostPath` **[Required]** `string`	`hostPath` is the path in the host that will be mounted inside the Pod.
`mountPath` **[Required]** `string`	`mountPath` is the path inside the Pod where `hostPath` will be mounted.
`readOnly` `bool`	`readOnly` controls write access to the volume.
`pathType` `core/v1.HostPathType`	`pathType` is the type of the `hostPath`.

ImageMeta

Appears in:

- DNS
- LocalEtcd

ImageMeta allows to customize the image used for components that are not originated from the Kubernetes/Kubernetes release process

Field	Description
`imageRepository` `string`	`imageRepository` sets the container registry to pull images from. If not set, the `imageRepository` defined in ClusterConfiguration will be used instead.
`imageTag` `string`	`imageTag` allows to specify a tag for the image. In case this value is set, kubeadm does not change automatically the version of the above components during upgrades.

JoinControlPlane

Appears in:

- JoinConfiguration

JoinControlPlane contains elements describing an additional control plane instance to be deployed on the joining node.

Field	Description
`localAPIEndpoint` APIEndpoint	`localAPIEndpoint` represents the endpoint of the API server instance to be deployed on this node.
`certificateKey` string	`certificateKey` is the key that is used for decryption of certificates after they are downloaded from the secret upon joining a new control plane node. The corresponding encryption key is in the InitConfiguration. The certificate key is a hex encoded string that is an AES key of size 32 bytes.

LocalEtcd

Appears in:

- Etcd

LocalEtcd describes that kubeadm should run an etcd cluster locally.

Field	Description
`ImageMeta` **[Required]** ImageMeta	(Members of `ImageMeta` are embedded into this type.) ImageMeta allows to customize the container used for etcd.
`dataDir` **[Required]** string	`dataDir` is the directory etcd will place its data. Defaults to "/var/lib/etcd".
`extraArgs` map[string]string	`extraArgs` are extra arguments provided to the etcd binary when run inside a static Pod. A key in this map is the flag name as it appears on the command line except without leading dash(es).
`serverCertSANs` []string	`serverCertSANs` sets extra Subject Alternative Names (SANs) for the etcd server signing certificate.
`peerCertSANs` []string	`peerCertSANs` sets extra Subject Alternative Names (SANs) for the etcd peer signing certificate.

Networking

Appears in:

- ClusterConfiguration

Networking contains elements describing cluster's networking configuration.

Field	Description
`serviceSubnet` string	`serviceSubnet` is the subnet used by Kubernetes Services. Defaults to "10.96.0.0/12".
`podSubnet` string	`podSubnet` is the subnet used by Pods.
`dnsDomain` string	`dnsDomain` is the DNS domain used by Kubernetes Services. Defaults to "cluster.local".

NodeRegistrationOptions

Appears in:

- InitConfiguration
- JoinConfiguration

NodeRegistrationOptions holds fields that relate to registering a new control-plane or node to the cluster, either via `kubeadm init` or `kubeadm join`.

Field	Description
`name` string	`name` is the `.metadata.name` field of the Node API object that will be created in this `kubeadm init` or `kubeadm join` operation. This field is also used in the `CommonName` field of the kubelet's client certificate to the API server. Defaults to the hostname of the node if not provided.
`criSocket` string	`criSocket` is used to retrieve container runtime info. This information will be annotated to the Node API object, for later re-use.
`taints` **[Required]** `[]core/v1.Taint`	`taints` specifies the taints the Node API object should be registered with. If this field is unset, i.e. nil, it will be defaulted with a control-plane taint for control-plane nodes. If you don't want to

Field	Description
	taint your control-plane node, set this field to an empty list, i.e. `taints: []` in the YAML file. This field is solely used for Node registration.
`kubeletExtraArgs` `map[string]string`	`kubeletExtraArgs` passes through extra arguments to the kubelet. The arguments here are passed to the kubelet command line via the environment file kubeadm writes at runtime for the kubelet to source. This overrides the generic base-level configuration in the `kubelet-config` ConfigMap. Flags have higher priority when parsing. These values are local and specific to the node kubeadm is executing on. A key in this map is the flag name as it appears on the command line except without leading dash(es).
`ignorePreflightErrors` `[]string`	`ignorePreflightErrors` provides a list of pre-flight errors to be ignored when the current node is registered, e.g. `IsPrivilegedUser,Swap`. Value `all` ignores errors from all checks.
`imagePullPolicy` `core/v1.PullPolicy`	`imagePullPolicy` specifies the policy for image pulling during kubeadm "init" and "join" operations. The value of this field must be one of "Always", "IfNotPresent" or "Never". If this field is not set, kubeadm will default it to "IfNotPresent", or pull the required images if not present on the host.

`Patches`

Appears in:

- InitConfiguration
- JoinConfiguration

Patches contains options related to applying patches to components deployed by kubeadm.

Field	Description
`directory` `string`	`directory` is a path to a directory that contains files named "target[suffix][+patchtype].extension". For example, "kube-apiserver0+merge.yaml" or just "etcd.json". "target" can be one of "kube-apiserver", "kube-controller-manager", "kube-scheduler", "etcd". "patchtype" can be one of "strategic" "merge" or "json" and they match the patch formats supported by kubectl. The default "patchtype" is "strategic". "extension" must be either "json" or "yaml". "suffix" is an optional string that can be used to determine which patches are applied first alpha-numerically.

Kubelet Configuration (v1beta1)

Resource Types

- CredentialProviderConfig
- KubeletConfiguration
- SerializedNodeConfigSource

`FormatOptions`

Appears in:

- LoggingConfiguration

FormatOptions contains options for the different logging formats.

Field	Description
`text` **[Required]** `TextOptions`	[Alpha] Text contains options for logging format "text". Only available when the LoggingAlphaOptions feature gate is enabled.
`json` **[Required]** `JSONOptions`	[Alpha] JSON contains options for logging format "json". Only available when the LoggingAlphaOptions feature gate is enabled.

`JSONOptions`

Appears in:

- FormatOptions

JSONOptions contains options for logging format "json".

Field	Description
`OutputRoutingOptions` **[Required]** OutputRoutingOptions	(Members of `OutputRoutingOptions` are embedded into this type.) No description provided.

LogFormatFactory

LogFormatFactory provides support for a certain additional, non-default log format.

LoggingConfiguration

Appears in:

- KubeletConfiguration

LoggingConfiguration contains logging options.

Field	Description
`format` **[Required]** `string`	Format Flag specifies the structure of log messages. default value of format is `text`
`flushFrequency` **[Required]** TimeOrMetaDuration	Maximum time between log flushes. If a string, parsed as a duration (i.e. "1s") If an int, the maximum number of nanoseconds (i.e. 1s = 1000000000). Ignored if the selected logging backend writes log messages without buffering.
`verbosity` **[Required]** VerbosityLevel	Verbosity is the threshold that determines which log messages are logged. Default is zero which logs only the most important messages. Higher values enable additional messages. Error messages are always logged.
`vmodule` **[Required]** VModuleConfiguration	VModule overrides the verbosity threshold for individual files. Only supported for "text" log format.
`options` **[Required]** FormatOptions	[Alpha] Options holds additional parameters that are specific to the different logging formats. Only the options for the selected format get used, but all of them get validated. Only available when the LoggingAlphaOptions feature gate is enabled.

LoggingOptions

LoggingOptions can be used with ValidateAndApplyWithOptions to override certain global defaults.

Field	Description
`ErrorStream` **[Required]** io.Writer	ErrorStream can be used to override the os.Stderr default.
`InfoStream` **[Required]** io.Writer	InfoStream can be used to override the os.Stdout default.

OutputRoutingOptions

Appears in:

- JSONOptions
- TextOptions

OutputRoutingOptions contains options that are supported by both "text" and "json".

Field	Description
`splitStream` **[Required]** `bool`	[Alpha] SplitStream redirects error messages to stderr while info messages go to stdout, with buffering. The default is to write both to stdout, without buffering. Only available when the LoggingAlphaOptions feature gate is enabled.
`infoBufferSize` **[Required]** k8s.io/apimachinery/pkg/api/resource.QuantityValue	[Alpha] InfoBufferSize sets the size of the info stream when using split streams. The default is zero, which disables buffering. Only available when the LoggingAlphaOptions feature gate is enabled.

TextOptions

Appears in:

- FormatOptions

TextOptions contains options for logging format "text".

Field	Description
`OutputRoutingOptions` **[Required]** `OutputRoutingOptions`	(Members of `OutputRoutingOptions` are embedded into this type.) No description provided.

`TimeOrMetaDuration`

Appears in:

- LoggingConfiguration

TimeOrMetaDuration is present only for backwards compatibility for the flushFrequency field, and new fields should use metav1.Duration.

Field	Description
`Duration` **[Required]** `meta/v1.Duration`	Duration holds the duration
`-` **[Required]** `bool`	SerializeAsString controls whether the value is serialized as a string or an integer

`TracingConfiguration`

Appears in:

- KubeletConfiguration

TracingConfiguration provides versioned configuration for OpenTelemetry tracing clients.

Field	Description
`endpoint` `string`	Endpoint of the collector this component will report traces to. The connection is insecure, and does not currently support TLS. Recommended is unset, and endpoint is the otlp grpc default, localhost:4317.
`samplingRatePerMillion` `int32`	SamplingRatePerMillion is the number of samples to collect per million spans. Recommended is unset. If unset, sampler respects its parent span's sampling rate, but otherwise never samples.

`VModuleConfiguration`

(Alias of `[]k8s.io/component-base/logs/api/v1.VModuleItem`)

Appears in:

- LoggingConfiguration

VModuleConfiguration is a collection of individual file names or patterns and the corresponding verbosity threshold.

`VerbosityLevel`

(Alias of `uint32`)

Appears in:

- LoggingConfiguration

VerbosityLevel represents a klog or logr verbosity threshold.

`CredentialProviderConfig`

CredentialProviderConfig is the configuration containing information about each exec credential provider. Kubelet reads this configuration from disk and enables each provider as specified by the CredentialProvider type.

Field	Description
`apiVersion` string	`kubelet.config.k8s.io/v1beta1`
`kind` string	`CredentialProviderConfig`
`providers` **[Required]** `[]CredentialProvider`	providers is a list of credential provider plugins that will be enabled by the kubelet. Multiple providers may match against a single image, in which case credentials from all providers will be returned to the kubelet. If multiple providers are called for a single image, the results are combined. If providers return overlapping auth keys, the value from the provider earlier in this list is attempted first.

KubeletConfiguration

KubeletConfiguration contains the configuration for the Kubelet

Field	Description
apiVersion string	kubelet.config.k8s.io/v1beta1
kind string	KubeletConfiguration
enableServer **[Required]** bool	enableServer enables Kubelet's secured server. Note: Kubelet's insecure port is controlled by the readOnlyPort option. Default: true
staticPodPath string	staticPodPath is the path to the directory containing local (static) pods to run, or the path to a single static pod file. Default: ""
podLogsDir string	podLogsDir is a custom root directory path kubelet will use to place pod's log files. Default: "/var/log/pods/" Note: it is not recommended to use the temp folder as a log directory as it may cause unexpected behavior in many places.
syncFrequency meta/v1.Duration	syncFrequency is the max period between synchronizing running containers and config. Default: "1m"
fileCheckFrequency meta/v1.Duration	fileCheckFrequency is the duration between checking config files for new data. Default: "20s"
httpCheckFrequency meta/v1.Duration	httpCheckFrequency is the duration between checking http for new data. Default: "20s"
staticPodURL string	staticPodURL is the URL for accessing static pods to run. Default: ""
staticPodURLHeader map[string][]string	staticPodURLHeader is a map of slices with HTTP headers to use when accessing the podURL. Default: nil
address string	address is the IP address for the Kubelet to serve on (set to 0.0.0.0 for all interfaces). Default: "0.0.0.0"
port int32	port is the port for the Kubelet to serve on. The port number must be between 1 and 65535, inclusive. Default: 10250
readOnlyPort int32	readOnlyPort is the read-only port for the Kubelet to serve on with no authentication/authorization. The port number must be between 1 and 65535, inclusive. Setting this field to 0 disables the read-only service. Default: 0 (disabled)
tlsCertFile string	tlsCertFile is the file containing x509 Certificate for HTTPS. (CA cert, if any, concatenated after server cert). If tlsCertFile and tlsPrivateKeyFile are not provided, a self-signed certificate and key are generated for the public address and saved to the directory passed to the Kubelet's --cert-dir flag. Default: ""
tlsPrivateKeyFile string	tlsPrivateKeyFile is the file containing x509 private key matching tlsCertFile. Default: ""
tlsCipherSuites []string	tlsCipherSuites is the list of allowed cipher suites for the server. Note that TLS 1.3 ciphersuites are not configurable. Values are from tls package constants (https://golang.org/pkg/crypto/tls/#pkg-constants). Default: nil
tlsMinVersion string	tlsMinVersion is the minimum TLS version supported. Values are from tls package constants (https://golang.org/pkg/crypto/tls/#pkg-constants). Default: ""
rotateCertificates bool	rotateCertificates enables client certificate rotation. The Kubelet will request a new certificate from the certificates.k8s.io API. This requires an approver to approve the certificate signing requests. Default: false
serverTLSBootstrap bool	serverTLSBootstrap enables server certificate bootstrap. Instead of self signing a serving certificate, the Kubelet will request a certificate from the 'certificates.k8s.io' API. This requires an approver to approve the certificate signing requests (CSR). The RotateKubeletServerCertificate feature must be enabled when setting this field. Default: false
authentication KubeletAuthentication	authentication specifies how requests to the Kubelet's server are authenticated. Defaults: anonymous: enabled: false webhook: enabled: true cacheTTL: "2m"
authorization KubeletAuthorization	authorization specifies how requests to the Kubelet's server are authorized. Defaults: mode: Webhook webhook: cacheAuthorizedTTL: "5m" cacheUnauthorizedTTL: "30s"
registryPullQPS int32	registryPullQPS is the limit of registry pulls per second. The value must not be a negative number. Setting it to 0 means no limit. Default: 5
registryBurst int32	registryBurst is the maximum size of bursty pulls, temporarily allows pulls to burst to this number, while still not exceeding registryPullQPS. The value must not be a negative number. Only used if registryPullQPS is greater than 0. Default: 10
imagePullCredentialsVerificationPolicy ImagePullCredentialsVerificationPolicy	imagePullCredentialsVerificationPolicy determines how credentials should be verified when pod requests an image that is already present on the node: • NeverVerify ◦ anyone on a node can use any image present on the node • NeverVerifyPreloadedImages

Field	Description
	- images that were pulled to the node by something else than the kubelet can be used without reverifying pull credentials - NeverVerifyAllowlistedImages - like "NeverVerifyPreloadedImages" but only node images from <code>preloadedImagesVerificationAllowlist</code> don't require reverification - AlwaysVerify - all images require credential reverification
<code>preloadedImagesVerificationAllowlist</code> <code>[]string</code>	preloadedImagesVerificationAllowlist specifies a list of images that are exempted from credential reverification for the "NeverVerifyAllowlistedImages" <code>imagePullCredentialsVerificationPolicy</code>. The list accepts a full path segment wildcard suffix "/*". Only use image specs without an image tag or digest.
<code>eventRecordQPS</code> <code>int32</code>	eventRecordQPS is the maximum event creations per second. If 0, there is no limit enforced. The value cannot be a negative number. Default: 50
<code>eventBurst</code> <code>int32</code>	eventBurst is the maximum size of a burst of event creations, temporarily allows event creations to burst to this number, while still not exceeding eventRecordQPS. This field canot be a negative number and it is only used when eventRecordQPS > 0. Default: 100
<code>enableDebuggingHandlers</code> <code>bool</code>	enableDebuggingHandlers enables server endpoints for log access and local running of containers and commands, including the exec, attach, logs, and portforward features. Default: true
<code>enableContentionProfiling</code> <code>bool</code>	enableContentionProfiling enables block profiling, if enableDebuggingHandlers is true. Default: false
<code>healthzPort</code> <code>int32</code>	healthzPort is the port of the localhost healthz endpoint (set to 0 to disable). A valid number is between 1 and 65535. Default: 10248
<code>healthzBindAddress</code> <code>string</code>	healthzBindAddress is the IP address for the healthz server to serve on. Default: "127.0.0.1"
<code>oomScoreAdj</code> <code>int32</code>	oomScoreAdj is The oom-score-adj value for kubelet process. Values must be within the range [-1000, 1000]. Default: -999
<code>clusterDomain</code> <code>string</code>	clusterDomain is the DNS domain for this cluster. If set, kubelet will configure all containers to search this domain in addition to the host's search domains. Default: ""
<code>clusterDNS</code> <code>[]string</code>	clusterDNS is a list of IP addresses for the cluster DNS server. If set, kubelet will configure all containers to use this for DNS resolution instead of the host's DNS servers. Default: nil
<code>streamingConnectionIdleTimeout</code> <code>meta/v1.Duration</code>	streamingConnectionIdleTimeout is the maximum time a streaming connection can be idle before the connection is automatically closed. Deprecated: no longer has any effect. Default: "4h"
<code>nodeStatusUpdateFrequency</code> <code>meta/v1.Duration</code>	nodeStatusUpdateFrequency is the frequency that kubelet computes node status. If node lease feature is not enabled, it is also the frequency that kubelet posts node status to master. Note: When node lease feature is not enabled, be cautious when changing the constant, it must work with nodeMonitorGracePeriod in nodecontroller. Default: "10s"
<code>nodeStatusReportFrequency</code> <code>meta/v1.Duration</code>	nodeStatusReportFrequency is the frequency that kubelet posts node status to master if node status does not change. Kubelet will ignore this frequency and post node status immediately if any change is detected. It is only used when node lease feature is enabled. nodeStatusReportFrequency's default value is 5m. But if nodeStatusUpdateFrequency is set explicitly, nodeStatusReportFrequency's default value will be set to nodeStatusUpdateFrequency for backward compatibility. Default: "5m"
<code>nodeLeaseDurationSeconds</code> <code>int32</code>	nodeLeaseDurationSeconds is the duration the Kubelet will set on its corresponding Lease. NodeLease provides an indicator of node health by having the Kubelet create and periodically renew a lease, named after the node, in the kube-node-lease namespace. If the lease expires, the node can be considered unhealthy. The lease is currently renewed every 10s, per KEP-0009. In the future, the lease renewal interval may be set based on the lease duration. The field value must be greater than 0. Default: 40
<code>imageMinimumGCAge</code> <code>meta/v1.Duration</code>	imageMinimumGCAge is the minimum age for an unused image before it is garbage collected. Default: "2m"
<code>imageMaximumGCAge</code> <code>meta/v1.Duration</code>	imageMaximumGCAge is the maximum age an image can be unused before it is garbage collected. The default of this field is "0s", which disables this field--meaning images won't be garbage collected based on being unused for too long. Default: "0s" (disabled)
<code>imageGCHighThresholdPercent</code> <code>int32</code>	imageGCHighThresholdPercent is the percent of disk usage after which image garbage collection is always run. The percent is calculated by dividing this field value by 100, so this field must be between 0 and 100, inclusive. When specified, the value must be greater than imageGCLowThresholdPercent. Default: 85
<code>imageGCLowThresholdPercent</code> <code>int32</code>	imageGCLowThresholdPercent is the percent of disk usage before which image garbage collection is never run. Lowest disk usage to garbage collect to. The percent is calculated by dividing this field value by 100, so the field value must be between 0 and

Field	Description
	100, inclusive. When specified, the value must be less than imageGCHighThresholdPercent. Default: 80
`volumeStatsAggPeriod` meta/v1.Duration	volumeStatsAggPeriod is the frequency for calculating and caching volume disk usage for all pods. Default: "1m"
`kubeletCgroups` `string`	kubeletCgroups is the absolute name of cgroups to isolate the kubelet in Default: ""
`systemCgroups` `string`	systemCgroups is absolute name of cgroups in which to place all non-kernel processes that are not already in a container. Empty for no container. Rolling back the flag requires a reboot. The cgroupRoot must be specified if this field is not empty. Default: ""
`cgroupRoot` `string`	cgroupRoot is the root cgroup to use for pods. This is handled by the container runtime on a best effort basis.
`cgroupsPerQOS` `bool`	cgroupsPerQOS enable QoS based CGroup hierarchy: top level CGroups for QoS classes and all Burstable and BestEffort Pods are brought up under their specific top level QoS CGroup. Default: true
`cgroupDriver` `string`	cgroupDriver is the driver kubelet uses to manipulate CGroups on the host (cgroupfs or systemd). Default: "cgroupfs"
`cpuManagerPolicy` `string`	cpuManagerPolicy is the name of the policy to use. Default: "None"
`singleProcessOOMKill` `bool`	singleProcessOOMKill, if true, will prevent the `memory.oom.group` flag from being set for container cgroups in cgroups v2. This causes processes in the container to be OOM killed individually instead of as a group. It means that if true, the behavior aligns with the behavior of cgroups v1. The default value is determined automatically when you don't specify. On non-linux such as windows, only null / absent is allowed. On cgroup v1 linux, only null / absent and true are allowed. On cgroup v2 linux, null / absent, true and false are allowed. The default value is false.
`cpuManagerPolicyOptions` `map[string]string`	cpuManagerPolicyOptions is a set of key=value which allows to set extra options to fine tune the behaviour of the cpu manager policies. Default: nil
`cpuManagerReconcilePeriod` meta/v1.Duration	cpuManagerReconcilePeriod is the reconciliation period for the CPU Manager. Default: "10s"
`memoryManagerPolicy` `string`	memoryManagerPolicy is the name of the policy to use by memory manager. Requires the MemoryManager feature gate to be enabled. Default: "none"
`topologyManagerPolicy` `string`	topologyManagerPolicy is the name of the topology manager policy to use. Valid values include: • `restricted`: kubelet only allows pods with optimal NUMA node alignment for requested resources; • `best-effort`: kubelet will favor pods with NUMA alignment of CPU and device resources; • `none`: kubelet has no knowledge of NUMA alignment of a pod's CPU and device resources. • `single-numa-node`: kubelet only allows pods with a single NUMA alignment of CPU and device resources. Default: "none"
`topologyManagerScope` `string`	topologyManagerScope represents the scope of topology hint generation that topology manager requests and hint providers generate. Valid values include: • `container`: topology policy is applied on a per-container basis. • `pod`: topology policy is applied on a per-pod basis. Default: "container"
`topologyManagerPolicyOptions` `map[string]string`	TopologyManagerPolicyOptions is a set of key=value which allows to set extra options to fine tune the behaviour of the topology manager policies. Requires both the "TopologyManager" and "TopologyManagerPolicyOptions" feature gates to be enabled. Default: nil
`qosReserved` `map[string]string`	qosReserved is a set of resource name to percentage pairs that specify the minimum percentage of a resource reserved for exclusive use by the guaranteed QoS tier. Currently supported resources: "memory" Requires the QOSReserved feature gate to be enabled. Default: nil
`runtimeRequestTimeout` meta/v1.Duration	runtimeRequestTimeout is the timeout for all runtime requests except long running requests - pull, logs, exec and attach. Default: "2m"
`hairpinMode` `string`	hairpinMode specifies how the Kubelet should configure the container bridge for hairpin packets. Setting this flag allows endpoints in a Service to loadbalance back to themselves if they should try to access their own Service. Values: • "promiscuous-bridge": make the container bridge promiscuous. • "hairpin-veth": set the hairpin flag on container veth interfaces. • "none": do nothing.

Field	Description
	Generally, one must set `--hairpin-mode=hairpin-veth to` achieve hairpin NAT, because promiscuous-bridge assumes the existence of a container bridge named cbr0. Default: "promiscuous-bridge"
`maxPods` `int32`	maxPods is the maximum number of Pods that can run on this Kubelet. The value must be a non-negative integer. Default: 110
`podCIDR` `string`	podCIDR is the CIDR to use for pod IP addresses, only used in standalone mode. In cluster mode, this is obtained from the control plane. Default: ""
`podPidsLimit` `int64`	podPidsLimit is the maximum number of PIDs in any pod. Default: -1
`resolvConf` `string`	resolvConf is the resolver configuration file used as the basis for the container DNS resolution configuration. If set to the empty string, will override the default and effectively disable DNS lookups. Default: "/etc/resolv.conf"
`runOnce` `bool`	runOnce causes the Kubelet to check the API server once for pods, run those in addition to the pods specified by static pod files, and exit. Default: false
`cpuCFSQuota` `bool`	cpuCFSQuota enables CPU CFS quota enforcement for containers that specify CPU limits. Default: true
`cpuCFSQuotaPeriod` `meta/v1.Duration`	cpuCFSQuotaPeriod is the CPU CFS quota period value, `cpu.cfs_period_us`. The value must be between 1 ms and 1 second, inclusive. Requires the CustomCPUCFSQuotaPeriod feature gate to be enabled. Default: "100ms"
`nodeStatusMaxImages` `int32`	nodeStatusMaxImages caps the number of images reported in Node.status.images. The value must be greater than -2. Note: If -1 is specified, no cap will be applied. If 0 is specified, no image is returned. Default: 50
`maxOpenFiles` `int64`	maxOpenFiles is Number of files that can be opened by Kubelet process. The value must be a non-negative number. Default: 1000000
`contentType` `string`	contentType is contentType of requests sent to apiserver. Default: "application/vnd.kubernetes.protobuf"
`kubeAPIQPS` `int32`	kubeAPIQPS is the QPS to use while talking with kubernetes apiserver. Default: 50
`kubeAPIBurst` `int32`	kubeAPIBurst is the burst to allow while talking with kubernetes API server. This field cannot be a negative number. Default: 100
`serializeImagePulls` `bool`	serializeImagePulls when enabled, tells the Kubelet to pull images one at a time. We recommend *not* changing the default value on nodes that run docker daemon with version < 1.9 or an Aufs storage backend. Issue #10959 has more details. Default: true
`maxParallelImagePulls` `int32`	MaxParallelImagePulls sets the maximum number of image pulls in parallel. This field cannot be set if SerializeImagePulls is true. Setting it to nil means no limit. Default: nil
`evictionHard` `map[string]string`	evictionHard is a map of signal names to quantities that defines hard eviction thresholds. For example: `{"memory.available": "300Mi"}`. To explicitly disable, pass a 0% or 100% threshold on an arbitrary resource. Default: memory.available: "100Mi" nodefs.available: "10%" nodefs.inodesFree: "5%" imagefs.available: "15%"
`evictionSoft` `map[string]string`	evictionSoft is a map of signal names to quantities that defines soft eviction thresholds. For example: `{"memory.available": "300Mi"}`. Default: nil
`evictionSoftGracePeriod` `map[string]string`	evictionSoftGracePeriod is a map of signal names to quantities that defines grace periods for each soft eviction signal. For example: `{"memory.available": "30s"}`. Default: nil
`evictionPressureTransitionPeriod` `meta/v1.Duration`	evictionPressureTransitionPeriod is the duration for which the kubelet has to wait before transitioning out of an eviction pressure condition. A duration of 0s will be converted to the default value of 5m Default: "5m"
`evictionMaxPodGracePeriod` `int32`	evictionMaxPodGracePeriod is the maximum allowed grace period (in seconds) to use when terminating pods in response to a soft eviction threshold being met. This value effectively caps the Pod's terminationGracePeriodSeconds value during soft evictions. The pod's effective grace period is calculated as: min(evictionMaxPodGracePeriod, pod.terminationGracePeriodSeconds). Note: A negative value will cause pods to be terminated immediately, as if the value was 0. Default: 0
`evictionMinimumReclaim` `map[string]string`	evictionMinimumReclaim is a map of signal names to quantities that defines minimum reclaims, which describe the minimum amount of a given resource the kubelet will reclaim when performing a pod eviction while that resource is under pressure. For example: `{"imagefs.available": "2Gi"}`. Default: nil
`mergeDefaultEvictionSettings` `bool`	mergeDefaultEvictionSettings indicates that defaults for the evictionHard, evictionSoft, evictionSoftGracePeriod, and evictionMinimumReclaim fields should be merged into values specified for those fields in this configuration. Signals specified in this configuration take precedence. Signals not specified in this configuration inherit their defaults. If false, and if any signal is specified in this configuration then other signals that are not specified in this configuration will be set to 0. It applies to merging the fields for which the default exists, and currently only evictionHard has default values. Default: false
`podsPerCore` `int32`	podsPerCore is the maximum number of pods per core. Cannot exceed maxPods. The value must be a non-negative integer. If 0, there is no limit on the number of Pods.

Field	Description
	Default: 0
`enableControllerAttachDetach` `bool`	enableControllerAttachDetach enables the Attach/Detach controller to manage attachment/detachment of volumes scheduled to this node, and disables kubelet from executing any attach/detach operations. Note: attaching/detaching CSI volumes is not supported by the kubelet, so this option needs to be true for that use case. Default: true
`protectKernelDefaults` `bool`	protectKernelDefaults, if true, causes the Kubelet to error if kernel flags are not as it expects. Otherwise the Kubelet will attempt to modify kernel flags to match its expectation. Default: false
`makeIPTablesUtilChains` `bool`	makeIPTablesUtilChains, if true, causes the Kubelet to create the KUBE-IPTABLES-HINT chain in iptables as a hint to other components about the configuration of iptables on the system. Default: true
`iptablesMasqueradeBit` `int32`	iptablesMasqueradeBit formerly controlled the creation of the KUBE-MARK-MASQ chain. Deprecated: no longer has any effect. Default: 14
`iptablesDropBit` `int32`	iptablesDropBit formerly controlled the creation of the KUBE-MARK-DROP chain. Deprecated: no longer has any effect. Default: 15
`featureGates` `map[string]bool`	featureGates is a map of feature names to bools that enable or disable experimental features. This field modifies piecemeal the built-in default values from "k8s.io/kubernetes/pkg/features/kube_features.go". Default: nil
`failSwapOn` `bool`	failSwapOn tells the Kubelet to fail to start if swap is enabled on the node. Default: true
`memorySwap` `MemorySwapConfiguration`	memorySwap configures swap memory available to container workloads.
`containerLogMaxSize` `string`	containerLogMaxSize is a quantity defining the maximum size of the container log file before it is rotated. For example: "5Mi" or "256Ki". Default: "10Mi"
`containerLogMaxFiles` `int32`	containerLogMaxFiles specifies the maximum number of container log files that can be present for a container. Default: 5
`containerLogMaxWorkers` `int32`	ContainerLogMaxWorkers specifies the maximum number of concurrent workers to spawn for performing the log rotate operations. Set this count to 1 for disabling the concurrent log rotation workflows Default: 1
`containerLogMonitorInterval` `meta/v1.Duration`	ContainerLogMonitorInterval specifies the duration at which the container logs are monitored for performing the log rotate operation. This defaults to 10 * time.Seconds. But can be customized to a smaller value based on the log generation rate and the size required to be rotated against Default: 10s
`configMapAndSecretChangeDetectionStrategy` `ResourceChangeDetectionStrategy`	configMapAndSecretChangeDetectionStrategy is a mode in which ConfigMap and Secret managers are running. Valid values include: • `Get`: kubelet fetches necessary objects directly from the API server; • `Cache`: kubelet uses TTL cache for object fetched from the API server; • `Watch`: kubelet uses watches to observe changes to objects that are in its interest. Default: "Watch"
`systemReserved` `map[string]string`	systemReserved is a set of ResourceName=ResourceQuantity (e.g. cpu=200m,memory=150G) pairs that describe resources reserved for non-kubernetes components. Currently only cpu and memory are supported. See https://kubernetes.io/docs/tasks/administer-cluster/reserve-compute-resources for more detail. Default: nil
`kubeReserved` `map[string]string`	kubeReserved is a set of ResourceName=ResourceQuantity (e.g. cpu=200m,memory=150G) pairs that describe resources reserved for kubernetes system components. Currently cpu, memory and local storage for root file system are supported. See https://kubernetes.io/docs/tasks/administer-cluster/reserve-compute-resources for more details. Default: nil
`reservedSystemCPUs` **[Required]** `string`	The reservedSystemCPUs option specifies the CPU list reserved for the host level system threads and kubernetes related threads. This provide a "static" CPU list rather than the "dynamic" list by systemReserved and kubeReserved. This option does not support systemReservedCgroup or kubeReservedCgroup.
`showHiddenMetricsForVersion` `string`	showHiddenMetricsForVersion is the previous version for which you want to show hidden metrics. Only the previous minor version is meaningful, other values will not be allowed. The format is `<major>.<minor>`, e.g.: `1.16`. The purpose of this format is make sure you have the opportunity to notice if the next release hides additional metrics, rather than being surprised when they are permanently removed in the release after that. Default: ""
`systemReservedCgroup` `string`	systemReservedCgroup helps the kubelet identify absolute name of top level CGroup used to enforce `systemReserved` compute resource reservation for OS system daemons. Refer to Node Allocatable doc for more information. Default: ""
`kubeReservedCgroup` `string`	kubeReservedCgroup helps the kubelet identify absolute name of top level CGroup used to enforce `KubeReserved` compute resource reservation for Kubernetes node system daemons. Refer to Node Allocatable doc for more information. Default: ""

Field	Description
enforceNodeAllocatable []string	This flag specifies the various Node Allocatable enforcements that Kubelet needs to perform. This flag accepts a list of options. Acceptable options are `none`, `pods`, `system-reserved` and `kube-reserved`. If `none` is specified, no other options may be specified. When `system-reserved` is in the list, systemReservedCgroup must be specified. When `kube-reserved` is in the list, kubeReservedCgroup must be specified. This field is supported only when `cgroupsPerQOS` is set to true. Refer to Node Allocatable for more information. Default: ["pods"]
allowedUnsafeSysctls []string	A comma separated whitelist of unsafe sysctls or sysctl patterns (ending in `*`). Unsafe sysctl groups are `kernel.shm*`, `kernel.msg*`, `kernel.sem`, `fs.mqueue.*`, and `net.*`. For example: "`kernel.msg*,net.ipv4.route.min_pmtu`" Default: []
volumePluginDir string	volumePluginDir is the full path of the directory in which to search for additional third party volume plugins. Default: "/usr/libexec/kubernetes/kubelet-plugins/volume/exec/"
providerID string	providerID, if set, sets the unique ID of the instance that an external provider (i.e. cloudprovider) can use to identify a specific node. Default: ""
kernelMemcgNotification bool	kernelMemcgNotification, if set, instructs the kubelet to integrate with the kernel memcg notification for determining if memory eviction thresholds are exceeded rather than polling. Default: false
`logging` **[Required]** LoggingConfiguration	logging specifies the options of logging. Refer to Logs Options for more information. Default: Format: text
enableSystemLogHandler bool	enableSystemLogHandler enables system logs via web interface host:port/logs/ Default: true
enableSystemLogQuery bool	enableSystemLogQuery enables the node log query feature on the /logs endpoint. EnableSystemLogHandler has to be enabled in addition for this feature to work. Enabling this feature has security implications. The recommendation is to enable it on a need basis for debugging purposes and disabling otherwise. Default: false
shutdownGracePeriod meta/v1.Duration	shutdownGracePeriod specifies the total duration that the node should delay the shutdown and total grace period for pod termination during a node shutdown. Default: "0s"
shutdownGracePeriodCriticalPods meta/v1.Duration	shutdownGracePeriodCriticalPods specifies the duration used to terminate critical pods during a node shutdown. This should be less than shutdownGracePeriod. For example, if shutdownGracePeriod=30s, and shutdownGracePeriodCriticalPods=10s, during a node shutdown the first 20 seconds would be reserved for gracefully terminating normal pods, and the last 10 seconds would be reserved for terminating critical pods. Default: "0s"
shutdownGracePeriodByPodPriority []ShutdownGracePeriodByPodPriority	shutdownGracePeriodByPodPriority specifies the shutdown grace period for Pods based on their associated priority class value. When a shutdown request is received, the Kubelet will initiate shutdown on all pods running on the node with a grace period that depends on the priority of the pod, and then wait for all pods to exit. Each entry in the array represents the graceful shutdown time a pod with a priority class value that lies in the range of that value and the next higher entry in the list when the node is shutting down. For example, to allow critical pods 10s to shutdown, priority>=10000 pods 20s to shutdown, and all remaining pods 30s to shutdown. shutdownGracePeriodByPodPriority: • priority: 2000000000 shutdownGracePeriodSeconds: 10 • priority: 10000 shutdownGracePeriodSeconds: 20 • priority: 0 shutdownGracePeriodSeconds: 30 The time the Kubelet will wait before exiting will at most be the maximum of all shutdownGracePeriodSeconds for each priority class range represented on the node. When all pods have exited or reached their grace periods, the Kubelet will release the shutdown inhibit lock. Requires the GracefulNodeShutdown feature gate to be enabled. This configuration must be empty if either ShutdownGracePeriod or ShutdownGracePeriodCriticalPods is set. Default: nil
crashLoopBackOff CrashLoopBackOffConfig	CrashLoopBackOff contains config to modify node-level parameters for container restart behavior
reservedMemory []MemoryReservation	reservedMemory specifies a comma-separated list of memory reservations for NUMA nodes. The parameter makes sense only in the context of the memory manager feature. The memory manager will not allocate reserved memory for container workloads. For example, if you have a NUMA0 with 10Gi of memory and the reservedMemory was specified to reserve 1Gi of memory at NUMA0, the memory manager will assume that only 9Gi is available for allocation. You can specify a different amount of NUMA node and memory types. You can omit this parameter at all, but you should be aware that the amount of reserved memory from all NUMA nodes should be equal to the amount of memory specified by the node allocatable. If at least one node allocatable parameter has a non-zero value, you will need to specify at least one NUMA node. Also, avoid specifying: 1. Duplicates, the same NUMA node, and memory type, but with a different value.

Field	Description
	2. zero limits for any memory type. 3. NUMAs nodes IDs that do not exist under the machine. 4. memory types except for memory and hugepages- Default: nil
`enableProfilingHandler` `bool`	enableProfilingHandler enables profiling via web interface host:port/debug/pprof/ Default: true
`enableDebugFlagsHandler` `bool`	enableDebugFlagsHandler enables flags endpoint via web interface host:port/debug/flags/v Default: true
`seccompDefault` `bool`	SeccompDefault enables the use of `RuntimeDefault` as the default seccomp profile for all workloads. Default: false
`memoryThrottlingFactor` `float64`	MemoryThrottlingFactor specifies the factor multiplied by the memory limit or node allocatable memory when setting the cgroupv2 memory.high value to enforce MemoryQoS. Decreasing this factor will set lower high limit for container cgroups and put heavier reclaim pressure while increasing will put less reclaim pressure. See https://kep.k8s.io/2570 for more details. Default: 0.9
`registerWithTaints` `[]core/v1.Taint`	registerWithTaints are an array of taints to add to a node object when the kubelet registers itself. This only takes effect when registerNode is true and upon the initial registration of the node. Default: nil
`registerNode` `bool`	registerNode enables automatic registration with the apiserver. Default: true
`tracing` `TracingConfiguration`	Tracing specifies the versioned configuration for OpenTelemetry tracing clients. See https://kep.k8s.io/2832 for more details. Default: nil
`localStorageCapacityIsolation` `bool`	LocalStorageCapacityIsolation enables local ephemeral storage isolation feature. The default setting is true. This feature allows users to set request/limit for container's ephemeral storage and manage it in a similar way as cpu and memory. It also allows setting sizeLimit for emptyDir volume, which will trigger pod eviction if disk usage from the volume exceeds the limit. This feature depends on the capability of detecting correct root file system disk usage. For certain systems, such as kind rootless, if this capability cannot be supported, the feature LocalStorageCapacityIsolation should be disabled. Once disabled, user should not set request/limit for container's ephemeral storage, or sizeLimit for emptyDir. Default: true
`containerRuntimeEndpoint` **[Required]** `string`	ContainerRuntimeEndpoint is the endpoint of container runtime. Unix Domain Sockets are supported on Linux, while npipe and tcp endpoints are supported on Windows. Examples:'unix:///path/to/runtime.sock', 'npipe:////./pipe/runtime'
`imageServiceEndpoint` `string`	ImageServiceEndpoint is the endpoint of container image service. Unix Domain Socket are supported on Linux, while npipe and tcp endpoints are supported on Windows. Examples:'unix:///path/to/runtime.sock', 'npipe:////./pipe/runtime'. If not specified, the value in containerRuntimeEndpoint is used.
`failCgroupV1` `bool`	FailCgroupV1 prevents the kubelet from starting on hosts that use cgroup v1. By default, this is set to 'false', meaning the kubelet is allowed to start on cgroup v1 hosts unless this option is explicitly enabled. Default: false
`userNamespaces` `UserNamespaces`	UserNamespaces contains User Namespace configurations.

`SerializedNodeConfigSource`

SerializedNodeConfigSource allows us to serialize v1.NodeConfigSource. This type is used internally by the Kubelet for tracking checkpointed dynamic configs. It exists in the kubeletconfig API group because it is classified as a versioned input to the Kubelet.

Field	Description
`apiVersion` string	`kubelet.config.k8s.io/v1beta1`
`kind` string	`SerializedNodeConfigSource`
`source` `core/v1.NodeConfigSource`	source is the source that we are serializing.

`CrashLoopBackOffConfig`

Appears in:

- KubeletConfiguration

Field	Description
`maxContainerRestartPeriod` `meta/v1.Duration`	maxContainerRestartPeriod is the maximum duration the backoff delay can accrue to for container restarts, minimum 1 second, maximum 300 seconds. If not set, defaults to the internal

Field	Description
	crashloopbackoff maximum (300s).

`CredentialProvider`

Appears in:

- CredentialProviderConfig

CredentialProvider represents an exec plugin to be invoked by the kubelet. The plugin is only invoked when an image being pulled matches the images handled by the plugin (see matchImages).

Field	Description
`name` **[Required]** `string`	name is the required name of the credential provider. It must match the name of the provider executable as seen by the kubelet. The executable must be in the kubelet's bin directory (set by the --image-credential-provider-bin-dir flag). Required to be unique across all providers.
`matchImages` **[Required]** `[]string`	matchImages is a required list of strings used to match against images in order to determine if this provider should be invoked. If one of the strings matches the requested image from the kubelet, the plugin will be invoked and given a chance to provide credentials. Images are expected to contain the registry domain and URL path. Each entry in matchImages is a pattern which can optionally contain a port and a path. Globs can be used in the domain, but not in the port or the path. Globs are supported as subdomains like '*.k8s.io' or 'k8s.*.io', and top-level-domains such as 'k8s.*'. Matching partial subdomains like 'app*.k8s.io' is also supported. Each glob can only match a single subdomain segment, so *.io does not match *.k8s.io. A match exists between an image and a matchImage when all of the below are true: • Both contain the same number of domain parts and each part matches. • The URL path of an imageMatch must be a prefix of the target image URL path. • If the imageMatch contains a port, then the port must match in the image as well. Example values of matchImages: • 123456789.dkr.ecr.us-east-1.amazonaws.com • *.azurecr.io • gcr.io • *.registry.io • registry.io:8080/path
`defaultCacheDuration` **[Required]** meta/v1.Duration	defaultCacheDuration is the default duration the plugin will cache credentials in-memory if a cache duration is not provided in the plugin response. This field is required.
`apiVersion` **[Required]** `string`	Required input version of the exec CredentialProviderRequest. The returned CredentialProviderResponse MUST use the same encoding version as the input. Current supported values are: • credentialprovider.kubelet.k8s.io/v1beta1
`args` `[]string`	Arguments to pass to the command when executing it.
`env` []ExecEnvVar	Env defines additional environment variables to expose to the process. These are unioned with the host's environment, as well as variables client-go uses to pass argument to the plugin.

`ExecEnvVar`

Appears in:

- CredentialProvider

ExecEnvVar is used for setting environment variables when executing an exec-based credential plugin.

Field	Description
`name` **[Required]** `string`	No description provided.
`value` **[Required]** `string`	No description provided.

`ImagePullCredentialsVerificationPolicy`

(Alias of `string`)

Appears in:

- KubeletConfiguration

ImagePullCredentialsVerificationPolicy is an enum for the policy that is enforced when pod is requesting an image that appears on the system

KubeletAnonymousAuthentication

Appears in:

- KubeletAuthentication

Field	Description
`enabled` `bool`	enabled allows anonymous requests to the kubelet server. Requests that are not rejected by another authentication method are treated as anonymous requests. Anonymous requests have a username of `system:anonymous`, and a group name of `system:unauthenticated`.

KubeletAuthentication

Appears in:

- KubeletConfiguration

Field	Description
`x509` `KubeletX509Authentication`	x509 contains settings related to x509 client certificate authentication.
`webhook` `KubeletWebhookAuthentication`	webhook contains settings related to webhook bearer token authentication.
`anonymous` `KubeletAnonymousAuthentication`	anonymous contains settings related to anonymous authentication.

KubeletAuthorization

Appears in:

- KubeletConfiguration

Field	Description
`mode` `KubeletAuthorizationMode`	mode is the authorization mode to apply to requests to the kubelet server. Valid values are `AlwaysAllow` and `Webhook`. Webhook mode uses the SubjectAccessReview API to determine authorization.
`webhook` `KubeletWebhookAuthorization`	webhook contains settings related to Webhook authorization.

KubeletAuthorizationMode

(Alias of `string`)

Appears in:

- KubeletAuthorization

KubeletWebhookAuthentication

Appears in:

- KubeletAuthentication

Field	Description
`enabled` `bool`	enabled allows bearer token authentication backed by the tokenreviews.authentication.k8s.io API.
`cacheTTL` `meta/v1.Duration`	cacheTTL enables caching of authentication results

KubeletWebhookAuthorization

Appears in:

- KubeletAuthorization

Field	Description
`cacheAuthorizedTTL` meta/v1.Duration	cacheAuthorizedTTL is the duration to cache 'authorized' responses from the webhook authorizer.
`cacheUnauthorizedTTL` meta/v1.Duration	cacheUnauthorizedTTL is the duration to cache 'unauthorized' responses from the webhook authorizer.

KubeletX509Authentication

Appears in:

- KubeletAuthentication

Field	Description
`clientCAFile` string	clientCAFile is the path to a PEM-encoded certificate bundle. If set, any request presenting a client certificate signed by one of the authorities in the bundle is authenticated with a username corresponding to the CommonName, and groups corresponding to the Organization in the client certificate.

MemoryReservation

Appears in:

- KubeletConfiguration

MemoryReservation specifies the memory reservation of different types for each NUMA node

Field	Description
`numaNode` **[Required]** int32	No description provided.
`limits` **[Required]** core/v1.ResourceList	No description provided.

MemorySwapConfiguration

Appears in:

- KubeletConfiguration

Field	Description
`swapBehavior` string	swapBehavior configures swap memory available to container workloads. May be one of "", "NoSwap": workloads can not use swap, default option. "LimitedSwap": workload swap usage is limited. The swap limit is proportionate to the container's memory request.

ResourceChangeDetectionStrategy

(Alias of `string`)

Appears in:

- KubeletConfiguration

ResourceChangeDetectionStrategy denotes a mode in which internal managers (secret, configmap) are discovering object changes.

ShutdownGracePeriodByPodPriority

Appears in:

- KubeletConfiguration

ShutdownGracePeriodByPodPriority specifies the shutdown grace period for Pods based on their associated priority class value

Field	Description
`priority` **[Required]** int32	priority is the priority value associated with the shutdown grace period

Field	Description

`shutdownGracePeriodSeconds`
[Required]
`int64`
	shutdownGracePeriodSeconds is the shutdown grace period in seconds

`UserNamespaces`

Appears in:

- KubeletConfiguration

UserNamespaces contains User Namespace configurations.

Field	Description
`idsPerPod` `int64`	IDsPerPod is the mapping length of UIDs and GIDs. The length must be a multiple of 65536, and must be less than 1<<32. On non-linux such as windows, only null / absent is allowed. Changing the value may require recreating all containers on the node. Default: 65536

Component tools

Feature Gates

Feature Gates (removed)

kube-apiserver

kube-controller-manager

kube-proxy

kube-scheduler

kubelet

---

kube-apiserver Admission (v1)

Resource Types

- AdmissionReview

`AdmissionReview`

AdmissionReview describes an admission review request/response.

Field	Description
`apiVersion` string	`admission.k8s.io/v1`
`kind` string	`AdmissionReview`
`request` `AdmissionRequest`	Request describes the attributes for the admission request.
`response` `AdmissionResponse`	Response describes the attributes for the admission response.

`AdmissionRequest`

Appears in:

- AdmissionReview

AdmissionRequest describes the admission.Attributes for the admission request.

Field	Description
`uid` **[Required]** [k8s.io/apimachinery/pkg/types.UID](#)	UID is an identifier for the individual request/response. It allows us to distinguish instances of requests which are otherwise identical (parallel requests, requests when earlier requests did not modify etc) The UID is meant to track the round trip (request/response) between the KAS and the WebHook, not the user request. It is suitable for correlating log entries between the webhook and apiserver, for either auditing or debugging.
`kind` **[Required]** [meta/v1.GroupVersionKind](#)	Kind is the fully-qualified type of object being submitted (for example, v1.Pod or autoscaling.v1.Scale)
`resource` **[Required]** [meta/v1.GroupVersionResource](#)	Resource is the fully-qualified resource being requested (for example, v1.pods)
`subResource` `string`	SubResource is the subresource being requested, if any (for example, "status" or "scale")
`requestKind` [meta/v1.GroupVersionKind](#)	RequestKind is the fully-qualified type of the original API request (for example, v1.Pod or autoscaling.v1.Scale). If this is specified and differs from the value in "kind", an equivalent match and conversion was performed. For example, if deployments can be modified via apps/v1 and apps/v1beta1, and a webhook registered a rule of `apiGroups:["apps"], apiVersions:["v1"], resources: ["deployments"]` and `matchPolicy: Equivalent`, an API request to apps/v1beta1 deployments would be converted and sent to the webhook with `kind: {group:"apps", version:"v1", kind:"Deployment"}` (matching the rule the webhook registered for), and `requestKind: {group:"apps", version:"v1beta1", kind:"Deployment"}` (indicating the kind of the original API request). See documentation for the "matchPolicy" field in the webhook configuration type for more details.
`requestResource` [meta/v1.GroupVersionResource](#)	RequestResource is the fully-qualified resource of the original API request (for example, v1.pods). If this is specified and differs from the value in "resource", an equivalent match and conversion was performed. For example, if deployments can be modified via apps/v1 and apps/v1beta1, and a webhook registered a rule of `apiGroups:["apps"], apiVersions:["v1"], resources: ["deployments"]` and `matchPolicy: Equivalent`, an API request to apps/v1beta1 deployments would be converted and sent to the webhook with `resource: {group:"apps", version:"v1", resource:"deployments"}` (matching the resource the webhook registered for), and `requestResource: {group:"apps", version:"v1beta1", resource:"deployments"}` (indicating the resource of the original API request). See documentation for the "matchPolicy" field in the webhook configuration type.
`requestSubResource` `string`	RequestSubResource is the name of the subresource of the original API request, if any (for example, "status" or "scale") If this is specified and differs from the value in "subResource", an equivalent match and conversion was performed. See documentation for the "matchPolicy" field in the webhook configuration type.
`name` `string`	Name is the name of the object as presented in the request. On a CREATE operation, the client may omit name and rely on the server to generate the name. If that is the case, this field will contain an empty string.
`namespace` `string`	Namespace is the namespace associated with the request (if any).
`operation` **[Required]** [Operation](#)	Operation is the operation being performed. This may be different than the operation requested. e.g. a patch can result in either a CREATE or UPDATE Operation.
`userInfo` **[Required]** [authentication/v1.UserInfo](#)	UserInfo is information about the requesting user
`object` [k8s.io/apimachinery/pkg/runtime.RawExtension](#)	Object is the object from the incoming request.
`oldObject` [k8s.io/apimachinery/pkg/runtime.RawExtension](#)	OldObject is the existing object. Only populated for DELETE and UPDATE requests.
`dryRun` `bool`	DryRun indicates that modifications will definitely not be persisted for this request. Defaults to false.
`options` [k8s.io/apimachinery/pkg/runtime.RawExtension](#)	Options is the operation option structure of the operation being performed. e.g. `meta.k8s.io/v1.DeleteOptions` or `meta.k8s.io/v1.CreateOptions`. This may be different than the options the caller provided. e.g. for a patch request the performed Operation might be a CREATE, in which case the Options will a `meta.k8s.io/v1.CreateOptions` even though the caller provided `meta.k8s.io/v1.PatchOptions`.

AdmissionResponse

Appears in:

- AdmissionReview

AdmissionResponse describes an admission response.

Field	Description
`uid` **[Required]** k8s.io/apimachinery/pkg/types.UID	UID is an identifier for the individual request/response. This must be copied over from the corresponding AdmissionRequest.
`allowed` **[Required]** bool	Allowed indicates whether or not the admission request was permitted.
`status` meta/v1.Status	Result contains extra details into why an admission request was denied. This field IS NOT consulted in any way if "Allowed" is "true".
`patch` []byte	The patch body. Currently we only support "JSONPatch" which implements RFC 6902.
`patchType` PatchType	The type of Patch. Currently we only allow "JSONPatch".
`auditAnnotations` map[string]string	AuditAnnotations is an unstructured key value map set by remote admission controller (e.g. error=image-blacklisted). MutatingAdmissionWebhook and ValidatingAdmissionWebhook admission controller will prefix the keys with admission webhook name (e.g. imagepolicy.example.com/error=image-blacklisted). AuditAnnotations will be provided by the admission webhook to add additional context to the audit log for this request.
`warnings` []string	warnings is a list of warning messages to return to the requesting API client. Warning messages describe a problem the client making the API request should correct or be aware of. Limit warnings to 120 characters if possible. Warnings over 256 characters and large numbers of warnings may be truncated.

`Operation`

(Alias of `string`)

Appears in:

- AdmissionRequest

Operation is the type of resource operation being checked for admission control

`PatchType`

(Alias of `string`)

Appears in:

- AdmissionResponse

PatchType is the type of patch being used to represent the mutated object

---

kube-apiserver Audit Configuration (v1)

Resource Types

- Event
- EventList
- Policy
- PolicyList

`Event`

Appears in:

- EventList

Event captures all the information that can be included in an API audit log.

Field	Description
`apiVersion` string	`audit.k8s.io/v1`

Field	Description
`kind` string	`Event`
`level` **[Required]** `Level`	AuditLevel at which event was generated
`auditID` **[Required]** `k8s.io/apimachinery/pkg/types.UID`	Unique audit ID, generated for each request.
`stage` **[Required]** `Stage`	Stage of the request handling when this event instance was generated.
`requestURI` **[Required]** `string`	RequestURI is the request URI as sent by the client to a server.
`verb` **[Required]** `string`	Verb is the kubernetes verb associated with the request. For non-resource requests, this is the lower-cased HTTP method.
`user` **[Required]** `authentication/v1.UserInfo`	Authenticated user information.
`impersonatedUser` `authentication/v1.UserInfo`	Impersonated user information.
`sourceIPs` `[]string`	Source IPs, from where the request originated and intermediate proxies. The source IPs are listed from (in order): 1. X-Forwarded-For request header IPs 2. X-Real-Ip header, if not present in the X-Forwarded-For list 3. The remote address for the connection, if it doesn't match the last IP in the list up to here (X-Forwarded-For or X-Real-Ip). Note: All but the last IP can be arbitrarily set by the client.
`userAgent` `string`	UserAgent records the user agent string reported by the client. Note that the UserAgent is provided by the client, and must not be trusted.
`objectRef` `ObjectReference`	Object reference this request is targeted at. Does not apply for List-type requests, or non-resource requests.
`responseStatus` `meta/v1.Status`	The response status, populated even when the ResponseObject is not a Status type. For successful responses, this will only include the Code and StatusSuccess. For non-status type error responses, this will be auto-populated with the error Message.
`requestObject` `k8s.io/apimachinery/pkg/runtime.Unknown`	API object from the request, in JSON format. The RequestObject is recorded as-is in the request (possibly re-encoded as JSON), prior to version conversion, defaulting, admission or merging. It is an external versioned object type, and may not be a valid object on its own. Omitted for non-resource requests. Only logged at Request Level and higher.
`responseObject` `k8s.io/apimachinery/pkg/runtime.Unknown`	API object returned in the response, in JSON. The ResponseObject is recorded after conversion to the external type, and serialized as JSON. Omitted for non-resource requests. Only logged at Response Level.
`requestReceivedTimestamp` `meta/v1.MicroTime`	Time the request reached the apiserver.
`stageTimestamp` `meta/v1.MicroTime`	Time the request reached current audit stage.
`annotations` `map[string]string`	Annotations is an unstructured key value map stored with an audit event that may be set by plugins invoked in the request serving chain, including authentication, authorization and admission plugins. Note that these annotations are for the audit event, and do not correspond to the metadata.annotations of the submitted object. Keys should uniquely identify the informing component to avoid name collisions (e.g. podsecuritypolicy.admission.k8s.io/policy). Values should be short. Annotations are included in the Metadata level.

`EventList`

EventList is a list of audit Events.

Field	Description
`apiVersion` string	`audit.k8s.io/v1`
`kind` string	`EventList`
`metadata` `meta/v1.ListMeta`	No description provided.
`items` **[Required]** `[]Event`	No description provided.

`Policy`

Appears in:

- PolicyList

Policy defines the configuration of audit logging, and the rules for how different request categories are logged.

Field	Description
`apiVersion` string	`audit.k8s.io/v1`
`kind` string	`Policy`
`metadata` meta/v1.ObjectMeta	ObjectMeta is included for interoperability with API infrastructure. Refer to the Kubernetes API documentation for the fields of the `metadata` field.
`rules` **[Required]** []PolicyRule	Rules specify the audit Level a request should be recorded at. A request may match multiple rules, in which case the FIRST matching rule is used. The default audit level is None, but can be overridden by a catch-all rule at the end of the list. PolicyRules are strictly ordered.
`omitStages` []Stage	OmitStages is a list of stages for which no events are created. Note that this can also be specified per rule in which case the union of both are omitted.
`omitManagedFields` bool	OmitManagedFields indicates whether to omit the managed fields of the request and response bodies from being written to the API audit log. This is used as a global default - a value of 'true' will omit the managed fields, otherwise the managed fields will be included in the API audit log. Note that this can also be specified per rule in which case the value specified in a rule will override the global default.

`PolicyList`

PolicyList is a list of audit Policies.

Field	Description
`apiVersion` string	`audit.k8s.io/v1`
`kind` string	`PolicyList`
`metadata` meta/v1.ListMeta	No description provided.
`items` **[Required]** []Policy	No description provided.

`GroupResources`

Appears in:

- PolicyRule

GroupResources represents resource kinds in an API group.

Field	Description
`group` string	Group is the name of the API group that contains the resources. The empty string represents the core API group.
`resources` []string	Resources is a list of resources this rule applies to. For example: - `pods` matches pods. - `pods/log` matches the log subresource of pods. - `*` matches all resources and their subresources. - `pods/*` matches all subresources of pods. - `*/scale` matches all scale subresources. If wildcard is present, the validation rule will ensure resources do not overlap with each other. An empty list implies all resources and subresources in this API groups apply.
`resourceNames` []string	ResourceNames is a list of resource instance names that the policy matches. Using this field requires Resources to be specified. An empty list implies that every instance of the resource is matched.

`Level`

(Alias of `string`)

Appears in:

- Event
- PolicyRule

Level defines the amount of information logged during auditing

`ObjectReference`

Appears in:

- Event

ObjectReference contains enough information to let you inspect or modify the referred object.

Field	Description
`resource` `string`	No description provided.
`namespace` `string`	No description provided.
`name` `string`	No description provided.
`uid` `k8s.io/apimachinery/pkg/types.UID`	No description provided.
`apiGroup` `string`	APIGroup is the name of the API group that contains the referred object. The empty string represents the core API group.
`apiVersion` `string`	APIVersion is the version of the API group that contains the referred object.
`resourceVersion` `string`	No description provided.
`subresource` `string`	No description provided.

`PolicyRule`

Appears in:

- Policy

PolicyRule maps requests based off metadata to an audit Level. Requests must match the rules of every field (an intersection of rules).

Field	Description
`level` **[Required]** `Level`	The Level that requests matching this rule are recorded at.
`users` `[]string`	The users (by authenticated user name) this rule applies to. An empty list implies every user.
`userGroups` `[]string`	The user groups this rule applies to. A user is considered matching if it is a member of any of the UserGroups. An empty list implies every user group.
`verbs` `[]string`	The verbs that match this rule. An empty list implies every verb.
`resources` `[]GroupResources`	Resources that this rule matches. An empty list implies all kinds in all API groups.
`namespaces` `[]string`	Namespaces that this rule matches. The empty string "" matches non-namespaced resources. An empty list implies every namespace.
`nonResourceURLs` `[]string`	NonResourceURLs is a set of URL paths that should be audited. `*`s are allowed, but only as the full, final step in the path. Examples: • `/metrics` - Log requests for apiserver metrics • `/healthz*` - Log all health checks
`omitStages` `[]Stage`	OmitStages is a list of stages for which no events are created. Note that this can also be specified policy wide in which case the union of both are omitted. An empty list means no restrictions will apply.
`omitManagedFields` `bool`	OmitManagedFields indicates whether to omit the managed fields of the request and response bodies from being written to the API audit log. • a value of 'true' will drop the managed fields from the API audit log • a value of 'false' indicates that the managed fields should be included in the API audit log Note that the value, if specified, in this rule will override the global default If a value is not specified then the global default specified in Policy.OmitManagedFields will stand.

`Stage`

(Alias of `string`)

Appears in:

- Event
- Policy
- PolicyRule

Stage defines the stages in request handling that audit events may be generated.

---

kube-scheduler

Synopsis

The Kubernetes scheduler is a control plane process which assigns Pods to Nodes. The scheduler determines which Nodes are valid placements for each Pod in the scheduling queue according to constraints and available resources. The scheduler then ranks each valid Node and binds the Pod to a suitable Node. Multiple different schedulers may be used within a cluster; kube-scheduler is the reference implementation. See scheduling for more information about scheduling and the kube-scheduler component.

```
kube-scheduler [flags]
```

Options

--allow-metric-labels stringToString Default: []
The map from metric-label to value allow-list of this label. The key's format is <MetricName>,<LabelName>. The value's format is <allowed_value>,<allowed_value>...e.g. metric1,label1='v1,v2,v3', metric1,label2='v1,v2,v3' metric2,label1='v1,v2,v3'.

--allow-metric-labels-manifest string
The path to the manifest file that contains the allow-list mapping. The format of the file is the same as the flag --allow-metric-labels. Note that the flag --allow-metric-labels will override the manifest file.

--authentication-kubeconfig string
kubeconfig file pointing at the 'core' kubernetes server with enough rights to create tokenreviews.authentication.k8s.io. This is optional. If empty, all token requests are considered to be anonymous and no client CA is looked up in the cluster.

--authentication-skip-lookup
If false, the authentication-kubeconfig will be used to lookup missing authentication configuration from the cluster.

--authentication-token-webhook-cache-ttl duration Default: 10s
The duration to cache responses from the webhook token authenticator.

--authentication-tolerate-lookup-failure Default: true
If true, failures to look up missing authentication configuration from the cluster are not considered fatal. Note that this can result in authentication that treats all requests as anonymous.

--authorization-always-allow-paths strings Default: "/healthz,/readyz,/livez"
A list of HTTP paths to skip during authorization, i.e. these are authorized without contacting the 'core' kubernetes server.

--authorization-kubeconfig string
kubeconfig file pointing at the 'core' kubernetes server with enough rights to create subjectaccessreviews.authorization.k8s.io. This is optional. If empty, all requests not skipped by authorization are forbidden.

--authorization-webhook-cache-authorized-ttl duration Default: 10s
The duration to cache 'authorized' responses from the webhook authorizer.

--authorization-webhook-cache-unauthorized-ttl duration Default: 10s
The duration to cache 'unauthorized' responses from the webhook authorizer.

--bind-address string Default: 0.0.0.0
The IP address on which to listen for the --secure-port port. The associated interface(s) must be reachable by the rest of the cluster, and by CLI/web clients. If blank or an unspecified address (0.0.0.0 or ::), all interfaces and IP address families will be used.

--cert-dir string
The directory where the TLS certs are located. If --tls-cert-file and --tls-private-key-file are provided, this flag will be ignored.

--client-ca-file string
If set, any request presenting a client certificate signed by one of the authorities in the client-ca-file is authenticated with an identity corresponding to the CommonName of the client certificate.

--config string

The path to the configuration file.

--contention-profiling Default: true

DEPRECATED: enable block profiling, if profiling is enabled. This parameter is ignored if a config file is specified in --config.

--disable-http2-serving

If true, HTTP2 serving will be disabled [default=false]

--disabled-metrics strings

This flag provides an escape hatch for misbehaving metrics. You must provide the fully qualified metric name in order to disable it.
Disclaimer: disabling metrics is higher in precedence than showing hidden metrics.

--emulated-version strings

The versions different components emulate their capabilities (APIs, features, ...) of.
If set, the component will emulate the behavior of this version instead of the underlying binary version.
Version format could only be major.minor, for example: '--emulated-version=wardle=1.2,kube=1.31'.
Options are: kube=1.31..1.34(default:1.34)
If the component is not specified, defaults to "kube"

--feature-gates colonSeparatedMultimapStringString

Comma-separated list of component:key=value pairs that describe feature gates for alpha/experimental features of different components.
If the component is not specified, defaults to "kube". This flag can be repeatedly invoked. For example: --feature-gates
'wardle:featureA=true,wardle:featureB=false' --feature-gates 'kube:featureC=true'Options are:
kube:APIResponseCompression=true|false (BETA - default=true)
kube:APIServerIdentity=true|false (BETA - default=true)
kube:APIServingWithRoutine=true|false (ALPHA - default=false)
kube:AllAlpha=true|false (ALPHA - default=false)
kube:AllBeta=true|false (BETA - default=false)
kube:AllowParsingUserUIDFromCertAuth=true|false (BETA - default=true)
kube:AllowUnsafeMalformedObjectDeletion=true|false (ALPHA - default=false)
kube:CBORServingAndStorage=true|false (ALPHA - default=false)
kube:CPUManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
kube:CPUManagerPolicyBetaOptions=true|false (BETA - default=true)
kube:CSIVolumeHealth=true|false (ALPHA - default=false)
kube:ClearingNominatedNodeNameAfterBinding=true|false (ALPHA - default=false)
kube:ClientsAllowCBOR=true|false (ALPHA - default=false)
kube:ClientsPreferCBOR=true|false (ALPHA - default=false)
kube:CloudControllerManagerWebhook=true|false (ALPHA - default=false)
kube:ClusterTrustBundle=true|false (BETA - default=false)
kube:ClusterTrustBundleProjection=true|false (BETA - default=false)
kube:ComponentFlagz=true|false (ALPHA - default=false)
kube:ComponentStatusz=true|false (ALPHA - default=false)
kube:ConcurrentWatchObjectDecode=true|false (BETA - default=false)
kube:ContainerCheckpoint=true|false (BETA - default=true)
kube:ContainerRestartRules=true|false (ALPHA - default=false)
kube:ContainerStopSignals=true|false (ALPHA - default=false)
kube:ContextualLogging=true|false (BETA - default=true)
kube:CoordinatedLeaderElection=true|false (BETA - default=false)
kube:CrossNamespaceVolumeDataSource=true|false (ALPHA - default=false)
kube:CustomCPUCFSQuotaPeriod=true|false (ALPHA - default=false)
kube:DRAAdminAccess=true|false (BETA - default=true)
kube:DRAConsumableCapacity=true|false (ALPHA - default=false)
kube:DRADeviceBindingConditions=true|false (ALPHA - default=false)
kube:DRADeviceTaints=true|false (ALPHA - default=false)
kube:DRAExtendedResource=true|false (ALPHA - default=false)
kube:DRAPartitionableDevices=true|false (ALPHA - default=false)
kube:DRAPrioritizedList=true|false (BETA - default=true)
kube:DRAResourceClaimDeviceStatus=true|false (BETA - default=true)
kube:DRASchedulerFilterTimeout=true|false (BETA - default=true)
kube:DeclarativeValidation=true|false (BETA - default=true)
kube:DeclarativeValidationTakeover=true|false (BETA - default=false)
kube:DeploymentReplicaSetTerminatingReplicas=true|false (ALPHA - default=false)
kube:DetectCacheInconsistency=true|false (BETA - default=true)
kube:DisableCPUQuotaWithExclusiveCPUs=true|false (BETA - default=true)
kube:EnvFiles=true|false (ALPHA - default=false)

kube:EventedPLEG=true|false (ALPHA - default=false)
kube:ExternalServiceAccountTokenSigner=true|false (BETA - default=true)
kube:GracefulNodeShutdown=true|false (BETA - default=true)
kube:GracefulNodeShutdownBasedOnPodPriority=true|false (BETA - default=true)
kube:HPAConfigurableTolerance=true|false (ALPHA - default=false)
kube:HPAScaleToZero=true|false (ALPHA - default=false)
kube:HostnameOverride=true|false (ALPHA - default=false)
kube:ImageMaximumGCAge=true|false (BETA - default=true)
kube:ImageVolume=true|false (BETA - default=false)
kube:InOrderInformers=true|false (BETA - default=true)
kube:InPlacePodVerticalScaling=true|false (BETA - default=true)
kube:InPlacePodVerticalScalingExclusiveCPUs=true|false (ALPHA - default=false)
kube:InPlacePodVerticalScalingExclusiveMemory=true|false (ALPHA - default=false)
kube:InTreePluginPortworxUnregister=true|false (ALPHA - default=false)
kube:InformerResourceVersion=true|false (ALPHA - default=false)
kube:JobManagedBy=true|false (BETA - default=true)
kube:KubeletCrashLoopBackOffMax=true|false (ALPHA - default=false)
kube:KubeletEnsureSecretPulledImages=true|false (ALPHA - default=false)
kube:KubeletFineGrainedAuthz=true|false (BETA - default=true)
kube:KubeletInUserNamespace=true|false (ALPHA - default=false)
kube:KubeletPSI=true|false (BETA - default=true)
kube:KubeletPodResourcesDynamicResources=true|false (BETA - default=true)
kube:KubeletPodResourcesGet=true|false (BETA - default=true)
kube:KubeletSeparateDiskGC=true|false (BETA - default=true)
kube:KubeletServiceAccountTokenForCredentialProviders=true|false (BETA - default=true)
kube:ListFromCacheSnapshot=true|false (BETA - default=true)
kube:LocalStorageCapacityIsolationFSQuotaMonitoring=true|false (BETA - default=false)
kube:LoggingAlphaOptions=true|false (ALPHA - default=false)
kube:LoggingBetaOptions=true|false (BETA - default=true)
kube:MatchLabelKeysInPodTopologySpread=true|false (BETA - default=true)
kube:MatchLabelKeysInPodTopologySpreadSelectorMerge=true|false (BETA - default=true)
kube:MaxUnavailableStatefulSet=true|false (ALPHA - default=false)
kube:MemoryQoS=true|false (ALPHA - default=false)
kube:MutableCSINodeAllocatableCount=true|false (BETA - default=false)
kube:MutatingAdmissionPolicy=true|false (BETA - default=false)
kube:NodeLogQuery=true|false (BETA - default=false)
kube:NominatedNodeNameForExpectation=true|false (ALPHA - default=false)
kube:OpenAPIEnums=true|false (BETA - default=true)
kube:PodAndContainerStatsFromCRI=true|false (ALPHA - default=false)
kube:PodCertificateRequest=true|false (ALPHA - default=false)
kube:PodDeletionCost=true|false (BETA - default=true)
kube:PodLevelResources=true|false (BETA - default=true)
kube:PodLogsQuerySplitStreams=true|false (ALPHA - default=false)
kube:PodObservedGenerationTracking=true|false (BETA - default=true)
kube:PodReadyToStartContainersCondition=true|false (BETA - default=true)
kube:PodTopologyLabelsAdmission=true|false (ALPHA - default=false)
kube:PortForwardWebsockets=true|false (BETA - default=true)
kube:PreferSameTrafficDistribution=true|false (BETA - default=true)
kube:PreventStaticPodAPIReferences=true|false (BETA - default=true)
kube:ProcMountType=true|false (BETA - default=true)
kube:QOSReserved=true|false (ALPHA - default=false)
kube:ReduceDefaultCrashLoopBackOffDecay=true|false (ALPHA - default=false)
kube:RelaxedServiceNameValidation=true|false (ALPHA - default=false)
kube:ReloadKubeletServerCertificateFile=true|false (BETA - default=true)
kube:RemoteRequestHeaderUID=true|false (BETA - default=true)
kube:ResourceHealthStatus=true|false (ALPHA - default=false)
kube:RotateKubeletServerCertificate=true|false (BETA - default=true)
kube:RuntimeClassInImageCriApi=true|false (ALPHA - default=false)
kube:SELinuxChangePolicy=true|false (BETA - default=true)
kube:SELinuxMount=true|false (BETA - default=false)
kube:SELinuxMountReadWriteOncePod=true|false (BETA - default=true)
kube:SchedulerAsyncAPICalls=true|false (BETA - default=true)

kube:SchedulerAsyncPreemption=true|false (BETA - default=true)
kube:SchedulerPopFromBackoffQ=true|false (BETA - default=true)
kube:ServiceAccountNodeAudienceRestriction=true|false (BETA - default=true)
kube:SizeBasedListCostEstimate=true|false (BETA - default=true)
kube:StorageCapacityScoring=true|false (ALPHA - default=false)
kube:StorageVersionAPI=true|false (ALPHA - default=false)
kube:StorageVersionHash=true|false (BETA - default=true)
kube:StorageVersionMigrator=true|false (ALPHA - default=false)
kube:StrictIPCIDRValidation=true|false (ALPHA - default=false)
kube:StructuredAuthenticationConfigurationEgressSelector=true|false (BETA - default=true)
kube:SupplementalGroupsPolicy=true|false (BETA - default=true)
kube:SystemdWatchdog=true|false (BETA - default=true)
kube:TokenRequestServiceAccountUIDValidation=true|false (BETA - default=true)
kube:TopologyManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
kube:TopologyManagerPolicyBetaOptions=true|false (BETA - default=true)
kube:TranslateStreamCloseWebsocketRequests=true|false (BETA - default=true)
kube:UnauthenticatedHTTP2DOSMitigation=true|false (BETA - default=true)
kube:UnknownVersionInteroperabilityProxy=true|false (ALPHA - default=false)
kube:UserNamespacesPodSecurityStandards=true|false (ALPHA - default=false)
kube:UserNamespacesSupport=true|false (BETA - default=true)
kube:WatchCacheInitializationPostStartHook=true|false (BETA - default=false)
kube:WatchList=true|false (BETA - default=true)
kube:WatchListClient=true|false (BETA - default=false)
kube:WindowsCPUAndMemoryAffinity=true|false (ALPHA - default=false)
kube:WindowsGracefulNodeShutdown=true|false (BETA - default=true)

-h, --help

help for kube-scheduler

--http2-max-streams-per-connection int

The limit that the server gives to clients for the maximum number of streams in an HTTP/2 connection. Zero means to use golang's default.

--kube-api-burst int32 Default: 100

DEPRECATED: burst to use while talking with kubernetes apiserver. This parameter is ignored if a config file is specified in --config.

--kube-api-content-type string Default: "application/vnd.kubernetes.protobuf"

DEPRECATED: content type of requests sent to apiserver. This parameter is ignored if a config file is specified in --config.

--kube-api-qps float Default: 50

DEPRECATED: QPS to use while talking with kubernetes apiserver. This parameter is ignored if a config file is specified in --config.

--kubeconfig string

DEPRECATED: path to kubeconfig file with authorization and master location information. This parameter is ignored if a config file is specified in --config.

--leader-elect Default: true

Start a leader election client and gain leadership before executing the main loop. Enable this when running replicated components for high availability.

--leader-elect-lease-duration duration Default: 15s

The duration that non-leader candidates will wait after observing a leadership renewal until attempting to acquire leadership of a led but unrenewed leader slot. This is effectively the maximum duration that a leader can be stopped before it is replaced by another candidate. This is only applicable if leader election is enabled.

--leader-elect-renew-deadline duration Default: 10s

The interval between attempts by the acting master to renew a leadership slot before it stops leading. This must be less than the lease duration. This is only applicable if leader election is enabled.

--leader-elect-resource-lock string Default: "leases"

The type of resource object that is used for locking during leader election. Supported options are 'leases'.

--leader-elect-resource-name string Default: "kube-scheduler"

The name of resource object that is used for locking during leader election.

--leader-elect-resource-namespace string Default: "kube-system"

The namespace of resource object that is used for locking during leader election.

--leader-elect-retry-period duration Default: 2s

The duration the clients should wait between attempting acquisition and renewal of a leadership. This is only applicable if leader election is enabled.

--log-flush-frequency duration Default: 5s

Maximum number of seconds between log flushes

--log-text-info-buffer-size quantity

[Alpha] In text format with split output streams, the info messages can be buffered for a while to increase performance. The default value of zero bytes disables buffering. The size can be specified as number of bytes (512), multiples of 1000 (1K), multiples of 1024 (2Ki), or powers of those (3M, 4G, 5Mi, 6Gi). Enable the LoggingAlphaOptions feature gate to use this.

--log-text-split-stream

[Alpha] In text format, write error messages to stderr and info messages to stdout. The default is to write a single stream to stdout. Enable the LoggingAlphaOptions feature gate to use this.

--logging-format string Default: "text"

Sets the log format. Permitted formats: "text".

--master string

The address of the Kubernetes API server (overrides any value in kubeconfig)

--permit-address-sharing

If true, SO_REUSEADDR will be used when binding the port. This allows binding to wildcard IPs like 0.0.0.0 and specific IPs in parallel, and it avoids waiting for the kernel to release sockets in TIME_WAIT state. [default=false]

--permit-port-sharing

If true, SO_REUSEPORT will be used when binding the port, which allows more than one instance to bind on the same address and port. [default=false]

--pod-max-in-unschedulable-pods-duration duration Default: 5m0s

DEPRECATED: the maximum time a pod can stay in unschedulablePods. If a pod stays in unschedulablePods for longer than this value, the pod will be moved from unschedulablePods to backoffQ or activeQ. This flag is deprecated and will be removed in a future version.

--profiling Default: true

DEPRECATED: enable profiling via web interface host:port/debug/pprof/. This parameter is ignored if a config file is specified in --config.

--requestheader-allowed-names strings

List of client certificate common names to allow to provide usernames in headers specified by --requestheader-username-headers. If empty, any client certificate validated by the authorities in --requestheader-client-ca-file is allowed.

--requestheader-client-ca-file string

Root certificate bundle to use to verify client certificates on incoming requests before trusting usernames in headers specified by --requestheader-username-headers. WARNING: generally do not depend on authorization being already done for incoming requests.

--requestheader-extra-headers-prefix strings Default: "x-remote-extra-"

List of request header prefixes to inspect. X-Remote-Extra- is suggested.

--requestheader-group-headers strings Default: "x-remote-group"

List of request headers to inspect for groups. X-Remote-Group is suggested.

--requestheader-uid-headers strings

List of request headers to inspect for UIDs. X-Remote-Uid is suggested. Requires the RemoteRequestHeaderUID feature to be enabled.

--requestheader-username-headers strings Default: "x-remote-user"

List of request headers to inspect for usernames. X-Remote-User is common.

--secure-port int Default: 10259

The port on which to serve HTTPS with authentication and authorization. If 0, don't serve HTTPS at all.

--show-hidden-metrics-for-version string

The previous version for which you want to show hidden metrics. Only the previous minor version is meaningful, other values will not be allowed. The format is <major>.<minor>, e.g.: '1.16'. The purpose of this format is make sure you have the opportunity to notice if the next release hides additional metrics, rather than being surprised when they are permanently removed in the release after that.

--tls-cert-file string

File containing the default x509 Certificate for HTTPS. (CA cert, if any, concatenated after server cert). If HTTPS serving is enabled, and --tls-cert-file and --tls-private-key-file are not provided, a self-signed certificate and key are generated for the public address and saved to the directory specified by --cert-dir.

--tls-cipher-suites strings

Comma-separated list of cipher suites for the server. If omitted, the default Go cipher suites will be used.
Preferred values: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256.
Insecure values: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_ECDSA_WITH_RC4_128_SHA,

TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_RSA_WITH_RC4_128_SHA, TLS_RSA_WITH_3DES_EDE_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_256_CBC_SHA, TLS_RSA_WITH_AES_256_GCM_SHA384, TLS_RSA_WITH_RC4_128_SHA.

--tls-min-version string

Minimum TLS version supported. Possible values: VersionTLS10, VersionTLS11, VersionTLS12, VersionTLS13

--tls-private-key-file string

File containing the default x509 private key matching --tls-cert-file.

--tls-sni-cert-key string

A pair of x509 certificate and private key file paths, optionally suffixed with a list of domain patterns which are fully qualified domain names, possibly with prefixed wildcard segments. The domain patterns also allow IP addresses, but IPs should only be used if the apiserver has visibility to the IP address requested by a client. If no domain patterns are provided, the names of the certificate are extracted. Non-wildcard matches trump over wildcard matches, explicit domain patterns trump over extracted names. For multiple key/certificate pairs, use the --tls-sni-cert-key multiple times. Examples: "example.crt,example.key" or "foo.crt,foo.key:*.foo.com,foo.com".

-v, --v int

number for the log level verbosity

--version version[=true]

--version, --version=raw prints version information and quits; --version=vX.Y.Z... sets the reported version

--vmodule pattern=N,...

comma-separated list of pattern=N settings for file-filtered logging (only works for text log format)

--write-config-to string

If set, write the configuration values to this file and exit.

---

kube-proxy Configuration (v1alpha1)

Resource Types

- KubeProxyConfiguration

`FormatOptions`

Appears in:

- LoggingConfiguration

FormatOptions contains options for the different logging formats.

Field	Description
`text` **[Required]** `TextOptions`	[Alpha] Text contains options for logging format "text". Only available when the LoggingAlphaOptions feature gate is enabled.
`json` **[Required]** `JSONOptions`	[Alpha] JSON contains options for logging format "json". Only available when the LoggingAlphaOptions feature gate is enabled.

`JSONOptions`

Appears in:

- FormatOptions

JSONOptions contains options for logging format "json".

Field	Description
`OutputRoutingOptions` **[Required]** `OutputRoutingOptions`	(Members of `OutputRoutingOptions` are embedded into this type.) No description provided.

`LogFormatFactory`

LogFormatFactory provides support for a certain additional, non-default log format.

`LoggingConfiguration`

Appears in:

- KubeProxyConfiguration
- KubeletConfiguration

LoggingConfiguration contains logging options.

Field	Description
`format` **[Required]** `string`	Format Flag specifies the structure of log messages. default value of format is `text`
`flushFrequency` **[Required]** `TimeOrMetaDuration`	Maximum time between log flushes. If a string, parsed as a duration (i.e. "1s") If an int, the maximum number of nanoseconds (i.e. 1s = 1000000000). Ignored if the selected logging backend writes log messages without buffering.
`verbosity` **[Required]** `VerbosityLevel`	Verbosity is the threshold that determines which log messages are logged. Default is zero which logs only the most important messages. Higher values enable additional messages. Error messages are always logged.
`vmodule` **[Required]** `VModuleConfiguration`	VModule overrides the verbosity threshold for individual files. Only supported for "text" log format.
`options` **[Required]** `FormatOptions`	[Alpha] Options holds additional parameters that are specific to the different logging formats. Only the options for the selected format get used, but all of them get validated. Only available when the LoggingAlphaOptions feature gate is enabled.

`LoggingOptions`

LoggingOptions can be used with ValidateAndApplyWithOptions to override certain global defaults.

Field	Description
`ErrorStream` **[Required]** `io.Writer`	ErrorStream can be used to override the os.Stderr default.
`InfoStream` **[Required]** `io.Writer`	InfoStream can be used to override the os.Stdout default.

`OutputRoutingOptions`

Appears in:

- JSONOptions
- TextOptions

OutputRoutingOptions contains options that are supported by both "text" and "json".

Field	Description
`splitStream` **[Required]** `bool`	[Alpha] SplitStream redirects error messages to stderr while info messages go to stdout, with buffering. The default is to write both to stdout, without buffering. Only available when the LoggingAlphaOptions feature gate is enabled.
`infoBufferSize` **[Required]** `k8s.io/apimachinery/pkg/api/resource.QuantityValue`	[Alpha] InfoBufferSize sets the size of the info stream when using split streams. The default is zero, which disables buffering. Only available when the LoggingAlphaOptions feature gate is enabled.

`TextOptions`

Appears in:

- FormatOptions

TextOptions contains options for logging format "text".

Field	Description
`OutputRoutingOptions` **[Required]** `OutputRoutingOptions`	(Members of `OutputRoutingOptions` are embedded into this type.) No description provided.

`TimeOrMetaDuration`

Appears in:

- LoggingConfiguration

TimeOrMetaDuration is present only for backwards compatibility for the flushFrequency field, and new fields should use metav1.Duration.

Field	Description
`Duration` **[Required]** `meta/v1.Duration`	Duration holds the duration
`-` **[Required]** `bool`	SerializeAsString controls whether the value is serialized as a string or an integer

`VModuleConfiguration`

(Alias of `[]k8s.io/component-base/logs/api/v1.VModuleItem`)

Appears in:

- LoggingConfiguration

VModuleConfiguration is a collection of individual file names or patterns and the corresponding verbosity threshold.

`VerbosityLevel`

(Alias of `uint32`)

Appears in:

- LoggingConfiguration

VerbosityLevel represents a klog or logr verbosity threshold.

`ClientConnectionConfiguration`

Appears in:

- KubeProxyConfiguration
- KubeSchedulerConfiguration
- GenericControllerManagerConfiguration

ClientConnectionConfiguration contains details for constructing a client.

Field	Description
`kubeconfig` **[Required]** `string`	kubeconfig is the path to a KubeConfig file.
`acceptContentTypes` **[Required]** `string`	acceptContentTypes defines the Accept header sent by clients when connecting to a server, overriding the default value of 'application/json'. This field will control all connections to the server used by a particular client.
`contentType` **[Required]** `string`	contentType is the content type used when sending data to the server from this client.
`qps` **[Required]** `float32`	qps controls the number of queries per second allowed for this connection.
`burst` **[Required]** `int32`	burst allows extra queries to accumulate when a client is exceeding its rate.

`DebuggingConfiguration`

Appears in:

- KubeSchedulerConfiguration
- GenericControllerManagerConfiguration

DebuggingConfiguration holds configuration for Debugging related features.

Field	Description
`enableProfiling` **[Required]** `bool`	enableProfiling enables profiling via web interface host:port/debug/pprof/
`enableContentionProfiling` **[Required]** `bool`	enableContentionProfiling enables block profiling, if enableProfiling is true.

LeaderElectionConfiguration

Appears in:

- KubeSchedulerConfiguration
- GenericControllerManagerConfiguration

LeaderElectionConfiguration defines the configuration of leader election clients for components that can run with leader election enabled.

Field	Description
`leaderElect` **[Required]** `bool`	leaderElect enables a leader election client to gain leadership before executing the main loop. Enable this when running replicated components for high availability.
`leaseDuration` **[Required]** `meta/v1.Duration`	leaseDuration is the duration that non-leader candidates will wait after observing a leadership renewal until attempting to acquire leadership of a led but unrenewed leader slot. This is effectively the maximum duration that a leader can be stopped before it is replaced by another candidate. This is only applicable if leader election is enabled.
`renewDeadline` **[Required]** `meta/v1.Duration`	renewDeadline is the interval between attempts by the acting master to renew a leadership slot before it stops leading. This must be less than or equal to the lease duration. This is only applicable if leader election is enabled.
`retryPeriod` **[Required]** `meta/v1.Duration`	retryPeriod is the duration the clients should wait between attempting acquisition and renewal of a leadership. This is only applicable if leader election is enabled.
`resourceLock` **[Required]** `string`	resourceLock indicates the resource object type that will be used to lock during leader election cycles.
`resourceName` **[Required]** `string`	resourceName indicates the name of resource object that will be used to lock during leader election cycles.
`resourceNamespace` **[Required]** `string`	resourceName indicates the namespace of resource object that will be used to lock during leader election cycles.

KubeProxyConfiguration

KubeProxyConfiguration contains everything necessary to configure the Kubernetes proxy server.

Field	Description
`apiVersion` string	`kubeproxy.config.k8s.io/v1alpha1`
`kind` string	`KubeProxyConfiguration`
`featureGates` **[Required]** `map[string]bool`	featureGates is a map of feature names to bools that enable or disable alpha/experimental features.
`clientConnection` **[Required]** `ClientConnectionConfiguration`	clientConnection specifies the kubeconfig file and client connection settings for the proxy server to use when communicating with the apiserver.
`logging` **[Required]** `LoggingConfiguration`	logging specifies the options of logging. Refer to Logs Options for more information.
`hostnameOverride` **[Required]** `string`	hostnameOverride, if non-empty, will be used as the name of the Node that kube-proxy is running on. If unset, the node name is assumed to be the same as the node's hostname.
`bindAddress` **[Required]** `string`	bindAddress can be used to override kube-proxy's idea of what its node's primary IP is. Note that the name is a historical artifact, and kube-proxy does not actually bind any sockets to this IP.
`healthzBindAddress` **[Required]** `string`	healthzBindAddress is the IP address and port for the health check server to serve on, defaulting to "0.0.0.0:10256" (if bindAddress is unset or IPv4), or "[::]:10256" (if bindAddress is IPv6).
`metricsBindAddress` **[Required]** `string`	metricsBindAddress is the IP address and port for the metrics server to serve on, defaulting to "127.0.0.1:10249" (if bindAddress is unset or IPv4), or "[::1]:10249" (if bindAddress is IPv6). (Set to "0.0.0.0:10249" / "[::]:10249" to bind on all interfaces.)
`bindAddressHardFail` **[Required]** `bool`	bindAddressHardFail, if true, tells kube-proxy to treat failure to bind to a port as fatal and exit
`enableProfiling` **[Required]** `bool`	enableProfiling enables profiling via web interface on /debug/pprof handler. Profiling handlers will be handled by metrics server.
`showHiddenMetricsForVersion` **[Required]** `string`	showHiddenMetricsForVersion is the version for which you want to show hidden metrics.
`mode` **[Required]** `ProxyMode`	mode specifies which proxy mode to use.
`iptables` **[Required]** `KubeProxyIPTablesConfiguration`	iptables contains iptables-related configuration options.
`ipvs` **[Required]** `KubeProxyIPVSConfiguration`	ipvs contains ipvs-related configuration options.
`nftables` **[Required]** `KubeProxyNFTablesConfiguration`	nftables contains nftables-related configuration options.

Field	Description
`winkernel` **[Required]** KubeProxyWinkernelConfiguration	winkernel contains winkernel-related configuration options.
`detectLocalMode` **[Required]** LocalMode	detectLocalMode determines mode to use for detecting local traffic, defaults to ClusterCIDR
`detectLocal` **[Required]** DetectLocalConfiguration	detectLocal contains optional configuration settings related to DetectLocalMode.
`clusterCIDR` **[Required]** `string`	clusterCIDR is the CIDR range of the pods in the cluster. (For dual-stack clusters, this can be a comma-separated dual-stack pair of CIDR ranges.). When DetectLocalMode is set to ClusterCIDR, kube-proxy will consider traffic to be local if its source IP is in this range. (Otherwise it is not used.)
`nodePortAddresses` **[Required]** `[]string`	nodePortAddresses is a list of CIDR ranges that contain valid node IPs, or alternatively, the single string 'primary'. If set to a list of CIDRs, connections to NodePort services will only be accepted on node IPs in one of the indicated ranges. If set to 'primary', NodePort services will only be accepted on the node's primary IPv4 and/or IPv6 address according to the Node object. If unset, NodePort connections will be accepted on all local IPs.
`oomScoreAdj` **[Required]** `int32`	oomScoreAdj is the oom-score-adj value for kube-proxy process. Values must be within the range [-1000, 1000]
`conntrack` **[Required]** KubeProxyConntrackConfiguration	conntrack contains conntrack-related configuration options.
`configSyncPeriod` **[Required]** meta/v1.Duration	configSyncPeriod is how often configuration from the apiserver is refreshed. Must be greater than 0.
`portRange` **[Required]** `string`	portRange was previously used to configure the userspace proxy, but is now unused.
`windowsRunAsService` **[Required]** `bool`	windowsRunAsService, if true, enables Windows service control manager API integration.

DetectLocalConfiguration

Appears in:

- KubeProxyConfiguration

DetectLocalConfiguration contains optional settings related to DetectLocalMode option

Field	Description
`bridgeInterface` **[Required]** `string`	bridgeInterface is a bridge interface name. When DetectLocalMode is set to LocalModeBridgeInterface, kube-proxy will consider traffic to be local if it originates from this bridge.
`interfaceNamePrefix` **[Required]** `string`	interfaceNamePrefix is an interface name prefix. When DetectLocalMode is set to LocalModeInterfaceNamePrefix, kube-proxy will consider traffic to be local if it originates from any interface whose name begins with this prefix.

KubeProxyConntrackConfiguration

Appears in:

- KubeProxyConfiguration

KubeProxyConntrackConfiguration contains conntrack settings for the Kubernetes proxy server.

Field	Description
`maxPerCore` **[Required]** `int32`	maxPerCore is the maximum number of NAT connections to track per CPU core (0 to leave the limit as-is and ignore min).
`min` **[Required]** `int32`	min is the minimum value of connect-tracking records to allocate, regardless of maxPerCore (set maxPerCore=0 to leave the limit as-is).
`tcpEstablishedTimeout` **[Required]** meta/v1.Duration	tcpEstablishedTimeout is how long an idle TCP connection will be kept open (e.g. '2s'). Must be greater than 0 to set.
`tcpCloseWaitTimeout` **[Required]** meta/v1.Duration	tcpCloseWaitTimeout is how long an idle conntrack entry in CLOSE_WAIT state will remain in the conntrack table. (e.g. '60s'). Must be greater than 0 to set.
`tcpBeLiberal` **[Required]** `bool`	tcpBeLiberal, if true, kube-proxy will configure conntrack to run in liberal mode for TCP connections and packets with out-of-window sequence numbers won't be marked INVALID.
`udpTimeout` **[Required]** meta/v1.Duration	udpTimeout is how long an idle UDP conntrack entry in UNREPLIED state will remain in the conntrack table (e.g. '30s'). Must be greater than 0 to set.
`udpStreamTimeout` **[Required]** meta/v1.Duration	udpStreamTimeout is how long an idle UDP conntrack entry in ASSURED state will remain in the conntrack table (e.g. '300s'). Must be greater than 0 to set.

KubeProxyIPTablesConfiguration

Appears in:

- KubeProxyConfiguration

KubeProxyIPTablesConfiguration contains iptables-related configuration details for the Kubernetes proxy server.

Field	Description
`masqueradeBit` **[Required]** `int32`	masqueradeBit is the bit of the iptables fwmark space to use for SNAT if using the iptables or ipvs proxy mode. Values must be within the range [0, 31].
`masqueradeAll` **[Required]** `bool`	masqueradeAll tells kube-proxy to SNAT all traffic sent to Service cluster IPs, when using the iptables or ipvs proxy mode. This may be required with some CNI plugins.
`localhostNodePorts` **[Required]** `bool`	localhostNodePorts, if false, tells kube-proxy to disable the legacy behavior of allowing NodePort services to be accessed via localhost. (Applies only to iptables mode and IPv4; localhost NodePorts are never allowed with other proxy modes or with IPv6.)
`syncPeriod` **[Required]** `meta/v1.Duration`	syncPeriod is an interval (e.g. '5s', '1m', '2h22m') indicating how frequently various re-synchronizing and cleanup operations are performed. Must be greater than 0.
`minSyncPeriod` **[Required]** `meta/v1.Duration`	minSyncPeriod is the minimum period between iptables rule resyncs (e.g. '5s', '1m', '2h22m'). A value of 0 means every Service or EndpointSlice change will result in an immediate iptables resync.

KubeProxyIPVSConfiguration

Appears in:

- KubeProxyConfiguration

KubeProxyIPVSConfiguration contains ipvs-related configuration details for the Kubernetes proxy server.

Field	Description
`syncPeriod` **[Required]** `meta/v1.Duration`	syncPeriod is an interval (e.g. '5s', '1m', '2h22m') indicating how frequently various re-synchronizing and cleanup operations are performed. Must be greater than 0.
`minSyncPeriod` **[Required]** `meta/v1.Duration`	minSyncPeriod is the minimum period between IPVS rule resyncs (e.g. '5s', '1m', '2h22m'). A value of 0 means every Service or EndpointSlice change will result in an immediate IPVS resync.
`scheduler` **[Required]** `string`	scheduler is the IPVS scheduler to use
`excludeCIDRs` **[Required]** `[]string`	excludeCIDRs is a list of CIDRs which the ipvs proxier should not touch when cleaning up ipvs services.
`strictARP` **[Required]** `bool`	strictARP configures arp_ignore and arp_announce to avoid answering ARP queries from kube-ipvs0 interface
`tcpTimeout` **[Required]** `meta/v1.Duration`	tcpTimeout is the timeout value used for idle IPVS TCP sessions. The default value is 0, which preserves the current timeout value on the system.
`tcpFinTimeout` **[Required]** `meta/v1.Duration`	tcpFinTimeout is the timeout value used for IPVS TCP sessions after receiving a FIN. The default value is 0, which preserves the current timeout value on the system.
`udpTimeout` **[Required]** `meta/v1.Duration`	udpTimeout is the timeout value used for IPVS UDP packets. The default value is 0, which preserves the current timeout value on the system.

KubeProxyNFTablesConfiguration

Appears in:

- KubeProxyConfiguration

KubeProxyNFTablesConfiguration contains nftables-related configuration details for the Kubernetes proxy server.

Field	Description
`masqueradeBit` **[Required]** `int32`	masqueradeBit is the bit of the iptables fwmark space to use for SNAT if using the nftables proxy mode. Values must be within the range [0, 31].
`masqueradeAll` **[Required]** `bool`	masqueradeAll tells kube-proxy to SNAT all traffic sent to Service cluster IPs, when using the nftables mode. This may be required with some CNI plugins.
`syncPeriod` **[Required]** `meta/v1.Duration`	syncPeriod is an interval (e.g. '5s', '1m', '2h22m') indicating how frequently various re-synchronizing and cleanup operations are performed. Must be greater than 0.
`minSyncPeriod` **[Required]** `meta/v1.Duration`	minSyncPeriod is the minimum period between iptables rule resyncs (e.g. '5s', '1m', '2h22m'). A value of 0 means every Service or EndpointSlice change will result in an immediate iptables resync.

`KubeProxyWinkernelConfiguration`

Appears in:

- KubeProxyConfiguration

KubeProxyWinkernelConfiguration contains Windows/HNS settings for the Kubernetes proxy server.

Field	Description
`networkName` **[Required]** `string`	networkName is the name of the network kube-proxy will use to create endpoints and policies
`sourceVip` **[Required]** `string`	sourceVip is the IP address of the source VIP endpoint used for NAT when loadbalancing
`enableDSR` **[Required]** `bool`	enableDSR tells kube-proxy whether HNS policies should be created with DSR
`rootHnsEndpointName` **[Required]** `string`	rootHnsEndpointName is the name of hnsendpoint that is attached to l2bridge for root network namespace
`forwardHealthCheckVip` **[Required]** `bool`	forwardHealthCheckVip forwards service VIP for health check port on Windows

`LocalMode`

(Alias of `string`)

Appears in:

- KubeProxyConfiguration

LocalMode represents modes to detect local traffic from the node

`ProxyMode`

(Alias of `string`)

Appears in:

- KubeProxyConfiguration

ProxyMode represents modes used by the Kubernetes proxy server.

Three modes of proxy are available on Linux platforms: `iptables`, `ipvs`, and `nftables`. One mode of proxy is available on Windows platforms: `kernelspace`.

If the proxy mode is unspecified, a default proxy mode will be used (currently this is `iptables` on Linux and `kernelspace` on Windows). If the selected proxy mode cannot be used (due to lack of kernel support, missing userspace components, etc) then kube-proxy will exit with an error.

Event Rate Limit Configuration (v1alpha1)

Resource Types

- Configuration

`Configuration`

Configuration provides configuration for the EventRateLimit admission controller.

Field	Description
`apiVersion` string	`eventratelimit.admission.k8s.io/v1alpha1`
`kind` string	`Configuration`
`limits` **[Required]** `[]Limit`	limits are the limits to place on event queries received. Limits can be placed on events received server-wide, per namespace, per user, and per source+object. At least one limit is required.

`Limit`

Appears in:

- Configuration

Limit is the configuration for a particular limit type

Field	Description
`type` **[Required]** `LimitType`	type is the type of limit to which this configuration applies
`qps` **[Required]** `int32`	qps is the number of event queries per second that are allowed for this type of limit. The qps and burst fields are used together to determine if a particular event query is accepted. The qps determines how many queries are accepted once the burst amount of queries has been exhausted.
`burst` **[Required]** `int32`	burst is the burst number of event queries that are allowed for this type of limit. The qps and burst fields are used together to determine if a particular event query is accepted. The burst determines the maximum size of the allowance granted for a particular bucket. For example, if the burst is 10 and the qps is 3, then the admission control will accept 10 queries before blocking any queries. Every second, 3 more queries will be allowed. If some of that allowance is not used, then it will roll over to the next second, until the maximum allowance of 10 is reached.
`cacheSize` `int32`	cacheSize is the size of the LRU cache for this type of limit. If a bucket is evicted from the cache, then the allowance for that bucket is reset. If more queries are later received for an evicted bucket, then that bucket will re-enter the cache with a clean slate, giving that bucket a full allowance of burst queries. The default cache size is 4096. If limitType is 'server', then cacheSize is ignored.

`LimitType`

(Alias of `string`)

Appears in:

- Limit

LimitType is the type of the limit (e.g., per-namespace)

---

kubelet

Synopsis

The kubelet is the primary "node agent" that runs on each node. It can register the node with the apiserver using one of: the hostname; a flag to override the hostname; or specific logic for a cloud provider.

The kubelet works in terms of a PodSpec. A PodSpec is a YAML or JSON object that describes a pod. The kubelet takes a set of PodSpecs that are provided through various mechanisms (primarily through the apiserver) and ensures that the containers described in those PodSpecs are running and healthy. The kubelet doesn't manage containers which were not created by Kubernetes.

Other than from an PodSpec from the apiserver, there are two ways that a container manifest can be provided to the Kubelet.

File: Path passed as a flag on the command line. Files under this path will be monitored periodically for updates. The monitoring period is 20s by default and is configurable via a flag.

HTTP endpoint: HTTP endpoint passed as a parameter on the command line. This endpoint is checked every 20 seconds (also configurable with a flag).

```
kubelet [flags]
```

Options

--address string Default: 0.0.0.0

The IP address for the Kubelet to serve on (set to '0.0.0.0' or '::' for listening on all interfaces and IP address families) (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--allowed-unsafe-sysctls strings

Comma-separated whitelist of unsafe sysctls or unsafe sysctl patterns (ending in *). Use these at your own risk. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-

cluster/kubelet-config-file/ for more information.)

--anonymous-auth Default: true

Enables anonymous requests to the Kubelet server. Requests that are not rejected by another authentication method are treated as anonymous requests. Anonymous requests have a username of system:anonymous, and a group name of system:unauthenticated. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--authentication-token-webhook

Use the TokenReview API to determine authentication for bearer tokens. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--authentication-token-webhook-cache-ttl duration Default: 2m0s

The duration to cache responses from the webhook token authenticator. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--authorization-mode string Default: "AlwaysAllow"

Authorization mode for Kubelet server. Valid options are AlwaysAllow or Webhook. Webhook mode uses the SubjectAccessReview API to determine authorization. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--authorization-webhook-cache-authorized-ttl duration Default: 5m0s

The duration to cache 'authorized' responses from the webhook authorizer. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--authorization-webhook-cache-unauthorized-ttl duration Default: 30s

The duration to cache 'unauthorized' responses from the webhook authorizer. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--bootstrap-kubeconfig string

Path to a kubeconfig file that will be used to get client certificate for kubelet. If the file specified by --kubeconfig does not exist, the bootstrap kubeconfig is used to request a client certificate from the API server. On success, a kubeconfig file referencing the generated client certificate and key is written to the path specified by --kubeconfig. The client certificate and key file will be stored in the directory pointed by --cert-dir.

--cert-dir string Default: "/var/lib/kubelet/pki"

The directory where the TLS certs are located. If --tls-cert-file and --tls-private-key-file are provided, this flag will be ignored.

--cgroup-driver string Default: "cgroupfs"

Driver that the kubelet uses to manipulate cgroups on the host. Possible values: 'cgroupfs', 'systemd' (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cgroup-root string

Optional root cgroup to use for pods. This is handled by the container runtime on a best effort basis. Default: '', which means use the container runtime default. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cgroups-per-qos Default: true

Enable creation of QoS cgroup hierarchy, if true top level QoS and pod cgroups are created. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--client-ca-file string

If set, any request presenting a client certificate signed by one of the authorities in the client-ca-file is authenticated with an identity corresponding to the CommonName of the client certificate. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cloud-provider string

The provider for cloud services. Set to empty string for running with no cloud provider. Set to 'external' for running with an external cloud provider.

--cluster-dns strings

Comma-separated list of DNS server IP address. This value is used for containers DNS server in case of Pods with "dnsPolicy=ClusterFirst". Note: all DNS servers appearing in the list MUST serve the same set of records otherwise name resolution within the cluster may not work correctly. There is no guarantee as to which DNS server may be contacted for name resolution. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cluster-domain string

Domain for this cluster. If set, kubelet will configure all containers to search this domain in addition to the host's search domains (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--config string

The Kubelet will load its initial configuration from this file. The path may be absolute or relative; relative paths start at the Kubelet's current working directory. Omit this flag to use the built-in default configuration values. Command-line flags override configuration from this file.

--config-dir string

Path to a directory to specify drop-ins, allows the user to optionally specify additional configs to overwrite what is provided by default and in the KubeletConfigFile flag. [default='']

--container-log-max-files int32 Default: 5

<Warning: Beta feature> Set the maximum number of container log files that can be present for a container. The number must be >= 2. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--container-log-max-size string Default: "10Mi"

<Warning: Beta feature> Set the maximum size (e.g. 10Mi) of container log file before it is rotated. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--container-runtime-endpoint string Default: "unix:///run/containerd/containerd.sock"

The endpoint of container runtime service. Unix Domain Sockets are supported on Linux, while npipe and tcp endpoints are supported on Windows. Examples:'unix:///path/to/runtime.sock', 'npipe:////./pipe/runtime' (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--contention-profiling

Enable block profiling, if profiling is enabled (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cpu-cfs-quota Default: true

Enable CPU CFS quota enforcement for containers that specify CPU limits (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cpu-cfs-quota-period duration Default: 100ms

Sets CPU CFS quota period value, cpu.cfs_period_us, defaults to Linux Kernel default (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cpu-manager-policy string Default: "none"

CPU Manager policy to use. Possible values: 'none', 'static'. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cpu-manager-policy-options <comma-separated 'key=value' pairs>

A set of key=value CPU Manager policy options to use, to fine tune their behaviour. If not supplied, keep the default behaviour. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--cpu-manager-reconcile-period duration Default: 10s

<Warning: Alpha feature> CPU Manager reconciliation period. Examples: '10s', or '1m'. If not supplied, defaults to 'NodeStatusUpdateFrequency' (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--enable-controller-attach-detach Default: true

Enables the Attach/Detach controller to manage attachment/detachment of volumes scheduled to this node, and disables kubelet from executing any attach/detach operations (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--enable-debugging-handlers Default: true

Enables server endpoints for log collection and local running of containers and commands (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--enable-server Default: true

Enable the Kubelet's server (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--enforce-node-allocatable strings Default: "pods"

A comma separated list of levels of node allocatable enforcement to be enforced by kubelet. Acceptable options are 'none', 'pods', 'system-reserved', 'system-reserved-compressible', 'kube-reserved' and 'kube-reserved-compressible'. If any of the latter four options are specified, '--system-reserved-cgroup' and '--kube-reserved-cgroup' must also be set, respectively. If 'none' is specified, no additional options should be set. See https://kubernetes.io/docs/tasks/administer-cluster/reserve-compute-resources/ for more details. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--event-burst int32 Default: 100

Maximum size of a bursty event records, temporarily allows event records to burst to this number, while still not exceeding event-qps. The number must be >= 0. If 0 will use DefaultBurst: 10. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--event-qps int32 Default: 50

QPS to limit event creations. The number must be >= 0. If 0 will use DefaultQPS: 5. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--eviction-hard <comma-separated 'key=value' pairs>

A set of eviction thresholds (e.g. memory.available<1Gi) that if met would trigger a pod eviction. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--eviction-max-pod-grace-period int32

Maximum allowed grace period (in seconds) to use when terminating pods in response to a soft eviction threshold being met. The pod's effective grace period is calculated as min(evictionMaxPodGracePeriod, pod.terminationGracePeriodSeconds). A negative value will cause pods to be terminated immediately, as if the value was 0. (DEPRECATED: This parameter should be set via the config file specified by the kubelet's `--config` flag. See kubelet-config-file for more information.)

--eviction-minimum-reclaim <comma-separated 'key=value' pairs>

A set of minimum reclaims (e.g. imagefs.available=2Gi) that describes the minimum amount of resource the kubelet will reclaim when performing a pod eviction if that resource is under pressure. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--eviction-pressure-transition-period duration Default: 5m0s

Duration for which the kubelet has to wait before transitioning out of an eviction pressure condition. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--eviction-soft <comma-separated 'key=value' pairs>

A set of eviction thresholds (e.g. memory.available<1.5Gi) that if met over a corresponding grace period would trigger a pod eviction. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--eviction-soft-grace-period <comma-separated 'key=value' pairs>

A set of eviction grace periods (e.g. memory.available=1m30s) that correspond to how long a soft eviction threshold must hold before triggering a pod eviction. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--exit-on-lock-contention

Whether kubelet should exit upon lock-file contention.

--experimental-allocatable-ignore-eviction

When set to 'true', Hard Eviction Thresholds will be ignored while calculating Node Allocatable. See https://kubernetes.io/docs/tasks/administer-cluster/reserve-compute-resources/ for more details. [default=false] (DEPRECATED: will be removed in 1.25 or later.)

--experimental-mounter-path string

[Experimental] Path of mounter binary. Leave empty to use the default mount. (DEPRECATED: will be removed in 1.25 or later. in favor of using CSI.)

--fail-cgroupv1

Prevent the kubelet from starting on the host using cgroup v1.

--fail-swap-on Default: true

Makes the Kubelet fail to start if swap is enabled on the node. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--feature-gates <comma-separated 'key=True|False' pairs>

A set of key=value pairs that describe feature gates for alpha/experimental features. Options are:
APIResponseCompression=true|false (BETA - default=true)
APIServerIdentity=true|false (BETA - default=true)
APIServingWithRoutine=true|false (ALPHA - default=false)
AllAlpha=true|false (ALPHA - default=false)

AllBeta=true|false (BETA - default=false)
AllowParsingUserUIDFromCertAuth=true|false (BETA - default=true)
AllowUnsafeMalformedObjectDeletion=true|false (ALPHA - default=false)
CBORServingAndStorage=true|false (ALPHA - default=false)
CPUManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
CPUManagerPolicyBetaOptions=true|false (BETA - default=true)
CSIVolumeHealth=true|false (ALPHA - default=false)
ClearingNominatedNodeNameAfterBinding=true|false (ALPHA - default=false)
ClientsAllowCBOR=true|false (ALPHA - default=false)
ClientsPreferCBOR=true|false (ALPHA - default=false)
CloudControllerManagerWebhook=true|false (ALPHA - default=false)
ClusterTrustBundle=true|false (BETA - default=false)
ClusterTrustBundleProjection=true|false (BETA - default=false)
ComponentFlagz=true|false (ALPHA - default=false)
ComponentStatusz=true|false (ALPHA - default=false)
ConcurrentWatchObjectDecode=true|false (BETA - default=false)
ContainerCheckpoint=true|false (BETA - default=true)
ContainerRestartRules=true|false (ALPHA - default=false)
ContainerStopSignals=true|false (ALPHA - default=false)
ContextualLogging=true|false (BETA - default=true)
CoordinatedLeaderElection=true|false (BETA - default=false)
CrossNamespaceVolumeDataSource=true|false (ALPHA - default=false)
CustomCPUCFSQuotaPeriod=true|false (ALPHA - default=false)
DRAAdminAccess=true|false (BETA - default=true)
DRAConsumableCapacity=true|false (ALPHA - default=false)
DRADeviceBindingConditions=true|false (ALPHA - default=false)
DRADeviceTaints=true|false (ALPHA - default=false)
DRAExtendedResource=true|false (ALPHA - default=false)
DRAPartitionableDevices=true|false (ALPHA - default=false)
DRAPrioritizedList=true|false (BETA - default=true)
DRAResourceClaimDeviceStatus=true|false (BETA - default=true)
DRASchedulerFilterTimeout=true|false (BETA - default=true)
DeclarativeValidation=true|false (BETA - default=true)
DeclarativeValidationTakeover=true|false (BETA - default=false)
DeploymentReplicaSetTerminatingReplicas=true|false (ALPHA - default=false)
DetectCacheInconsistency=true|false (BETA - default=true)
DisableCPUQuotaWithExclusiveCPUs=true|false (BETA - default=true)
EnvFiles=true|false (ALPHA - default=false)
EventedPLEG=true|false (ALPHA - default=false)
ExternalServiceAccountTokenSigner=true|false (BETA - default=true)
GracefulNodeShutdown=true|false (BETA - default=true)
GracefulNodeShutdownBasedOnPodPriority=true|false (BETA - default=true)
HPAConfigurableTolerance=true|false (ALPHA - default=false)
HPAScaleToZero=true|false (ALPHA - default=false)
HostnameOverride=true|false (ALPHA - default=false)
ImageMaximumGCAge=true|false (BETA - default=true)
ImageVolume=true|false (BETA - default=false)
InOrderInformers=true|false (BETA - default=true)
InPlacePodVerticalScaling=true|false (BETA - default=true)
InPlacePodVerticalScalingExclusiveCPUs=true|false (ALPHA - default=false)
InPlacePodVerticalScalingExclusiveMemory=true|false (ALPHA - default=false)
InTreePluginPortworxUnregister=true|false (ALPHA - default=false)
InformerResourceVersion=true|false (ALPHA - default=false)
JobManagedBy=true|false (BETA - default=true)
KubeletCrashLoopBackOffMax=true|false (ALPHA - default=false)
KubeletEnsureSecretPulledImages=true|false (ALPHA - default=false)
KubeletFineGrainedAuthz=true|false (BETA - default=true)
KubeletInUserNamespace=true|false (ALPHA - default=false)
KubeletPSI=true|false (BETA - default=true)
KubeletPodResourcesDynamicResources=true|false (BETA - default=true)
KubeletPodResourcesGet=true|false (BETA - default=true)
KubeletSeparateDiskGC=true|false (BETA - default=true)

KubeletServiceAccountTokenForCredentialProviders=true|false (BETA - default=true)
ListFromCacheSnapshot=true|false (BETA - default=true)
LocalStorageCapacityIsolationFSQuotaMonitoring=true|false (BETA - default=false)
LoggingAlphaOptions=true|false (ALPHA - default=false)
LoggingBetaOptions=true|false (BETA - default=true)
MatchLabelKeysInPodTopologySpread=true|false (BETA - default=true)
MatchLabelKeysInPodTopologySpreadSelectorMerge=true|false (BETA - default=true)
MaxUnavailableStatefulSet=true|false (ALPHA - default=false)
MemoryQoS=true|false (ALPHA - default=false)
MutableCSINodeAllocatableCount=true|false (BETA - default=false)
MutatingAdmissionPolicy=true|false (BETA - default=false)
NodeLogQuery=true|false (BETA - default=false)
NominatedNodeNameForExpectation=true|false (ALPHA - default=false)
OpenAPIEnums=true|false (BETA - default=true)
PodAndContainerStatsFromCRI=true|false (ALPHA - default=false)
PodCertificateRequest=true|false (ALPHA - default=false)
PodDeletionCost=true|false (BETA - default=true)
PodLevelResources=true|false (BETA - default=true)
PodLogsQuerySplitStreams=true|false (ALPHA - default=false)
PodObservedGenerationTracking=true|false (BETA - default=true)
PodReadyToStartContainersCondition=true|false (BETA - default=true)
PodTopologyLabelsAdmission=true|false (ALPHA - default=false)
PortForwardWebsockets=true|false (BETA - default=true)
PreferSameTrafficDistribution=true|false (BETA - default=true)
PreventStaticPodAPIReferences=true|false (BETA - default=true)
ProcMountType=true|false (BETA - default=true)
QOSReserved=true|false (ALPHA - default=false)
ReduceDefaultCrashLoopBackOffDecay=true|false (ALPHA - default=false)
RelaxedServiceNameValidation=true|false (ALPHA - default=false)
ReloadKubeletServerCertificateFile=true|false (BETA - default=true)
RemoteRequestHeaderUID=true|false (BETA - default=true)
ResourceHealthStatus=true|false (ALPHA - default=false)
RotateKubeletServerCertificate=true|false (BETA - default=true)
RuntimeClassInImageCriApi=true|false (ALPHA - default=false)
SELinuxChangePolicy=true|false (BETA - default=true)
SELinuxMount=true|false (BETA - default=false)
SELinuxMountReadWriteOncePod=true|false (BETA - default=true)
SchedulerAsyncAPICalls=true|false (BETA - default=true)
SchedulerAsyncPreemption=true|false (BETA - default=true)
SchedulerPopFromBackoffQ=true|false (BETA - default=true)
ServiceAccountNodeAudienceRestriction=true|false (BETA - default=true)
SizeBasedListCostEstimate=true|false (BETA - default=true)
StorageCapacityScoring=true|false (ALPHA - default=false)
StorageVersionAPI=true|false (ALPHA - default=false)
StorageVersionHash=true|false (BETA - default=true)
StorageVersionMigrator=true|false (ALPHA - default=false)
StrictIPCIDRValidation=true|false (ALPHA - default=false)
StructuredAuthenticationConfigurationEgressSelector=true|false (BETA - default=true)
SupplementalGroupsPolicy=true|false (BETA - default=true)
SystemdWatchdog=true|false (BETA - default=true)
TokenRequestServiceAccountUIDValidation=true|false (BETA - default=true)
TopologyManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
TopologyManagerPolicyBetaOptions=true|false (BETA - default=true)
TranslateStreamCloseWebsocketRequests=true|false (BETA - default=true)
UnauthenticatedHTTP2DOSMitigation=true|false (BETA - default=true)
UnknownVersionInteroperabilityProxy=true|false (ALPHA - default=false)
UserNamespacesPodSecurityStandards=true|false (ALPHA - default=false)
UserNamespacesSupport=true|false (BETA - default=true)
WatchCacheInitializationPostStartHook=true|false (BETA - default=false)
WatchList=true|false (BETA - default=true)
WatchListClient=true|false (BETA - default=false)
WindowsCPUAndMemoryAffinity=true|false (ALPHA - default=false)

WindowsGracefulNodeShutdown=true|false (BETA - default=true) (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--file-check-frequency duration Default: 20s

Duration between checking config files for new data (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--hairpin-mode string Default: "promiscuous-bridge"

How should the kubelet setup hairpin NAT. This allows endpoints of a Service to loadbalance back to themselves if they should try to access their own Service. Valid values are "promiscuous-bridge", "hairpin-veth" and "none". (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--healthz-bind-address string Default: 127.0.0.1

The IP address for the healthz server to serve on (set to '0.0.0.0' or '::' for listening on all interfaces and IP address families) (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--healthz-port int32 Default: 10248

The port of the localhost healthz endpoint (set to 0 to disable) (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

-h, --help

help for kubelet

--hostname-override string

If non-empty, will use this string as identification instead of the actual hostname.

--http-check-frequency duration Default: 20s

Duration between checking http for new data (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--image-credential-provider-bin-dir string

The path to the directory where credential provider plugin binaries are located.

--image-credential-provider-config string

Path to a credential provider plugin config file (JSON/YAML/YML) or a directory of such files (merged in lexicographical order; non-recursive search).

--image-gc-high-threshold int32 Default: 85

The percent of disk usage after which image garbage collection is always run. Values must be within the range [0, 100], To disable image garbage collection, set to 100. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--image-gc-low-threshold int32 Default: 80

The percent of disk usage before which image garbage collection is never run. Lowest disk usage to garbage collect to. Values must be within the range [0, 100] and must be less than that of --image-gc-high-threshold. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--image-service-endpoint string

The endpoint of container image service. If not specified, it will be the same with --container-runtime-endpoint by default. Unix Domain Socket are supported on Linux, while npipe and tcp endpoints are supported on Windows. Examples:'unix:///path/to/runtime.sock', 'npipe:////./pipe/runtime' (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--kernel-memcg-notification

If enabled, the kubelet will integrate with the kernel memcg notification to determine if memory eviction thresholds are crossed rather than polling. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--kube-api-burst int32 Default: 100

Burst to use while talking with kubernetes apiserver. The number must be >= 0. If 0 will use DefaultBurst: 100. Doesn't cover events and node heartbeat apis which rate limiting is controlled by a different set of flags (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--kube-api-content-type string Default: "application/vnd.kubernetes.protobuf"

Content type of requests sent to apiserver. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--kube-api-qps int32 Default: 50

QPS to use while talking with kubernetes apiserver. The number must be >= 0. If 0 will use DefaultQPS: 50. Doesn't cover events and node heartbeat apis which rate limiting is controlled by a different set of flags (DEPRECATED: This parameter should be set via the

config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--kube-reserved <comma-separated 'key=value' pairs>

A set of ResourceName=ResourceQuantity (e.g. cpu=200m,memory=500Mi,ephemeral-storage=1Gi,pid=1000) pairs that describe resources reserved for kubernetes system components. Currently only cpu, memory, pid and local ephemeral storage for root file system are supported. See https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ for more detail. [default=none] (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--kube-reserved-cgroup string

Absolute name of the top level cgroup that is used to manage kubernetes components for which compute resources were reserved via '--kube-reserved' flag. Ex. '/kube-reserved'. [default=''] (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--kubeconfig string

Path to a kubeconfig file, specifying how to connect to the API server. Providing --kubeconfig enables API server mode, omitting --kubeconfig enables standalone mode.

--kubelet-cgroups string

Optional absolute name of cgroups to create and run the Kubelet in. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--local-storage-capacity-isolation Default: true

If true, local ephemeral storage isolation is enabled. Otherwise, local storage isolation feature will be disabled (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--lock-file string

<Warning: Alpha feature> The path to file for kubelet to use as a lock file.

--log-flush-frequency duration Default: 5s

Maximum number of seconds between log flushes

--log-text-info-buffer-size quantity

[Alpha] In text format with split output streams, the info messages can be buffered for a while to increase performance. The default value of zero bytes disables buffering. The size can be specified as number of bytes (512), multiples of 1000 (1K), multiples of 1024 (2Ki), or powers of those (3M, 4G, 5Mi, 6Gi). Enable the LoggingAlphaOptions feature gate to use this. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--log-text-split-stream

[Alpha] In text format, write error messages to stderr and info messages to stdout. The default is to write a single stream to stdout. Enable the LoggingAlphaOptions feature gate to use this. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--logging-format string Default: "text"

Sets the log format. Permitted formats: "text". (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--make-iptables-util-chains Default: true

If true, kubelet will ensure iptables utility rules are present on host. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--manifest-url string

URL for accessing additional Pod specifications to run (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--manifest-url-header colonSeparatedMultimapStringString

Comma-separated list of HTTP headers to use when accessing the url provided to --manifest-url. Multiple headers with the same name will be added in the same order provided. This flag can be repeatedly invoked. For example: --manifest-url-header 'a:hello,b:again,c:world' --manifest-url-header 'b:beautiful' (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--max-open-files int Default: 1000000

Number of files that can be opened by Kubelet process. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--max-pods int32 Default: 110

Number of Pods that can run on this Kubelet. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--maximum-dead-containers int32 Default: -1

Maximum number of old instances of containers to retain globally. Each container takes up some disk space. To disable, set to a negative number. (DEPRECATED: Use --eviction-hard or --eviction-soft instead. Will be removed in a future version.)

--maximum-dead-containers-per-container int32 Default: 1

Maximum number of old instances to retain per container. Each container takes up some disk space. (DEPRECATED: Use --eviction-hard or --eviction-soft instead. Will be removed in a future version.)

--memory-manager-policy string Default: "None"

Memory Manager policy to use. Possible values: 'None', 'Static'. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--minimum-container-ttl-duration duration

Minimum age for a finished container before it is garbage collected. Examples: '300ms', '10s' or '2h45m' (DEPRECATED: Use --eviction-hard or --eviction-soft instead. Will be removed in a future version.)

--minimum-image-ttl-duration duration Default: 2m0s

Minimum age for an unused image before it is garbage collected. Examples: '300ms', '10s' or '2h45m'. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--node-ip string

IP address (or comma-separated dual-stack IP addresses) of the node. If unset, kubelet will use the node's default IPv4 address, if any, or its default IPv6 address if it has no IPv4 addresses. You can pass '::' to make it prefer the default IPv6 address rather than the default IPv4 address. If cloud-provider is set to external, this flag will help to bootstrap the node with the corresponding IP.

--node-labels <comma-separated 'key=value' pairs>

Labels to add when registering the node in the cluster. Labels must be key=value pairs separated by ','. Labels in the 'kubernetes.io' namespace must begin with an allowed prefix (kubelet.kubernetes.io, node.kubernetes.io) or be in the specifically allowed set (beta.kubernetes.io/arch, beta.kubernetes.io/instance-type, beta.kubernetes.io/os, failure-domain.beta.kubernetes.io/region, failure-domain.beta.kubernetes.io/zone, kubernetes.io/arch, kubernetes.io/hostname, kubernetes.io/os, node.kubernetes.io/instance-type, topology.kubernetes.io/region, topology.kubernetes.io/zone)

--node-status-max-images int32 Default: 50

The maximum number of images to report in Node.Status.Images. If -1 is specified, no cap will be applied. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--node-status-update-frequency duration Default: 10s

Specifies how often kubelet posts node status to master. Note: be cautious when changing the constant, it must work with nodeMonitorGracePeriod in nodecontroller. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--oom-score-adj int32 Default: -999

The oom-score-adj value for kubelet process. Values must be within the range [-1000, 1000] (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--pod-cidr string

The CIDR to use for pod IP addresses, only used in standalone mode. In cluster mode, this is obtained from the master. For IPv6, the maximum number of IP's allocated is 65536 (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--pod-infra-container-image string

Specified image will not be pruned by the image garbage collector. CRI implementations have their own configuration to set this image. (DEPRECATED: will be removed in 1.35. Image garbage collector will get sandbox image information from CRI.)

--pod-manifest-path string

Path to the directory containing static pod files to run, or the path to a single static pod file. Files starting with dots will be ignored. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--pod-max-pids int Default: -1

Set the maximum number of processes per pod. If -1, the kubelet defaults to the node allocatable pid capacity. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--pods-per-core int32

Number of Pods per core that can run on this Kubelet. The total number of Pods on this Kubelet cannot exceed max-pods, so max-pods will be used if this calculation results in a larger number of Pods allowed on the Kubelet. A value of 0 disables this limit. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--port int32 Default: 10250

The port for the Kubelet to serve on. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--protect-kernel-defaults

Default kubelet behaviour for kernel tuning. If set, kubelet errors if any of kernel tunables is different than kubelet defaults. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--provider-id string

Unique identifier for identifying the node in a machine database, i.e cloudprovider

--qos-reserved <comma-separated 'key=value' pairs>

<Warning: Alpha feature> A set of ResourceName=Percentage (e.g. memory=50%) pairs that describe how pod resource requests are reserved at the QoS level. Currently only memory is supported. Requires the QOSReserved feature gate to be enabled. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--read-only-port int32 Default: 10255

The read-only port for the Kubelet to serve on with no authentication/authorization (set to 0 to disable) (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--register-node Default: true

Register the node with the apiserver. If --kubeconfig is not provided, this flag is irrelevant, as the Kubelet won't have an apiserver to register with. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--register-with-taints []v1.Taint

Register the node with the given list of taints (comma separated "<key>=<value>:<effect>"). No-op if register-node is false. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--registry-burst int32 Default: 10

Maximum size of a bursty pulls, temporarily allows pulls to burst to this number, while still not exceeding registry-qps. Only used if --registry-qps > 0 (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--registry-qps int32 Default: 5

If > 0, limit registry pull QPS to this value. If 0, unlimited. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--reserved-cpus string

A comma-separated list of CPUs or CPU ranges that are reserved for system and kubernetes usage. This specific list will supersede cpu counts in --system-reserved and --kube-reserved. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--reserved-memory reserved-memory

A comma separated list of memory reservations for NUMA nodes. (e.g. --reserved-memory 0:memory=1Gi,hugepages-1M=2Gi --reserved-memory 1:memory=2Gi). The total sum for each memory type should be equal to the sum of kube-reserved, system-reserved and eviction-threshold. See https://kubernetes.io/docs/tasks/administer-cluster/memory-manager/#reserved-memory-flag for more details. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--resolv-conf string Default: "/etc/resolv.conf"

Resolver configuration file used as the basis for the container DNS resolution configuration. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--root-dir string Default: "/var/lib/kubelet"

Directory path for managing kubelet files (volume mounts,etc).

--rotate-certificates

Auto rotate the kubelet client certificates by requesting new certificates from the kube-apiserver when the certificate expiration approaches. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--rotate-server-certificates

Auto-request and rotate the kubelet serving certificates by requesting new certificates from the kube-apiserver when the certificate expiration approaches. Requires the RotateKubeletServerCertificate feature gate to be enabled, and approval of the submitted CertificateSigningRequest objects. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--runonce

If true, exit after spawning pods from static pod files or remote urls. Exclusive with --enable-server (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--runtime-cgroups string

Optional absolute name of cgroups to create and run the runtime in.

--runtime-request-timeout duration Default: 2m0s

Timeout of all runtime requests except long running request - pull, logs, exec and attach. When timeout exceeded, kubelet will cancel the request, throw out an error and retry later. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--seccomp-default RuntimeDefault

Enable the use of RuntimeDefault as the default seccomp profile for all workloads.

--serialize-image-pulls Default: true

Pull images one at a time. We recommend *not* changing the default value on nodes that run docker daemon with version < 1.9 or an Aufs storage backend. Issue #10959 has more details. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--streaming-connection-idle-timeout duration Default: 4h0m0s

Maximum time a streaming connection can be idle before the connection is automatically closed. 0 indicates no timeout. Example: '5m'. Note: All connections to the kubelet server have a maximum duration of 4 hours. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--sync-frequency duration Default: 1m0s

Max period between synchronizing running containers and config (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--system-cgroups string

Optional absolute name of cgroups in which to place all non-kernel processes that are not already inside a cgroup under '/'. Empty for no container. Rolling back the flag requires a reboot. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--system-reserved <comma-separated 'key=value' pairs>

A set of ResourceName=ResourceQuantity (e.g. cpu=200m,memory=500Mi,ephemeral-storage=1Gi,pid=1000) pairs that describe resources reserved for non-kubernetes components. Currently only cpu, memory, pid and local ephemeral storage for root file system are supported. See https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/ for more detail. [default=none] (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--system-reserved-cgroup string

Absolute name of the top level cgroup that is used to manage non-kubernetes components for which compute resources were reserved via '--system-reserved' flag. Ex. '/system-reserved'. [default=''] (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--tls-cert-file string

File containing x509 Certificate used for serving HTTPS (with intermediate certs, if any, concatenated after server cert). If --tls-cert-file and --tls-private-key-file are not provided, a self-signed certificate and key are generated for the public address and saved to the directory passed to --cert-dir. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--tls-cipher-suites strings

Comma-separated list of cipher suites for the server. If omitted, the default Go cipher suites will be used.
Preferred values: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256.
Insecure values: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_ECDSA_WITH_RC4_128_SHA, TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_RSA_WITH_RC4_128_SHA, TLS_RSA_WITH_3DES_EDE_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_256_CBC_SHA, TLS_RSA_WITH_AES_256_GCM_SHA384, TLS_RSA_WITH_RC4_128_SHA. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--tls-min-version string

Minimum TLS version supported. Possible values: VersionTLS10, VersionTLS11, VersionTLS12, VersionTLS13 (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--tls-private-key-file string

File containing x509 private key matching --tls-cert-file. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--topology-manager-policy string Default: "none"

Topology Manager policy to use. Possible values: 'none', 'best-effort', 'restricted', 'single-numa-node'. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--topology-manager-policy-options <comma-separated 'key=value' pairs>

A set of key=value Topology Manager policy options to use, to fine tune their behaviour. If not supplied, keep the default behaviour. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--topology-manager-scope string Default: "container"

Scope to which topology hints applied. Topology Manager collects hints from Hint Providers and applies them to defined scope to ensure the pod admission. Possible values: 'container', 'pod'. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

-v, --v int

number for the log level verbosity

--version version[=true]

--version, --version=raw prints version information and quits; --version=vX.Y.Z... sets the reported version

--vmodule pattern=N,...

comma-separated list of pattern=N settings for file-filtered logging (only works for text log format)

--volume-plugin-dir string Default: "/usr/libexec/kubernetes/kubelet-plugins/volume/exec/"

The full path of the directory in which to search for additional third party volume plugins (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

--volume-stats-agg-period duration Default: 1m0s

Specifies interval for kubelet to calculate and cache the volume disk usage for all pods and volumes. To disable volume calculations, set to a negative number. (DEPRECATED: This parameter should be set via the config file specified by the Kubelet's --config flag. See https://kubernetes.io/docs/tasks/administer-cluster/kubelet-config-file/ for more information.)

Kubelet Configuration (v1)

Resource Types

- CredentialProviderConfig

`CredentialProviderConfig`

CredentialProviderConfig is the configuration containing information about each exec credential provider. Kubelet reads this configuration from disk and enables each provider as specified by the CredentialProvider type.

Field	Description
`apiVersion` string	`kubelet.config.k8s.io/v1`
`kind` string	`CredentialProviderConfig`
`providers` **[Required]** `[]CredentialProvider`	providers is a list of credential provider plugins that will be enabled by the kubelet. Multiple providers may match against a single image, in which case credentials from all providers will be returned to the kubelet. If multiple providers are called for a single image, the results are combined. If providers return overlapping auth keys, the value from the provider earlier in this list is attempted first.

`CredentialProvider`

Appears in:

- CredentialProviderConfig

CredentialProvider represents an exec plugin to be invoked by the kubelet. The plugin is only invoked when an image being pulled matches the images handled by the plugin (see matchImages).

Field	Description
`name` **[Required]** `string`	name is the required name of the credential provider. It must match the name of the provider executable as seen by the kubelet. The executable must be in the kubelet's bin directory (set by the --image-credential-provider-bin-dir flag). Required to be unique across all providers.
`matchImages` **[Required]** `[]string`	matchImages is a required list of strings used to match against images in order to determine if this provider should be invoked. If one of the strings matches the requested image from the kubelet, the plugin will be invoked and given a chance to provide credentials. Images are expected to contain the registry domain and URL path. Each entry in matchImages is a pattern which can optionally contain a port and a path. Globs can be used in the domain, but not in the port or the path. Globs are supported as subdomains like '*.k8s.io' or 'k8s*..io', and top-level-domains such as 'k8s.'. *Matching partial subdomains like 'app*.k8s.io' is also supported. Each glob can only match a single subdomain segment, so *.io does not match *.k8s.io. A match exists between an image and a matchImage when all of the below are true: • Both contain the same number of domain parts and each part matches. • The URL path of an imageMatch must be a prefix of the target image URL path. • If the imageMatch contains a port, then the port must match in the image as well. Example values of matchImages: • 123456789.dkr.ecr.us-east-1.amazonaws.com • *.azurecr.io • gcr.io • *.registry.io • registry.io:8080/path
`defaultCacheDuration` **[Required]** `meta/v1.Duration`	defaultCacheDuration is the default duration the plugin will cache credentials in-memory if a cache duration is not provided in the plugin response. This field is required.
`apiVersion` **[Required]** `string`	Required input version of the exec CredentialProviderRequest. The returned CredentialProviderResponse MUST use the same encoding version as the input. Current supported values are: • credentialprovider.kubelet.k8s.io/v1
`args` `[]string`	Arguments to pass to the command when executing it.
`env` `[]ExecEnvVar`	Env defines additional environment variables to expose to the process. These are unioned with the host's environment, as well as variables client-go uses to pass argument to the plugin.
`tokenAttributes` `ServiceAccountTokenAttributes`	tokenAttributes is the configuration for the service account token that will be passed to the plugin. The credential provider opts in to using service account tokens for image pull by setting this field. When this field is set, kubelet will generate a service account token bound to the pod for which the image is being pulled and pass to the plugin as part of CredentialProviderRequest along with other attributes required by the plugin. The service account metadata and token attributes will be used as a dimension to cache the credentials in kubelet. The cache key is generated by combining the service account metadata (namespace, name, UID, and annotations key+value for the keys defined in serviceAccountTokenAttribute.requiredServiceAccountAnnotationKeys and serviceAccountTokenAttribute.optionalServiceAccountAnnotationKeys). The pod metadata (namespace, name, UID) that are in the service account token are not used as a dimension to cache the credentials in kubelet. This means workloads that are using the same service account could end up using the same credentials for image pull. For plugins that don't want this behavior, or plugins that operate in pass-through mode; i.e., they return the service account token as-is, they can set the credentialProviderResponse.cacheDuration to 0. This will disable the caching of credentials in kubelet and the plugin will be invoked for every image pull. This does result in token generation overhead for every image pull, but it is the only way to ensure that the credentials are not shared across pods (even if they are using the same service account).

`ExecEnvVar`

Appears in:

- CredentialProvider

ExecEnvVar is used for setting environment variables when executing an exec-based credential plugin.

Field	Description
`name` **[Required]** `string`	No description provided.
`value` **[Required]** `string`	No description provided.

`ServiceAccountTokenAttributes`

Appears in:

- CredentialProvider

ServiceAccountTokenAttributes is the configuration for the service account token that will be passed to the plugin.

Field	Description
`serviceAccountTokenAudience` **[Required]** `string`	serviceAccountTokenAudience is the intended audience for the projected service account token.
`cacheType` **[Required]** `ServiceAccountTokenCacheType`	cacheType indicates the type of cache key use for caching the credentials returned by the plugin when the service account token is used. The most conservative option is to set this to "Token", which means the kubelet will cache returned credentials on a per-token basis. This should be set if the returned credential's lifetime is limited to the service account token's lifetime. If the plugin's credential retrieval logic depends only on the service account and not on pod-specific claims, then the plugin can set this to "ServiceAccount". In this case, the kubelet will cache returned credentials on a per-serviceaccount basis. Use this when the returned credential is valid for all pods using the same service account.
`requireServiceAccount` **[Required]** `bool`	requireServiceAccount indicates whether the plugin requires the pod to have a service account. If set to true, kubelet will only invoke the plugin if the pod has a service account. If set to false, kubelet will invoke the plugin even if the pod does not have a service account and will not include a token in the CredentialProviderRequest in that scenario. This is useful for plugins that are used to pull images for pods without service accounts (e.g., static pods).
`requiredServiceAccountAnnotationKeys` `[]string`	requiredServiceAccountAnnotationKeys is the list of annotation keys that the plugin is interested in and that are required to be present in the service account. The keys defined in this list will be extracted from the corresponding service account and passed to the plugin as part of the CredentialProviderRequest. If any of the keys defined in this list are not present in the service account, kubelet will not invoke the plugin and will return an error. This field is optional and may be empty. Plugins may use this field to extract additional information required to fetch credentials or allow workloads to opt in to using service account tokens for image pull. If non-empty, requireServiceAccount must be set to true. Keys in this list must be unique. This list needs to be mutually exclusive with optionalServiceAccountAnnotationKeys.
`optionalServiceAccountAnnotationKeys` `[]string`	optionalServiceAccountAnnotationKeys is the list of annotation keys that the plugin is interested in and that are optional to be present in the service account. The keys defined in this list will be extracted from the corresponding service account and passed to the plugin as part of the CredentialProviderRequest. The plugin is responsible for validating the existence of annotations and their values. This field is optional and may be empty. Plugins may use this field to extract additional information required to fetch credentials. Keys in this list must be unique.

`ServiceAccountTokenCacheType`

(Alias of `string`)

Appears in:

- ServiceAccountTokenAttributes

ServiceAccountTokenCacheType is the type of cache key used for caching credentials returned by the plugin when the service account token is used.

---

kubeconfig (v1)

Resource Types

- Config

`Config`

Config holds the information needed to build connect to remote kubernetes clusters as a given user

Field	Description
`apiVersion` string	`/v1`
`kind` string	`Config`
`kind` string	Legacy field from pkg/api/types.go TypeMeta. TODO(jlowdermilk): remove this after eliminating downstream dependencies.
`apiVersion` string	Legacy field from pkg/api/types.go TypeMeta. TODO(jlowdermilk): remove this after eliminating downstream dependencies.
`preferences,omitzero` **[Required]** Preferences	Preferences holds general information to be use for cli interactions Deprecated: this field is deprecated in v1.34. It is not used by any of the Kubernetes components.
`clusters` **[Required]** []NamedCluster	Clusters is a map of referenceable names to cluster configs
`users` **[Required]** []NamedAuthInfo	AuthInfos is a map of referenceable names to user configs
`contexts` **[Required]** []NamedContext	Contexts is a map of referenceable names to context configs
`current-context` **[Required]** string	CurrentContext is the name of the context that you would like to use by default
`extensions` []NamedExtension	Extensions holds additional information. This is useful for extenders so that reads and writes don't clobber unknown fields

AuthInfo

Appears in:

- NamedAuthInfo

AuthInfo contains information that describes identity information. This is use to tell the kubernetes cluster who you are.

Field	Description
`client-certificate` string	ClientCertificate is the path to a client cert file for TLS.
`client-certificate-data` []byte	ClientCertificateData contains PEM-encoded data from a client cert file for TLS. Overrides ClientCertificate
`client-key` string	ClientKey is the path to a client key file for TLS.
`client-key-data` []byte	ClientKeyData contains PEM-encoded data from a client key file for TLS. Overrides ClientKey
`token` string	Token is the bearer token for authentication to the kubernetes cluster.
`tokenFile` string	TokenFile is a pointer to a file that contains a bearer token (as described above). If both Token and TokenFile are present, the TokenFile will be periodically read and the last successfully read value takes precedence over Token.
`as` string	Impersonate is the username to impersonate. The name matches the flag.
`as-uid` string	ImpersonateUID is the uid to impersonate.
`as-groups` []string	ImpersonateGroups is the groups to impersonate.
`as-user-extra` map[string][]string	ImpersonateUserExtra contains additional information for impersonated user.
`username` string	Username is the username for basic authentication to the kubernetes cluster.
`password` string	Password is the password for basic authentication to the kubernetes cluster.
`auth-provider` AuthProviderConfig	AuthProvider specifies a custom authentication plugin for the kubernetes cluster.
`exec` ExecConfig	Exec specifies a custom exec-based authentication plugin for the kubernetes cluster.
`extensions` []NamedExtension	Extensions holds additional information. This is useful for extenders so that reads and writes don't clobber unknown fields

AuthProviderConfig

Appears in:

- AuthInfo

AuthProviderConfig holds the configuration for a specified auth provider.

Field	Description
`name` **[Required]** `string`	No description provided.
`config` **[Required]** `map[string]string`	No description provided.

`Cluster`

Appears in:

- NamedCluster

Cluster contains information about how to communicate with a kubernetes cluster

Field	Description
`server` **[Required]** `string`	Server is the address of the kubernetes cluster (https://hostname:port).
`tls-server-name` `string`	TLSServerName is used to check server certificate. If TLSServerName is empty, the hostname used to contact the server is used.
`insecure-skip-tls-verify` `bool`	InsecureSkipTLSVerify skips the validity check for the server's certificate. This will make your HTTPS connections insecure.
`certificate-authority` `string`	CertificateAuthority is the path to a cert file for the certificate authority.
`certificate-authority-data` `[]byte`	CertificateAuthorityData contains PEM-encoded certificate authority certificates. Overrides CertificateAuthority
`proxy-url` `string`	ProxyURL is the URL to the proxy to be used for all requests made by this client. URLs with "http", "https", and "socks5" schemes are supported. If this configuration is not provided or the empty string, the client attempts to construct a proxy configuration from http_proxy and https_proxy environment variables. If these environment variables are not set, the client does not attempt to proxy requests. socks5 proxying does not currently support spdy streaming endpoints (exec, attach, port forward).
`disable-compression` `bool`	DisableCompression allows client to opt-out of response compression for all requests to the server. This is useful to speed up requests (specifically lists) when client-server network bandwidth is ample, by saving time on compression (server-side) and decompression (client-side): https://github.com/kubernetes/kubernetes/issues/112296.
`extensions` `[]NamedExtension`	Extensions holds additional information. This is useful for extenders so that reads and writes don't clobber unknown fields

`Context`

Appears in:

- NamedContext

Context is a tuple of references to a cluster (how do I communicate with a kubernetes cluster), a user (how do I identify myself), and a namespace (what subset of resources do I want to work with)

Field	Description
`cluster` **[Required]** `string`	Cluster is the name of the cluster for this context
`user` **[Required]** `string`	AuthInfo is the name of the authInfo for this context
`namespace` `string`	Namespace is the default namespace to use on unspecified requests
`extensions` `[]NamedExtension`	Extensions holds additional information. This is useful for extenders so that reads and writes don't clobber unknown fields

`ExecConfig`

Appears in:

- AuthInfo

ExecConfig specifies a command to provide client credentials. The command is exec'd and outputs structured stdout holding credentials.

See the client.authentication.k8s.io API group for specifications of the exact input and output format

Field	Description
`command` **[Required]** `string`	Command to execute.
`args` `[]string`	Arguments to pass to the command when executing it.
`env` `[]ExecEnvVar`	Env defines additional environment variables to expose to the process. These are unioned with the host's environment, as well as variables client-go uses to pass argument to the plugin.
`apiVersion` **[Required]** `string`	Preferred input version of the ExecInfo. The returned ExecCredentials MUST use the same encoding version as the input.
`installHint` **[Required]** `string`	This text is shown to the user when the executable doesn't seem to be present. For example, `brew install foo-cli` might be a good InstallHint for foo-cli on Mac OS systems.
`provideClusterInfo` **[Required]** `bool`	ProvideClusterInfo determines whether or not to provide cluster information, which could potentially contain very large CA data, to this exec plugin as a part of the KUBERNETES_EXEC_INFO environment variable. By default, it is set to false. Package k8s.io/client-go/tools/auth/exec provides helper methods for reading this environment variable.
`interactiveMode` `ExecInteractiveMode`	InteractiveMode determines this plugin's relationship with standard input. Valid values are "Never" (this exec plugin never uses standard input), "IfAvailable" (this exec plugin wants to use standard input if it is available), or "Always" (this exec plugin requires standard input to function). See ExecInteractiveMode values for more details. If APIVersion is client.authentication.k8s.io/v1alpha1 or client.authentication.k8s.io/v1beta1, then this field is optional and defaults to "IfAvailable" when unset. Otherwise, this field is required.

`ExecEnvVar`

Appears in:

- ExecConfig

ExecEnvVar is used for setting environment variables when executing an exec-based credential plugin.

Field	Description
`name` **[Required]** `string`	No description provided.
`value` **[Required]** `string`	No description provided.

`ExecInteractiveMode`

(Alias of `string`)

Appears in:

- ExecConfig

ExecInteractiveMode is a string that describes an exec plugin's relationship with standard input.

`NamedAuthInfo`

Appears in:

- Config

NamedAuthInfo relates nicknames to auth information

Field	Description
`name` **[Required]** `string`	Name is the nickname for this AuthInfo
`user` **[Required]** `AuthInfo`	AuthInfo holds the auth information

`NamedCluster`

Appears in:

- Config

NamedCluster relates nicknames to cluster information

Field	Description
`name` **[Required]** `string`	Name is the nickname for this Cluster
`cluster` **[Required]** `Cluster`	Cluster holds the cluster information

`NamedContext`

Appears in:

- Config

NamedContext relates nicknames to context information

Field	Description
`name` **[Required]** `string`	Name is the nickname for this Context
`context` **[Required]** `Context`	Context holds the context information

`NamedExtension`

Appears in:

- Config
- AuthInfo
- Cluster
- Context
- Preferences

NamedExtension relates nicknames to extension information

Field	Description
`name` **[Required]** `string`	Name is the nickname for this Extension
`extension` **[Required]** `k8s.io/apimachinery/pkg/runtime.RawExtension`	Extension holds the extension information

`Preferences`

Appears in:

- Config

Deprecated: this structure is deprecated in v1.34. It is not used by any of the Kubernetes components.

Field	Description
`colors` `bool`	No description provided.
`extensions` `[]NamedExtension`	Extensions holds additional information. This is useful for extenders so that reads and writes don't clobber unknown fields

---

Kubelet CredentialProvider (v1)

Resource Types

- CredentialProviderRequest
- CredentialProviderResponse

`CredentialProviderRequest`

CredentialProviderRequest includes the image that the kubelet requires authentication for. Kubelet will pass this request object to the plugin via stdin. In general, plugins should prefer responding with the same apiVersion they were sent.

Field	Description
`apiVersion` string	`credentialprovider.kubelet.k8s.io/v1`
`kind` string	`CredentialProviderRequest`
`image` **[Required]** `string`	image is the container image that is being pulled as part of the credential provider plugin request. Plugins may optionally parse the image to extract any information required to fetch credentials.
`serviceAccountToken` **[Required]** `string`	serviceAccountToken is the service account token bound to the pod for which the image is being pulled. This token is only sent to the plugin if the tokenAttributes.serviceAccountTokenAudience field is configured in the kubelet's credential provider configuration.
`serviceAccountAnnotations` **[Required]** `map[string]string`	serviceAccountAnnotations is a map of annotations on the service account bound to the pod for which the image is being pulled. The list of annotations in the service account that need to be passed to the plugin is configured in the kubelet's credential provider configuration.

`CredentialProviderResponse`

CredentialProviderResponse holds credentials that the kubelet should use for the specified image provided in the original request. Kubelet will read the response from the plugin via stdout. This response should be set to the same apiVersion as CredentialProviderRequest.

Field	Description
`apiVersion` string	`credentialprovider.kubelet.k8s.io/v1`
`kind` string	`CredentialProviderResponse`
`cacheKeyType` **[Required]** `PluginCacheKeyType`	cacheKeyType indicates the type of caching key to use based on the image provided in the request. There are three valid values for the cache key type: Image, Registry, and Global. If an invalid value is specified, the response will NOT be used by the kubelet.
`cacheDuration` `meta/v1.Duration`	cacheDuration indicates the duration the provided credentials should be cached for. The kubelet will use this field to set the in-memory cache duration for credentials in the AuthConfig. If null, the kubelet will use defaultCacheDuration provided in CredentialProviderConfig. If set to 0, the kubelet will not cache the provided AuthConfig.
`auth` `map[string]AuthConfig`	auth is a map containing authentication information passed into the kubelet. Each key is a match image string (more on this below). The corresponding authConfig value should be valid for all images that match against this key. A plugin should set this field to null if no valid credentials can be returned for the requested image. Each key in the map is a pattern which can optionally contain a port and a path. Globs can be used in the domain, but not in the port or the path. Globs are supported as subdomains like '*.k8s.io' or 'k8s*..io', and top-level-domains such as 'k8s.*'. Matching partial subdomains like '*app*.k8s.io' is also supported. Each glob can only match a single subdomain segment, so *.io does not match *.k8s.io. The kubelet will match images against the key when all of the below are true: • Both contain the same number of domain parts and each part matches. • The URL path of an imageMatch must be a prefix of the target image URL path. • If the imageMatch contains a port, then the port must match in the image as well. When multiple keys are returned, the kubelet will traverse all keys in reverse order so that: • longer keys come before shorter keys with the same prefix • non-wildcard keys come before wildcard keys with the same prefix. For any given match, the kubelet will attempt an image pull with the provided credentials, stopping after the first successfully authenticated pull. Example keys: • 123456789.dkr.ecr.us-east-1.amazonaws.com • *.azurecr.io • gcr.io • *.registry.io • registry.io:8080/path

`AuthConfig`

Appears in:

- CredentialProviderResponse

AuthConfig contains authentication information for a container registry. Only username/password based authentication is supported today, but more authentication mechanisms may be added in the future.

Field	Description
`username` **[Required]** `string`	username is the username used for authenticating to the container registry An empty username is valid.
`password` **[Required]** `string`	password is the password used for authenticating to the container registry An empty password is valid.

`PluginCacheKeyType`

(Alias of `string`)

Appears in:

- CredentialProviderResponse

---

kube-controller-manager Configuration (v1alpha1)

Resource Types

- CloudControllerManagerConfiguration
- LeaderMigrationConfiguration
- KubeControllerManagerConfiguration

`ClientConnectionConfiguration`

Appears in:

- KubeSchedulerConfiguration
- GenericControllerManagerConfiguration

ClientConnectionConfiguration contains details for constructing a client.

Field	Description
`kubeconfig` **[Required]** `string`	kubeconfig is the path to a KubeConfig file.
`acceptContentTypes` **[Required]** `string`	acceptContentTypes defines the Accept header sent by clients when connecting to a server, overriding the default value of 'application/json'. This field will control all connections to the server used by a particular client.
`contentType` **[Required]** `string`	contentType is the content type used when sending data to the server from this client.
`qps` **[Required]** `float32`	qps controls the number of queries per second allowed for this connection.
`burst` **[Required]** `int32`	burst allows extra queries to accumulate when a client is exceeding its rate.

`DebuggingConfiguration`

Appears in:

- KubeSchedulerConfiguration
- GenericControllerManagerConfiguration

DebuggingConfiguration holds configuration for Debugging related features.

Field	Description
`enableProfiling` **[Required]** `bool`	enableProfiling enables profiling via web interface host:port/debug/pprof/
`enableContentionProfiling` **[Required]** `bool`	enableContentionProfiling enables block profiling, if enableProfiling is true.

LeaderElectionConfiguration

Appears in:

- KubeSchedulerConfiguration
- GenericControllerManagerConfiguration

LeaderElectionConfiguration defines the configuration of leader election clients for components that can run with leader election enabled.

Field	Description
`leaderElect` **[Required]** `bool`	leaderElect enables a leader election client to gain leadership before executing the main loop. Enable this when running replicated components for high availability.
`leaseDuration` **[Required]** `meta/v1.Duration`	leaseDuration is the duration that non-leader candidates will wait after observing a leadership renewal until attempting to acquire leadership of a led but unrenewed leader slot. This is effectively the maximum duration that a leader can be stopped before it is replaced by another candidate. This is only applicable if leader election is enabled.
`renewDeadline` **[Required]** `meta/v1.Duration`	renewDeadline is the interval between attempts by the acting master to renew a leadership slot before it stops leading. This must be less than or equal to the lease duration. This is only applicable if leader election is enabled.
`retryPeriod` **[Required]** `meta/v1.Duration`	retryPeriod is the duration the clients should wait between attempting acquisition and renewal of a leadership. This is only applicable if leader election is enabled.
`resourceLock` **[Required]** `string`	resourceLock indicates the resource object type that will be used to lock during leader election cycles.
`resourceName` **[Required]** `string`	resourceName indicates the name of resource object that will be used to lock during leader election cycles.
`resourceNamespace` **[Required]** `string`	resourceName indicates the namespace of resource object that will be used to lock during leader election cycles.

NodeControllerConfiguration

Appears in:

- CloudControllerManagerConfiguration

NodeControllerConfiguration contains elements describing NodeController.

Field	Description
`ConcurrentNodeSyncs` **[Required]** `int32`	ConcurrentNodeSyncs is the number of workers concurrently synchronizing nodes

ServiceControllerConfiguration

Appears in:

- CloudControllerManagerConfiguration
- KubeControllerManagerConfiguration

ServiceControllerConfiguration contains elements describing ServiceController.

Field	Description
`ConcurrentServiceSyncs` **[Required]** `int32`	concurrentServiceSyncs is the number of services that are allowed to sync concurrently. Larger number = more responsive service management, but more CPU (and network) load.

CloudControllerManagerConfiguration

CloudControllerManagerConfiguration contains elements describing cloud-controller manager.

Field	Description
`apiVersion` string	`cloudcontrollermanager.config.k8s.io/v1alpha1`
`kind` string	`CloudControllerManagerConfiguration`
`Generic` **[Required]** `GenericControllerManagerConfiguration`	Generic holds configuration for a generic controller-manager
`KubeCloudShared` **[Required]** `KubeCloudSharedConfiguration`	KubeCloudSharedConfiguration holds configuration for shared related features both in cloud controller manager and kube-controller manager.

Field	Description
`NodeController` **[Required]** NodeControllerConfiguration	NodeController holds configuration for node controller related features.
`ServiceController` **[Required]** ServiceControllerConfiguration	ServiceControllerConfiguration holds configuration for ServiceController related features.
`NodeStatusUpdateFrequency` **[Required]** meta/v1.Duration	NodeStatusUpdateFrequency is the frequency at which the controller updates nodes' status
`Webhook` **[Required]** WebhookConfiguration	Webhook is the configuration for cloud-controller-manager hosted webhooks

`CloudProviderConfiguration`

Appears in:

- KubeCloudSharedConfiguration

CloudProviderConfiguration contains basically elements about cloud provider.

Field	Description
`Name` **[Required]** `string`	Name is the provider for cloud services.
`CloudConfigFile` **[Required]** `string`	cloudConfigFile is the path to the cloud provider configuration file.

`KubeCloudSharedConfiguration`

Appears in:

- CloudControllerManagerConfiguration
- KubeControllerManagerConfiguration

KubeCloudSharedConfiguration contains elements shared by both kube-controller manager and cloud-controller manager, but not genericconfig.

Field	Description
`CloudProvider` **[Required]** CloudProviderConfiguration	CloudProviderConfiguration holds configuration for CloudProvider related features.
`ExternalCloudVolumePlugin` **[Required]** `string`	externalCloudVolumePlugin specifies the plugin to use when cloudProvider is "external". It is currently used by the in repo cloud providers to handle node and volume control in the KCM.
`UseServiceAccountCredentials` **[Required]** `bool`	useServiceAccountCredentials indicates whether controllers should be run with individual service account credentials.
`AllowUntaggedCloud` **[Required]** `bool`	run with untagged cloud instances
`RouteReconciliationPeriod` **[Required]** meta/v1.Duration	routeReconciliationPeriod is the period for reconciling routes created for Nodes by cloud provider..
`NodeMonitorPeriod` **[Required]** meta/v1.Duration	nodeMonitorPeriod is the period for syncing NodeStatus in NodeController.
`ClusterName` **[Required]** `string`	clusterName is the instance prefix for the cluster.
`ClusterCIDR` **[Required]** `string`	clusterCIDR is CIDR Range for Pods in cluster.
`AllocateNodeCIDRs` **[Required]** `bool`	AllocateNodeCIDRs enables CIDRs for Pods to be allocated and, if ConfigureCloudRoutes is true, to be set on the cloud provider.
`CIDRAllocatorType` **[Required]** `string`	CIDRAllocatorType determines what kind of pod CIDR allocator will be used.
`ConfigureCloudRoutes` **[Required]** `bool`	configureCloudRoutes enables CIDRs allocated with allocateNodeCIDRs to be configured on the cloud provider.
`NodeSyncPeriod` **[Required]** meta/v1.Duration	nodeSyncPeriod is the period for syncing nodes from cloudprovider. Longer periods will result in fewer calls to cloud provider, but may delay addition of new nodes to cluster.

`WebhookConfiguration`

Appears in:

- CloudControllerManagerConfiguration

WebhookConfiguration contains configuration related to cloud-controller-manager hosted webhooks

Field	Description
`Webhooks` **[Required]** `[]string`	Webhooks is the list of webhooks to enable or disable '*' means "all enabled by default webhooks" 'foo' means "enable 'foo'" '-foo' means "disable 'foo'" first item for a particular name wins

`LeaderMigrationConfiguration`

Appears in:

- GenericControllerManagerConfiguration

LeaderMigrationConfiguration provides versioned configuration for all migrating leader locks.

Field	Description
`apiVersion` string	`controllermanager.config.k8s.io/v1alpha1`
`kind` string	`LeaderMigrationConfiguration`
`leaderName` **[Required]** `string`	LeaderName is the name of the leader election resource that protects the migration E.g. 1-20-KCM-to-1-21-CCM
`resourceLock` **[Required]** `string`	ResourceLock indicates the resource object type that will be used to lock Should be "leases" or "endpoints"
`controllerLeaders` **[Required]** `[]ControllerLeaderConfiguration`	ControllerLeaders contains a list of migrating leader lock configurations

`ControllerLeaderConfiguration`

Appears in:

- LeaderMigrationConfiguration

ControllerLeaderConfiguration provides the configuration for a migrating leader lock.

Field	Description
`name` **[Required]** `string`	Name is the name of the controller being migrated E.g. service-controller, route-controller, cloud-node-controller, etc
`component` **[Required]** `string`	Component is the name of the component in which the controller should be running. E.g. kube-controller-manager, cloud-controller-manager, etc Or '*' meaning the controller can be run under any component that participates in the migration

`GenericControllerManagerConfiguration`

Appears in:

- CloudControllerManagerConfiguration
- KubeControllerManagerConfiguration

GenericControllerManagerConfiguration holds configuration for a generic controller-manager.

Field	Description
`Port` **[Required]** `int32`	port is the port that the controller-manager's http service runs on.
`Address` **[Required]** `string`	address is the IP address to serve on (set to 0.0.0.0 for all interfaces).
`MinResyncPeriod` **[Required]** `meta/v1.Duration`	minResyncPeriod is the resync period in reflectors; will be random between minResyncPeriod and 2*minResyncPeriod.
`ClientConnection` **[Required]** `ClientConnectionConfiguration`	ClientConnection specifies the kubeconfig file and client connection settings for the proxy server to use when communicating with the apiserver.
`ControllerStartInterval` **[Required]** `meta/v1.Duration`	How long to wait between starting controller managers
`LeaderElection` **[Required]** `LeaderElectionConfiguration`	leaderElection defines the configuration of leader election client.
`Controllers` **[Required]** `[]string`	Controllers is the list of controllers to enable or disable '*' means "all enabled by default controllers" 'foo' means "enable 'foo'" '-foo' means "disable 'foo'" first item for a particular name wins

Field	Description
`Debugging` **[Required]** `DebuggingConfiguration`	DebuggingConfiguration holds configuration for Debugging related features.
`LeaderMigrationEnabled` **[Required]** `bool`	LeaderMigrationEnabled indicates whether Leader Migration should be enabled for the controller manager.
`LeaderMigration` **[Required]** `LeaderMigrationConfiguration`	LeaderMigration holds the configuration for Leader Migration.

`KubeControllerManagerConfiguration`

KubeControllerManagerConfiguration contains elements describing kube-controller manager.

Field	Description
`apiVersion` string	`kubecontrollermanager.config.k8s.io/v1alpha1`
`kind` string	`KubeControllerManagerConfiguration`
`Generic` **[Required]** `GenericControllerManagerConfiguration`	Generic holds configuration for a generic controller-manager
`KubeCloudShared` **[Required]** `KubeCloudSharedConfiguration`	KubeCloudSharedConfiguration holds configuration for shared related features both in cloud controller manager and kube-controller manager.
`AttachDetachController` **[Required]** `AttachDetachControllerConfiguration`	AttachDetachControllerConfiguration holds configuration for AttachDetachController related features.
`CSRSigningController` **[Required]** `CSRSigningControllerConfiguration`	CSRSigningControllerConfiguration holds configuration for CSRSigningController related features.
`DaemonSetController` **[Required]** `DaemonSetControllerConfiguration`	DaemonSetControllerConfiguration holds configuration for DaemonSetController related features.
`DeploymentController` **[Required]** `DeploymentControllerConfiguration`	DeploymentControllerConfiguration holds configuration for DeploymentController related features.
`StatefulSetController` **[Required]** `StatefulSetControllerConfiguration`	StatefulSetControllerConfiguration holds configuration for StatefulSetController related features.
`DeprecatedController` **[Required]** `DeprecatedControllerConfiguration`	DeprecatedControllerConfiguration holds configuration for some deprecated features.
`EndpointController` **[Required]** `EndpointControllerConfiguration`	EndpointControllerConfiguration holds configuration for EndpointController related features.
`EndpointSliceController` **[Required]** `EndpointSliceControllerConfiguration`	EndpointSliceControllerConfiguration holds configuration for EndpointSliceController related features.
`EndpointSliceMirroringController` **[Required]** `EndpointSliceMirroringControllerConfiguration`	EndpointSliceMirroringControllerConfiguration holds configuration for EndpointSliceMirroringController related features.
`EphemeralVolumeController` **[Required]** `EphemeralVolumeControllerConfiguration`	EphemeralVolumeControllerConfiguration holds configuration for EphemeralVolumeController related features.
`GarbageCollectorController` **[Required]** `GarbageCollectorControllerConfiguration`	GarbageCollectorControllerConfiguration holds configuration for GarbageCollectorController related features.
`HPAController` **[Required]** `HPAControllerConfiguration`	HPAControllerConfiguration holds configuration for HPAController related features.
`JobController` **[Required]** `JobControllerConfiguration`	JobControllerConfiguration holds configuration for JobController related features.
`CronJobController` **[Required]** `CronJobControllerConfiguration`	CronJobControllerConfiguration holds configuration for CronJobController related features.
`LegacySATokenCleaner` **[Required]** `LegacySATokenCleanerConfiguration`	LegacySATokenCleanerConfiguration holds configuration for LegacySATokenCleaner related features.
`NamespaceController` **[Required]** `NamespaceControllerConfiguration`	NamespaceControllerConfiguration holds configuration for NamespaceController related features.
`NodeIPAMController` **[Required]** `NodeIPAMControllerConfiguration`	NodeIPAMControllerConfiguration holds configuration for NodeIPAMController related features.
`NodeLifecycleController` **[Required]** `NodeLifecycleControllerConfiguration`	NodeLifecycleControllerConfiguration holds configuration for NodeLifecycleController related features.
`PersistentVolumeBinderController` **[Required]** `PersistentVolumeBinderControllerConfiguration`	PersistentVolumeBinderControllerConfiguration holds configuration for PersistentVolumeBinderController related features.
`PodGCController` **[Required]** `PodGCControllerConfiguration`	PodGCControllerConfiguration holds configuration for PodGCController related features.
`ReplicaSetController` **[Required]** `ReplicaSetControllerConfiguration`	ReplicaSetControllerConfiguration holds configuration for ReplicaSet related features.
`ReplicationController` **[Required]** `ReplicationControllerConfiguration`	ReplicationControllerConfiguration holds configuration for ReplicationController related features.

Field	Description
`ResourceQuotaController` **[Required]** ResourceQuotaControllerConfiguration	ResourceQuotaControllerConfiguration holds configuration for ResourceQuotaController related features.
`SAController` **[Required]** SAControllerConfiguration	SAControllerConfiguration holds configuration for ServiceAccountController related features.
`ServiceController` **[Required]** ServiceControllerConfiguration	ServiceControllerConfiguration holds configuration for ServiceController related features.
`TTLAfterFinishedController` **[Required]** TTLAfterFinishedControllerConfiguration	TTLAfterFinishedControllerConfiguration holds configuration for TTLAfterFinishedController related features.
`ValidatingAdmissionPolicyStatusController` **[Required]** ValidatingAdmissionPolicyStatusControllerConfiguration	ValidatingAdmissionPolicyStatusControllerConfiguration holds configuration for ValidatingAdmissionPolicyStatusController related features.

AttachDetachControllerConfiguration

Appears in:

- KubeControllerManagerConfiguration

AttachDetachControllerConfiguration contains elements describing AttachDetachController.

Field	Description
`DisableAttachDetachReconcilerSync` **[Required]** `bool`	Reconciler runs a periodic loop to reconcile the desired state of the with the actual state of the world by triggering attach detach operations. This flag enables or disables reconcile. Is false by default, and thus enabled.
`ReconcilerSyncLoopPeriod` **[Required]** meta/v1.Duration	ReconcilerSyncLoopPeriod is the amount of time the reconciler sync states loop wait between successive executions. Is set to 60 sec by default.
`disableForceDetachOnTimeout` **[Required]** `bool`	DisableForceDetachOnTimeout disables force detach when the maximum unmount time is exceeded. Is false by default, and thus force detach on unmount is enabled.

CSRSigningConfiguration

Appears in:

- CSRSigningControllerConfiguration

CSRSigningConfiguration holds information about a particular CSR signer

Field	Description
`CertFile` **[Required]** `string`	certFile is the filename containing a PEM-encoded X509 CA certificate used to issue certificates
`KeyFile` **[Required]** `string`	keyFile is the filename containing a PEM-encoded RSA or ECDSA private key used to issue certificates

CSRSigningControllerConfiguration

Appears in:

- KubeControllerManagerConfiguration

CSRSigningControllerConfiguration contains elements describing CSRSigningController.

Field	Description
`ClusterSigningCertFile` **[Required]** `string`	clusterSigningCertFile is the filename containing a PEM-encoded X509 CA certificate used to issue cluster-scoped certificates
`ClusterSigningKeyFile` **[Required]** `string`	clusterSigningCertFile is the filename containing a PEM-encoded RSA or ECDSA private key used to issue cluster-scoped certificates
`KubeletServingSignerConfiguration` **[Required]** CSRSigningConfiguration	kubeletServingSignerConfiguration holds the certificate and key used to issue certificates for the kubernetes.io/kubelet-serving signer
`KubeletClientSignerConfiguration` **[Required]** CSRSigningConfiguration	kubeletClientSignerConfiguration holds the certificate and key used to issue certificates for the kubernetes.io/kube-apiserver-client-kubelet
`KubeAPIServerClientSignerConfiguration` **[Required]** CSRSigningConfiguration	kubeAPIServerClientSignerConfiguration holds the certificate and key used to issue certificates for the kubernetes.io/kube-apiserver-client

Field	Description
`LegacyUnknownSignerConfiguration` **[Required]** `CSRSigningConfiguration`	legacyUnknownSignerConfiguration holds the certificate and key used to issue certificates for the kubernetes.io/legacy-unknown
`ClusterSigningDuration` **[Required]** `meta/v1.Duration`	clusterSigningDuration is the max length of duration signed certificates will be given. Individual CSRs may request shorter certs by setting spec.expirationSeconds.

`CronJobControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

CronJobControllerConfiguration contains elements describing CrongJob2Controller.

Field	Description
`ConcurrentCronJobSyncs` **[Required]** `int32`	concurrentCronJobSyncs is the number of job objects that are allowed to sync concurrently. Larger number = more responsive jobs, but more CPU (and network) load.

`DaemonSetControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

DaemonSetControllerConfiguration contains elements describing DaemonSetController.

Field	Description
`ConcurrentDaemonSetSyncs` **[Required]** `int32`	concurrentDaemonSetSyncs is the number of daemonset objects that are allowed to sync concurrently. Larger number = more responsive daemonset, but more CPU (and network) load.

`DeploymentControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

DeploymentControllerConfiguration contains elements describing DeploymentController.

Field	Description
`ConcurrentDeploymentSyncs` **[Required]** `int32`	concurrentDeploymentSyncs is the number of deployment objects that are allowed to sync concurrently. Larger number = more responsive deployments, but more CPU (and network) load.

`DeprecatedControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

DeprecatedControllerConfiguration contains elements be deprecated.

`EndpointControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

EndpointControllerConfiguration contains elements describing EndpointController.

Field	Description
`ConcurrentEndpointSyncs` **[Required]** `int32`	concurrentEndpointSyncs is the number of endpoint syncing operations that will be done concurrently. Larger number = faster endpoint updating, but more CPU (and network) load.
`EndpointUpdatesBatchPeriod` **[Required]** `meta/v1.Duration`	EndpointUpdatesBatchPeriod describes the length of endpoint updates batching period. Processing of pod changes will be delayed by this duration to join them with potential upcoming updates and reduce the overall number of endpoints updates.

`EndpointSliceControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

EndpointSliceControllerConfiguration contains elements describing EndpointSliceController.

Field	Description
`ConcurrentServiceEndpointSyncs` **[Required]** `int32`	concurrentServiceEndpointSyncs is the number of service endpoint syncing operations that will be done concurrently. Larger number = faster endpoint slice updating, but more CPU (and network) load.
`MaxEndpointsPerSlice` **[Required]** `int32`	maxEndpointsPerSlice is the maximum number of endpoints that will be added to an EndpointSlice. More endpoints per slice will result in fewer and larger endpoint slices, but larger resources.
`EndpointUpdatesBatchPeriod` **[Required]** `meta/v1.Duration`	EndpointUpdatesBatchPeriod describes the length of endpoint updates batching period. Processing of pod changes will be delayed by this duration to join them with potential upcoming updates and reduce the overall number of endpoints updates.

`EndpointSliceMirroringControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

EndpointSliceMirroringControllerConfiguration contains elements describing EndpointSliceMirroringController.

Field	Description
`MirroringConcurrentServiceEndpointSyncs` **[Required]** `int32`	mirroringConcurrentServiceEndpointSyncs is the number of service endpoint syncing operations that will be done concurrently. Larger number = faster endpoint slice updating, but more CPU (and network) load.
`MirroringMaxEndpointsPerSubset` **[Required]** `int32`	mirroringMaxEndpointsPerSubset is the maximum number of endpoints that will be mirrored to an EndpointSlice for an EndpointSubset.
`MirroringEndpointUpdatesBatchPeriod` **[Required]** `meta/v1.Duration`	mirroringEndpointUpdatesBatchPeriod can be used to batch EndpointSlice updates. All updates triggered by EndpointSlice changes will be delayed by up to 'mirroringEndpointUpdatesBatchPeriod'. If other addresses in the same Endpoints resource change in that period, they will be batched to a single EndpointSlice update. Default 0 value means that each Endpoints update triggers an EndpointSlice update.

`EphemeralVolumeControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

EphemeralVolumeControllerConfiguration contains elements describing EphemeralVolumeController.

Field	Description
`ConcurrentEphemeralVolumeSyncs` **[Required]** `int32`	ConcurrentEphemeralVolumeSyncseSyncs is the number of ephemeral volume syncing operations that will be done concurrently. Larger number = faster ephemeral volume updating, but more CPU (and network) load.

`GarbageCollectorControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

GarbageCollectorControllerConfiguration contains elements describing GarbageCollectorController.

Field	Description
`EnableGarbageCollector` **[Required]** `bool`	enables the generic garbage collector. MUST be synced with the corresponding flag of the kube-apiserver. WARNING: the generic garbage collector is an alpha feature.
`ConcurrentGCSyncs` **[Required]** `int32`	concurrentGCSyncs is the number of garbage collector workers that are allowed to sync concurrently.
`GCIgnoredResources` **[Required]** `[]GroupResource`	gcIgnoredResources is the list of GroupResources that garbage collection should ignore.

`GroupResource`

Appears in:

- GarbageCollectorControllerConfiguration

GroupResource describes an group resource.

Field	Description
`Group` **[Required]** `string`	group is the group portion of the GroupResource.
`Resource` **[Required]** `string`	resource is the resource portion of the GroupResource.

`HPAControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

HPAControllerConfiguration contains elements describing HPAController.

Field	Description
`ConcurrentHorizontalPodAutoscalerSyncs` **[Required]** `int32`	ConcurrentHorizontalPodAutoscalerSyncs is the number of HPA objects that are allowed to sync concurrently. Larger number = more responsive HPA processing, but more CPU (and network) load.
`HorizontalPodAutoscalerSyncPeriod` **[Required]** `meta/v1.Duration`	HorizontalPodAutoscalerSyncPeriod is the period for syncing the number of pods in horizontal pod autoscaler.
`HorizontalPodAutoscalerDownscaleStabilizationWindow` **[Required]** `meta/v1.Duration`	HorizontalPodAutoscalerDowncaleStabilizationWindow is a period for which autoscaler will look backwards and not scale down below any recommendation it made during that period.
`HorizontalPodAutoscalerTolerance` **[Required]** `float64`	HorizontalPodAutoscalerTolerance is the tolerance for when resource usage suggests upscaling/downscaling
`HorizontalPodAutoscalerCPUInitializationPeriod` **[Required]** `meta/v1.Duration`	HorizontalPodAutoscalerCPUInitializationPeriod is the period after pod start when CPU samples might be skipped.
`HorizontalPodAutoscalerInitialReadinessDelay` **[Required]** `meta/v1.Duration`	HorizontalPodAutoscalerInitialReadinessDelay is period after pod start during which readiness changes are treated as readiness being set for the first time. The only effect of this is that HPA will disregard CPU samples from unready pods that had last readiness change during that period.

`JobControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

JobControllerConfiguration contains elements describing JobController.

Field	Description
`ConcurrentJobSyncs` **[Required]** `int32`	concurrentJobSyncs is the number of job objects that are allowed to sync concurrently. Larger number = more responsive jobs, but more CPU (and network) load.

`LegacySATokenCleanerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

LegacySATokenCleanerConfiguration contains elements describing LegacySATokenCleaner

Field	Description
`CleanUpPeriod` **[Required]** `meta/v1.Duration`	CleanUpPeriod is the period of time since the last usage of an auto-generated service account token before it can be deleted.

`NamespaceControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

NamespaceControllerConfiguration contains elements describing NamespaceController.

Field	Description
`NamespaceSyncPeriod` **[Required]** meta/v1.Duration	namespaceSyncPeriod is the period for syncing namespace life-cycle updates.
`ConcurrentNamespaceSyncs` **[Required]** `int32`	concurrentNamespaceSyncs is the number of namespace objects that are allowed to sync concurrently.

`NodeIPAMControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

NodeIPAMControllerConfiguration contains elements describing NodeIpamController.

Field	Description
`ServiceCIDR` **[Required]** `string`	serviceCIDR is CIDR Range for Services in cluster.
`SecondaryServiceCIDR` **[Required]** `string`	secondaryServiceCIDR is CIDR Range for Services in cluster. This is used in dual stack clusters. SecondaryServiceCIDR must be of different IP family than ServiceCIDR
`NodeCIDRMaskSize` **[Required]** `int32`	NodeCIDRMaskSize is the mask size for node cidr in cluster.
`NodeCIDRMaskSizeIPv4` **[Required]** `int32`	NodeCIDRMaskSizeIPv4 is the mask size for node cidr in dual-stack cluster.
`NodeCIDRMaskSizeIPv6` **[Required]** `int32`	NodeCIDRMaskSizeIPv6 is the mask size for node cidr in dual-stack cluster.

`NodeLifecycleControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

NodeLifecycleControllerConfiguration contains elements describing NodeLifecycleController.

Field	Description
`NodeEvictionRate` **[Required]** `float32`	nodeEvictionRate is the number of nodes per second on which pods are deleted in case of node failure when a zone is healthy
`SecondaryNodeEvictionRate` **[Required]** `float32`	secondaryNodeEvictionRate is the number of nodes per second on which pods are deleted in case of node failure when a zone is unhealthy
`NodeStartupGracePeriod` **[Required]** meta/v1.Duration	nodeStartupGracePeriod is the amount of time which we allow starting a node to be unresponsive before marking it unhealthy.
`NodeMonitorGracePeriod` **[Required]** meta/v1.Duration	nodeMontiorGracePeriod is the amount of time which we allow a running node to be unresponsive before marking it unhealthy. Must be N times more than kubelet's nodeStatusUpdateFrequency, where N means number of retries allowed for kubelet to post node status. This value should also be greater than the sum of HTTP2_PING_TIMEOUT_SECONDS and HTTP2_READ_IDLE_TIMEOUT_SECONDS.
`PodEvictionTimeout` **[Required]** meta/v1.Duration	podEvictionTimeout is the grace period for deleting pods on failed nodes.
`LargeClusterSizeThreshold` **[Required]** `int32`	secondaryNodeEvictionRate is implicitly overridden to 0 for clusters smaller than or equal to largeClusterSizeThreshold
`UnhealthyZoneThreshold` **[Required]** `float32`	Zone is treated as unhealthy in nodeEvictionRate and secondaryNodeEvictionRate when at least unhealthyZoneThreshold (no less than 3) of Nodes in the zone are NotReady

`PersistentVolumeBinderControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

PersistentVolumeBinderControllerConfiguration contains elements describing PersistentVolumeBinderController.

Field	Description
`PVClaimBinderSyncPeriod` **[Required]** meta/v1.Duration	pvClaimBinderSyncPeriod is the period for syncing persistent volumes and persistent volume claims.

Field	Description
`VolumeConfiguration` **[Required]** [`VolumeConfiguration`](#)	volumeConfiguration holds configuration for volume related features.

`PersistentVolumeRecyclerConfiguration`

Appears in:

- VolumeConfiguration

PersistentVolumeRecyclerConfiguration contains elements describing persistent volume plugins.

Field	Description
`MaximumRetry` **[Required]** `int32`	maximumRetry is number of retries the PV recycler will execute on failure to recycle PV.
`MinimumTimeoutNFS` **[Required]** `int32`	minimumTimeoutNFS is the minimum ActiveDeadlineSeconds to use for an NFS Recycler pod.
`PodTemplateFilePathNFS` **[Required]** `string`	podTemplateFilePathNFS is the file path to a pod definition used as a template for NFS persistent volume recycling
`IncrementTimeoutNFS` **[Required]** `int32`	incrementTimeoutNFS is the increment of time added per Gi to ActiveDeadlineSeconds for an NFS scrubber pod.
`PodTemplateFilePathHostPath` **[Required]** `string`	podTemplateFilePathHostPath is the file path to a pod definition used as a template for HostPath persistent volume recycling. This is for development and testing only and will not work in a multi-node cluster.
`MinimumTimeoutHostPath` **[Required]** `int32`	minimumTimeoutHostPath is the minimum ActiveDeadlineSeconds to use for a HostPath Recycler pod. This is for development and testing only and will not work in a multi-node cluster.
`IncrementTimeoutHostPath` **[Required]** `int32`	incrementTimeoutHostPath is the increment of time added per Gi to ActiveDeadlineSeconds for a HostPath scrubber pod. This is for development and testing only and will not work in a multi-node cluster.

`PodGCControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

PodGCControllerConfiguration contains elements describing PodGCController.

Field	Description
`TerminatedPodGCThreshold` **[Required]** `int32`	terminatedPodGCThreshold is the number of terminated pods that can exist before the terminated pod garbage collector starts deleting terminated pods. If <= 0, the terminated pod garbage collector is disabled.

`ReplicaSetControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

ReplicaSetControllerConfiguration contains elements describing ReplicaSetController.

Field	Description
`ConcurrentRSSyncs` **[Required]** `int32`	concurrentRSSyncs is the number of replica sets that are allowed to sync concurrently. Larger number = more responsive replica management, but more CPU (and network) load.

`ReplicationControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

ReplicationControllerConfiguration contains elements describing ReplicationController.

Field	Description
`ConcurrentRCSyncs` **[Required]** `int32`	concurrentRCSyncs is the number of replication controllers that are allowed to sync concurrently. Larger number = more responsive replica management, but more CPU (and network) load.

`ResourceQuotaControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

ResourceQuotaControllerConfiguration contains elements describing ResourceQuotaController.

Field	Description
`ResourceQuotaSyncPeriod` **[Required]** `meta/v1.Duration`	resourceQuotaSyncPeriod is the period for syncing quota usage status in the system.
`ConcurrentResourceQuotaSyncs` **[Required]** `int32`	concurrentResourceQuotaSyncs is the number of resource quotas that are allowed to sync concurrently. Larger number = more responsive quota management, but more CPU (and network) load.

`SAControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

SAControllerConfiguration contains elements describing ServiceAccountController.

Field	Description
`ServiceAccountKeyFile` **[Required]** `string`	serviceAccountKeyFile is the filename containing a PEM-encoded private RSA key used to sign service account tokens.
`ConcurrentSATokenSyncs` **[Required]** `int32`	concurrentSATokenSyncs is the number of service account token syncing operations that will be done concurrently.
`RootCAFile` **[Required]** `string`	rootCAFile is the root certificate authority will be included in service account's token secret. This must be a valid PEM-encoded CA bundle.

`StatefulSetControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

StatefulSetControllerConfiguration contains elements describing StatefulSetController.

Field	Description
`ConcurrentStatefulSetSyncs` **[Required]** `int32`	concurrentStatefulSetSyncs is the number of statefulset objects that are allowed to sync concurrently. Larger number = more responsive statefulsets, but more CPU (and network) load.

`TTLAfterFinishedControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

TTLAfterFinishedControllerConfiguration contains elements describing TTLAfterFinishedController.

Field	Description
`ConcurrentTTLSyncs` **[Required]** `int32`	concurrentTTLSyncs is the number of TTL-after-finished collector workers that are allowed to sync concurrently.

`ValidatingAdmissionPolicyStatusControllerConfiguration`

Appears in:

- KubeControllerManagerConfiguration

ValidatingAdmissionPolicyStatusControllerConfiguration contains elements describing ValidatingAdmissionPolicyStatusController.

Field	Description
`ConcurrentPolicySyncs` **[Required]** `int32`	ConcurrentPolicySyncs is the number of policy objects that are allowed to sync concurrently. Larger number = quicker type checking, but more CPU (and network) load. The default value is 5.

`VolumeConfiguration`

Appears in:

- PersistentVolumeBinderControllerConfiguration

VolumeConfiguration contains *all* enumerated flags meant to configure all volume plugins. From this config, the controller-manager binary will create many instances of volume.VolumeConfig, each containing only the configuration needed for that plugin which are then passed to the appropriate plugin. The ControllerManager binary is the only part of the code which knows what plugins are supported and which flags correspond to each plugin.

Field	Description
`EnableHostPathProvisioning` **[Required]** `bool`	enableHostPathProvisioning enables HostPath PV provisioning when running without a cloud provider. This allows testing and development of provisioning features. HostPath provisioning is not supported in any way, won't work in a multi-node cluster, and should not be used for anything other than testing or development.
`EnableDynamicProvisioning` **[Required]** `bool`	enableDynamicProvisioning enables the provisioning of volumes when running within an environment that supports dynamic provisioning. Defaults to true.
`PersistentVolumeRecyclerConfiguration` **[Required]** `PersistentVolumeRecyclerConfiguration`	persistentVolumeRecyclerConfiguration holds configuration for persistent volume plugins.
`FlexVolumePluginDir` **[Required]** `string`	volumePluginDir is the full path of the directory in which the flex volume plugin should search for additional third party volume plugins

---

kube-scheduler Configuration (v1)

Resource Types

- DefaultPreemptionArgs
- DynamicResourcesArgs
- InterPodAffinityArgs
- KubeSchedulerConfiguration
- NodeAffinityArgs
- NodeResourcesBalancedAllocationArgs
- NodeResourcesFitArgs
- PodTopologySpreadArgs
- VolumeBindingArgs

`ClientConnectionConfiguration`

Appears in:

- KubeSchedulerConfiguration

ClientConnectionConfiguration contains details for constructing a client.

Field	Description
`kubeconfig` **[Required]** `string`	kubeconfig is the path to a KubeConfig file.
`acceptContentTypes` **[Required]** `string`	acceptContentTypes defines the Accept header sent by clients when connecting to a server, overriding the default value of 'application/json'. This field will control all connections to the server used by a particular client.
`contentType` **[Required]** `string`	contentType is the content type used when sending data to the server from this client.
`qps` **[Required]** `float32`	qps controls the number of queries per second allowed for this connection.
`burst` **[Required]** `int32`	burst allows extra queries to accumulate when a client is exceeding its rate.

`DebuggingConfiguration`

Appears in:

- KubeSchedulerConfiguration

DebuggingConfiguration holds configuration for Debugging related features.

Field	Description
`enableProfiling` **[Required]** `bool`	enableProfiling enables profiling via web interface host:port/debug/pprof/
`enableContentionProfiling` **[Required]** `bool`	enableContentionProfiling enables block profiling, if enableProfiling is true.

`LeaderElectionConfiguration`

Appears in:

- KubeSchedulerConfiguration

LeaderElectionConfiguration defines the configuration of leader election clients for components that can run with leader election enabled.

Field	Description
`leaderElect` **[Required]** `bool`	leaderElect enables a leader election client to gain leadership before executing the main loop. Enable this when running replicated components for high availability.
`leaseDuration` **[Required]** `meta/v1.Duration`	leaseDuration is the duration that non-leader candidates will wait after observing a leadership renewal until attempting to acquire leadership of a led but unrenewed leader slot. This is effectively the maximum duration that a leader can be stopped before it is replaced by another candidate. This is only applicable if leader election is enabled.
`renewDeadline` **[Required]** `meta/v1.Duration`	renewDeadline is the interval between attempts by the acting master to renew a leadership slot before it stops leading. This must be less than or equal to the lease duration. This is only applicable if leader election is enabled.
`retryPeriod` **[Required]** `meta/v1.Duration`	retryPeriod is the duration the clients should wait between attempting acquisition and renewal of a leadership. This is only applicable if leader election is enabled.
`resourceLock` **[Required]** `string`	resourceLock indicates the resource object type that will be used to lock during leader election cycles.
`resourceName` **[Required]** `string`	resourceName indicates the name of resource object that will be used to lock during leader election cycles.
`resourceNamespace` **[Required]** `string`	resourceName indicates the namespace of resource object that will be used to lock during leader election cycles.

`DefaultPreemptionArgs`

DefaultPreemptionArgs holds arguments used to configure the DefaultPreemption plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`DefaultPreemptionArgs`
`minCandidateNodesPercentage` **[Required]** `int32`	MinCandidateNodesPercentage is the minimum number of candidates to shortlist when dry running preemption as a percentage of number of nodes. Must be in the range [0, 100]. Defaults to 10% of the cluster size if unspecified.
`minCandidateNodesAbsolute` **[Required]** `int32`	MinCandidateNodesAbsolute is the absolute minimum number of candidates to shortlist. The likely number of candidates enumerated for dry running preemption is given by the formula: numCandidates = max(numNodes * minCandidateNodesPercentage, minCandidateNodesAbsolute) We say "likely" because there are other factors such as PDB violations that play a role in the number of candidates shortlisted. Must be at least 0 nodes. Defaults to 100 nodes if unspecified.

`DynamicResourcesArgs`

DynamicResourcesArgs holds arguments used to configure the DynamicResources plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`DynamicResourcesArgs`
`filterTimeout` **[Required]** `meta/v1.Duration`	FilterTimeout limits the amount of time that the filter operation may take per node to search for devices that can be allocated to scheduler a pod to that node. In typical scenarios, this operation should complete in 10 to 200 milliseconds, but could also be longer depending on the number of requests per ResourceClaim, number of ResourceClaims,

Field	Description
	number of published devices in ResourceSlices, and the complexity of the requests. Other checks besides CEL evaluation also take time (usage checks, match attributes, etc.). Therefore the scheduler plugin applies this timeout. If the timeout is reached, the Pod is considered unschedulable for the node. If filtering succeeds for some other node(s), those are picked instead. If filtering fails for all of them, the Pod is placed in the unschedulable queue. It will get checked again if changes in e.g. ResourceSlices or ResourceClaims indicate that another scheduling attempt might succeed. If this fails repeatedly, exponential backoff slows down future attempts. The default is 10 seconds. This is sufficient to prevent worst-case scenarios while not impacting normal usage of DRA. However, slow filtering can slow down Pod scheduling also for Pods not using DRA. Administators can reduce the timeout after checking the `scheduler_framework_extension_point_duration_seconds` metrics. Setting it to zero completely disables the timeout.

`InterPodAffinityArgs`

InterPodAffinityArgs holds arguments used to configure the InterPodAffinity plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`InterPodAffinityArgs`
`hardPodAffinityWeight` **[Required]** `int32`	HardPodAffinityWeight is the scoring weight for existing pods with a matching hard affinity to the incoming pod.
`ignorePreferredTermsOfExistingPods` **[Required]** `bool`	IgnorePreferredTermsOfExistingPods configures the scheduler to ignore existing pods' preferred affinity rules when scoring candidate nodes, unless the incoming pod has inter-pod affinities.

`KubeSchedulerConfiguration`

KubeSchedulerConfiguration configures a scheduler

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`KubeSchedulerConfiguration`
`parallelism` **[Required]** `int32`	Parallelism defines the amount of parallelism in algorithms for scheduling a Pods. Must be greater than 0. Defaults to 16
`leaderElection` **[Required]** LeaderElectionConfiguration	LeaderElection defines the configuration of leader election client.
`clientConnection` **[Required]** ClientConnectionConfiguration	ClientConnection specifies the kubeconfig file and client connection settings for the proxy server to use when communicating with the apiserver.
`DebuggingConfiguration` **[Required]** DebuggingConfiguration	(Members of `DebuggingConfiguration` are embedded into this type.) DebuggingConfiguration holds configuration for Debugging related features TODO: We might wanna make this a substruct like Debugging componentbaseconfigv1alpha1.DebuggingConfiguration
`percentageOfNodesToScore` **[Required]** `int32`	PercentageOfNodesToScore is the percentage of all nodes that once found feasible for running a pod, the scheduler stops its search for more feasible nodes in the cluster. This helps improve scheduler's performance. Scheduler always tries to find at least "minFeasibleNodesToFind" feasible nodes no matter what the value of this flag is. Example: if the cluster size is 500 nodes and the value of this flag is 30, then scheduler stops finding further feasible nodes once it finds 150 feasible ones. When the value is 0, default percentage (5%--50% based on the size of the cluster) of the nodes will be scored. It is overridden by profile level PercentageOfNodesToScore.
`podInitialBackoffSeconds` **[Required]** `int64`	PodInitialBackoffSeconds is the initial backoff for unschedulable pods. If specified, it must be greater than 0. If this value is null, the default value (1s) will be used.
`podMaxBackoffSeconds` **[Required]** `int64`	PodMaxBackoffSeconds is the max backoff for unschedulable pods. If specified, it must be greater than podInitialBackoffSeconds. If this value is null, the default value (10s) will be used.
`profiles` **[Required]** []KubeSchedulerProfile	Profiles are scheduling profiles that kube-scheduler supports. Pods can choose to be scheduled under a particular profile by setting its associated scheduler name. Pods that don't specify any scheduler name are scheduled with the "default-scheduler" profile, if present here.
`extenders` **[Required]** []Extender	Extenders are the list of scheduler extenders, each holding the values of how to communicate with the extender. These extenders are shared by all scheduler profiles.

Field	Description
`delayCacheUntilActive` **[Required]** `bool`	DelayCacheUntilActive specifies when to start caching. If this is true and leader election is enabled, the scheduler will wait to fill informer caches until it is the leader. Doing so will have slower failover with the benefit of lower memory overhead while waiting to become leader. Defaults to false.

NodeAffinityArgs

NodeAffinityArgs holds arguments to configure the NodeAffinity plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`NodeAffinityArgs`
`addedAffinity` core/v1.NodeAffinity	AddedAffinity is applied to all Pods additionally to the NodeAffinity specified in the PodSpec. That is, Nodes need to satisfy AddedAffinity AND .spec.NodeAffinity. AddedAffinity is empty by default (all Nodes match). When AddedAffinity is used, some Pods with affinity requirements that match a specific Node (such as Daemonset Pods) might remain unschedulable.

NodeResourcesBalancedAllocationArgs

NodeResourcesBalancedAllocationArgs holds arguments used to configure NodeResourcesBalancedAllocation plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`NodeResourcesBalancedAllocationArgs`
`resources` **[Required]** []ResourceSpec	Resources to be managed, the default is "cpu" and "memory" if not specified.

NodeResourcesFitArgs

NodeResourcesFitArgs holds arguments used to configure the NodeResourcesFit plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`NodeResourcesFitArgs`
`ignoredResources` **[Required]** `[]string`	IgnoredResources is the list of resources that NodeResources fit filter should ignore. This doesn't apply to scoring.
`ignoredResourceGroups` **[Required]** `[]string`	IgnoredResourceGroups defines the list of resource groups that NodeResources fit filter should ignore. e.g. if group is ["example.com"], it will ignore all resource names that begin with "example.com", such as "example.com/aaa" and "example.com/bbb". A resource group name can't contain '/'. This doesn't apply to scoring.
`scoringStrategy` **[Required]** ScoringStrategy	ScoringStrategy selects the node resource scoring strategy. The default strategy is LeastAllocated with an equal "cpu" and "memory" weight.

PodTopologySpreadArgs

PodTopologySpreadArgs holds arguments used to configure the PodTopologySpread plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`PodTopologySpreadArgs`
`defaultConstraints` []core/v1.TopologySpreadConstraint	DefaultConstraints defines topology spread constraints to be applied to Pods that don't define any in `pod.spec.topologySpreadConstraints`. `.defaultConstraints[*].labelSelectors` must be empty, as they are deduced from the Pod's membership to Services, ReplicationControllers, ReplicaSets or StatefulSets. When not empty, .defaultingType must be "List".
`defaultingType` PodTopologySpreadConstraintsDefaulting	DefaultingType determines how .defaultConstraints are deduced. Can be one of "System" or "List".

Field	Description
	• "System": Use kubernetes defined constraints that spread Pods among Nodes and Zones. • "List": Use constraints defined in .defaultConstraints. Defaults to "System".

`VolumeBindingArgs`

VolumeBindingArgs holds arguments used to configure the VolumeBinding plugin.

Field	Description
`apiVersion` string	`kubescheduler.config.k8s.io/v1`
`kind` string	`VolumeBindingArgs`
`bindTimeoutSeconds` **[Required]** `int64`	BindTimeoutSeconds is the timeout in seconds in volume binding operation. Value must be non-negative integer. The value zero indicates no waiting. If this value is nil, the default value (600) will be used.
`shape` `[]UtilizationShapePoint`	Shape specifies the points defining the score function shape, which is used to score nodes based on the utilization of provisioned PVs. The utilization is calculated by dividing the total requested storage of the pod by the total capacity of feasible PVs on each node. Each point contains utilization (ranges from 0 to 100) and its associated score (ranges from 0 to 10). You can turn the priority by specifying different scores for different utilization numbers. The default shape points are: 1. 10 for 0 utilization 2. 0 for 100 utilization All points must be sorted in increasing order by utilization.

`Extender`

Appears in:

- KubeSchedulerConfiguration

Extender holds the parameters used to communicate with the extender. If a verb is unspecified/empty, it is assumed that the extender chose not to provide that extension.

Field	Description
`urlPrefix` **[Required]** `string`	URLPrefix at which the extender is available
`filterVerb` **[Required]** `string`	Verb for the filter call, empty if not supported. This verb is appended to the URLPrefix when issuing the filter call to extender.
`preemptVerb` **[Required]** `string`	Verb for the preempt call, empty if not supported. This verb is appended to the URLPrefix when issuing the preempt call to extender.
`prioritizeVerb` **[Required]** `string`	Verb for the prioritize call, empty if not supported. This verb is appended to the URLPrefix when issuing the prioritize call to extender.
`weight` **[Required]** `int64`	The numeric multiplier for the node scores that the prioritize call generates. The weight should be a positive integer
`bindVerb` **[Required]** `string`	Verb for the bind call, empty if not supported. This verb is appended to the URLPrefix when issuing the bind call to extender. If this method is implemented by the extender, it is the extender's responsibility to bind the pod to apiserver. Only one extender can implement this function.
`enableHTTPS` **[Required]** `bool`	EnableHTTPS specifies whether https should be used to communicate with the extender
`tlsConfig` **[Required]** `ExtenderTLSConfig`	TLSConfig specifies the transport layer security config
`httpTimeout` **[Required]** `meta/v1.Duration`	HTTPTimeout specifies the timeout duration for a call to the extender. Filter timeout fails the scheduling of the pod. Prioritize timeout is ignored, k8s/other extenders priorities are used to select the node.
`nodeCacheCapable` **[Required]** `bool`	NodeCacheCapable specifies that the extender is capable of caching node information, so the scheduler should only send minimal information about the eligible nodes assuming that the extender already cached full details of all nodes in the cluster
`managedResources` `[]ExtenderManagedResource`	ManagedResources is a list of extended resources that are managed by this extender. • A pod will be sent to the extender on the Filter, Prioritize and Bind (if the extender is the binder) phases iff the pod requests at least one of the extended resources in this list. If empty or unspecified, all pods will be sent to this extender.

Field	Description
	• If IgnoredByScheduler is set to true for a resource, kube-scheduler will skip checking the resource in predicates.
`ignorable` **[Required]** `bool`	Ignorable specifies if the extender is ignorable, i.e. scheduling should not fail when the extender returns an error or is not reachable.

`ExtenderManagedResource`

Appears in:

- Extender

ExtenderManagedResource describes the arguments of extended resources managed by an extender.

Field	Description
`name` **[Required]** `string`	Name is the extended resource name.
`ignoredByScheduler` **[Required]** `bool`	IgnoredByScheduler indicates whether kube-scheduler should ignore this resource when applying predicates.

`ExtenderTLSConfig`

Appears in:

- Extender

ExtenderTLSConfig contains settings to enable TLS with extender

Field	Description
`insecure` **[Required]** `bool`	Server should be accessed without verifying the TLS certificate. For testing only.
`serverName` **[Required]** `string`	ServerName is passed to the server for SNI and is used in the client to check server certificates against. If ServerName is empty, the hostname used to contact the server is used.
`certFile` **[Required]** `string`	Server requires TLS client certificate authentication
`keyFile` **[Required]** `string`	Server requires TLS client certificate authentication
`caFile` **[Required]** `string`	Trusted root certificates for server
`certData` **[Required]** `[]byte`	CertData holds PEM-encoded bytes (typically read from a client certificate file). CertData takes precedence over CertFile
`keyData` **[Required]** `[]byte`	KeyData holds PEM-encoded bytes (typically read from a client certificate key file). KeyData takes precedence over KeyFile
`caData` **[Required]** `[]byte`	CAData holds PEM-encoded bytes (typically read from a root certificates bundle). CAData takes precedence over CAFile

`KubeSchedulerProfile`

Appears in:

- KubeSchedulerConfiguration

KubeSchedulerProfile is a scheduling profile.

Field	Description
`schedulerName` **[Required]** `string`	SchedulerName is the name of the scheduler associated to this profile. If SchedulerName matches with the pod's "spec.schedulerName", then the pod is scheduled with this profile.
`percentageOfNodesToScore` **[Required]** `int32`	PercentageOfNodesToScore is the percentage of all nodes that once found feasible for running a pod, the scheduler stops its search for more feasible nodes in the cluster. This helps improve scheduler's performance. Scheduler always tries to find at least "minFeasibleNodesToFind" feasible nodes no matter what the value of this flag is. Example: if the cluster size is 500 nodes and the value of this flag is 30, then scheduler stops finding further feasible nodes once it finds 150 feasible ones. When the value is 0, default percentage (5%--50% based on the size of the cluster) of the nodes will be scored. It will override global PercentageOfNodesToScore. If it is empty, global PercentageOfNodesToScore will be used.
`plugins` **[Required]** `Plugins`	Plugins specify the set of plugins that should be enabled or disabled. Enabled plugins are the ones that should be enabled in addition to the default plugins. Disabled plugins are any of the

Field	Description
	default plugins that should be disabled. When no enabled or disabled plugin is specified for an extension point, default plugins for that extension point will be used if there is any. If a QueueSort plugin is specified, the same QueueSort Plugin and PluginConfig must be specified for all profiles.
`pluginConfig` **[Required]** `[]PluginConfig`	PluginConfig is an optional set of custom plugin arguments for each plugin. Omitting config args for a plugin is equivalent to using the default config for that plugin.

`Plugin`

Appears in:

- PluginSet

Plugin specifies a plugin name and its weight when applicable. Weight is used only for Score plugins.

Field	Description
`name` **[Required]** `string`	Name defines the name of plugin
`weight` **[Required]** `int32`	Weight defines the weight of plugin, only used for Score plugins.

`PluginConfig`

Appears in:

- KubeSchedulerProfile

PluginConfig specifies arguments that should be passed to a plugin at the time of initialization. A plugin that is invoked at multiple extension points is initialized once. Args can have arbitrary structure. It is up to the plugin to process these Args.

Field	Description
`name` **[Required]** `string`	Name defines the name of plugin being configured
`args` **[Required]** `k8s.io/apimachinery/pkg/runtime.RawExtension`	Args defines the arguments passed to the plugins at the time of initialization. Args can have arbitrary structure.

`PluginSet`

Appears in:

- Plugins

PluginSet specifies enabled and disabled plugins for an extension point. If an array is empty, missing, or nil, default plugins at that extension point will be used.

Field	Description
`enabled` **[Required]** `[]Plugin`	Enabled specifies plugins that should be enabled in addition to default plugins. If the default plugin is also configured in the scheduler config file, the weight of plugin will be overridden accordingly. These are called after default plugins and in the same order specified here.
`disabled` **[Required]** `[]Plugin`	Disabled specifies default plugins that should be disabled. When all default plugins need to be disabled, an array containing only one "*" should be provided.

`Plugins`

Appears in:

- KubeSchedulerProfile

Plugins include multiple extension points. When specified, the list of plugins for a particular extension point are the only ones enabled. If an extension point is omitted from the config, then the default set of plugins is used for that extension point. Enabled plugins are called in the order specified here, after default plugins. If they need to be invoked before default plugins, default plugins must be disabled and re-enabled here in desired order.

Field	Description
`preEnqueue` **[Required]** `PluginSet`	PreEnqueue is a list of plugins that should be invoked before adding pods to the scheduling queue.

Field	Description
`queueSort` **[Required]** `PluginSet`	QueueSort is a list of plugins that should be invoked when sorting pods in the scheduling queue.
`preFilter` **[Required]** `PluginSet`	PreFilter is a list of plugins that should be invoked at "PreFilter" extension point of the scheduling framework.
`filter` **[Required]** `PluginSet`	Filter is a list of plugins that should be invoked when filtering out nodes that cannot run the Pod.
`postFilter` **[Required]** `PluginSet`	PostFilter is a list of plugins that are invoked after filtering phase, but only when no feasible nodes were found for the pod.
`preScore` **[Required]** `PluginSet`	PreScore is a list of plugins that are invoked before scoring.
`score` **[Required]** `PluginSet`	Score is a list of plugins that should be invoked when ranking nodes that have passed the filtering phase.
`reserve` **[Required]** `PluginSet`	Reserve is a list of plugins invoked when reserving/unreserving resources after a node is assigned to run the pod.
`permit` **[Required]** `PluginSet`	Permit is a list of plugins that control binding of a Pod. These plugins can prevent or delay binding of a Pod.
`preBind` **[Required]** `PluginSet`	PreBind is a list of plugins that should be invoked before a pod is bound.
`bind` **[Required]** `PluginSet`	Bind is a list of plugins that should be invoked at "Bind" extension point of the scheduling framework. The scheduler call these plugins in order. Scheduler skips the rest of these plugins as soon as one returns success.
`postBind` **[Required]** `PluginSet`	PostBind is a list of plugins that should be invoked after a pod is successfully bound.
`multiPoint` **[Required]** `PluginSet`	MultiPoint is a simplified config section to enable plugins for all valid extension points. Plugins enabled through MultiPoint will automatically register for every individual extension point the plugin has implemented. Disabling a plugin through MultiPoint disables that behavior. The same is true for disabling "*" through MultiPoint (no default plugins will be automatically registered). Plugins can still be disabled through their individual extension points. In terms of precedence, plugin config follows this basic hierarchy 1. Specific extension points 2. Explicitly configured MultiPoint plugins 3. The set of default plugins, as MultiPoint plugins This implies that a higher precedence plugin will run first and overwrite any settings within MultiPoint. Explicitly user-configured plugins also take a higher precedence over default plugins. Within this hierarchy, an Enabled setting takes precedence over Disabled. For example, if a plugin is set in both `multiPoint.Enabled` and `multiPoint.Disabled`, the plugin will be enabled. Similarly, including `multiPoint.Disabled = '*'` and `multiPoint.Enabled = pluginA` will still register that specific plugin through MultiPoint. This follows the same behavior as all other extension point configurations.

`PodTopologySpreadConstraintsDefaulting`

(Alias of `string`)

Appears in:

- PodTopologySpreadArgs

PodTopologySpreadConstraintsDefaulting defines how to set default constraints for the PodTopologySpread plugin.

`RequestedToCapacityRatioParam`

Appears in:

- ScoringStrategy

RequestedToCapacityRatioParam define RequestedToCapacityRatio parameters

Field	Description
`shape` **[Required]** `[]UtilizationShapePoint`	Shape is a list of points defining the scoring function shape.

`ResourceSpec`

Appears in:

- NodeResourcesBalancedAllocationArgs
- ScoringStrategy

ResourceSpec represents a single resource.

Field	Description
`name` **[Required]** `string`	Name of the resource.
`weight` **[Required]** `int64`	Weight of the resource.

`ScoringStrategy`

Appears in:

- NodeResourcesFitArgs

ScoringStrategy define ScoringStrategyType for node resource plugin

Field	Description
`type` **[Required]** `ScoringStrategyType`	Type selects which strategy to run.
`resources` **[Required]** `[]ResourceSpec`	Resources to consider when scoring. The default resource set includes "cpu" and "memory" with an equal weight. Allowed weights go from 1 to 100. Weight defaults to 1 if not specified or explicitly set to 0.
`requestedToCapacityRatio` **[Required]** `RequestedToCapacityRatioParam`	Arguments specific to RequestedToCapacityRatio strategy.

`ScoringStrategyType`

(Alias of `string`)

Appears in:

- ScoringStrategy

ScoringStrategyType the type of scoring strategy used in NodeResourcesFit plugin.

`UtilizationShapePoint`

Appears in:

- VolumeBindingArgs
- RequestedToCapacityRatioParam

UtilizationShapePoint represents single point of priority function shape.

Field	Description
`utilization` **[Required]** `int32`	Utilization (x axis). Valid values are 0 to 100. Fully utilized node maps to 100.
`score` **[Required]** `int32`	Score assigned to given utilization (y axis). Valid values are 0 to 10.

---

kube-proxy

Synopsis

The Kubernetes network proxy runs on each node. This reflects services as defined in the Kubernetes API on each node and can do simple TCP, UDP, and SCTP stream forwarding or round robin TCP, UDP, and SCTP forwarding across a set of backends. Service cluster IPs and ports are currently found through Docker-links-compatible environment variables specifying ports opened by the service proxy. There is an optional addon that provides cluster DNS for these cluster IPs. The user must create a service with the apiserver API to configure the proxy.

```
kube-proxy [flags]
```

Options

--add_dir_header

If true, adds the file directory to the header of the log messages

--alsologtostderr

log to standard error as well as files (no effect when -logtostderr=true)

--bind-address string Default: 0.0.0.0

Overrides kube-proxy's idea of what its node's primary IP is. Note that the name is a historical artifact, and kube-proxy does not actually bind any sockets to this IP. This parameter is ignored if a config file is specified by --config.

--bind-address-hard-fail

If true kube-proxy will treat failure to bind to a port as fatal and exit

--cleanup

If true cleanup iptables and ipvs rules and exit.

--cluster-cidr string

The CIDR range of the pods in the cluster. (For dual-stack clusters, this can be a comma-separated dual-stack pair of CIDR ranges.). When --detect-local-mode is set to ClusterCIDR, kube-proxy will consider traffic to be local if its source IP is in this range. (Otherwise it is not used.) This parameter is ignored if a config file is specified by --config.

--config string

The path to the configuration file.

--config-sync-period duration Default: 15m0s

How often configuration from the apiserver is refreshed. Must be greater than 0.

--conntrack-max-per-core int32 Default: 32768

Maximum number of NAT connections to track per CPU core (0 to leave the limit as-is and ignore conntrack-min).

--conntrack-min int32 Default: 131072

Minimum number of conntrack entries to allocate, regardless of conntrack-max-per-core (set conntrack-max-per-core=0 to leave the limit as-is).

--conntrack-tcp-be-liberal

Enable liberal mode for tracking TCP packets by setting nf_conntrack_tcp_be_liberal to 1

--conntrack-tcp-timeout-close-wait duration Default: 1h0m0s

NAT timeout for TCP connections in the CLOSE_WAIT state

--conntrack-tcp-timeout-established duration Default: 24h0m0s

Idle timeout for established TCP connections (0 to leave as-is)

--conntrack-udp-timeout duration

Idle timeout for UNREPLIED UDP connections (0 to leave as-is)

--conntrack-udp-timeout-stream duration

Idle timeout for ASSURED UDP connections (0 to leave as-is)

--detect-local-mode LocalMode

Mode to use to detect local traffic. This parameter is ignored if a config file is specified by --config.

--feature-gates <comma-separated 'key=True|False' pairs>

A set of key=value pairs that describe feature gates for alpha/experimental features. Options are:
APIResponseCompression=true|false (BETA - default=true)
APIServerIdentity=true|false (BETA - default=true)
APIServingWithRoutine=true|false (ALPHA - default=false)
AllAlpha=true|false (ALPHA - default=false)
AllBeta=true|false (BETA - default=false)
AllowParsingUserUIDFromCertAuth=true|false (BETA - default=true)
AllowUnsafeMalformedObjectDeletion=true|false (ALPHA - default=false)
CBORServingAndStorage=true|false (ALPHA - default=false)
CPUManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
CPUManagerPolicyBetaOptions=true|false (BETA - default=true)
CSIVolumeHealth=true|false (ALPHA - default=false)
ClearingNominatedNodeNameAfterBinding=true|false (ALPHA - default=false)
ClientsAllowCBOR=true|false (ALPHA - default=false)
ClientsPreferCBOR=true|false (ALPHA - default=false)
CloudControllerManagerWebhook=true|false (ALPHA - default=false)
ClusterTrustBundle=true|false (BETA - default=false)
ClusterTrustBundleProjection=true|false (BETA - default=false)
ComponentFlagz=true|false (ALPHA - default=false)
ComponentStatusz=true|false (ALPHA - default=false)
ConcurrentWatchObjectDecode=true|false (BETA - default=false)

ContainerCheckpoint=true|false (BETA - default=true)
ContainerRestartRules=true|false (ALPHA - default=false)
ContainerStopSignals=true|false (ALPHA - default=false)
ContextualLogging=true|false (BETA - default=true)
CoordinatedLeaderElection=true|false (BETA - default=false)
CrossNamespaceVolumeDataSource=true|false (ALPHA - default=false)
CustomCPUCFSQuotaPeriod=true|false (ALPHA - default=false)
DRAAdminAccess=true|false (BETA - default=true)
DRAConsumableCapacity=true|false (ALPHA - default=false)
DRADeviceBindingConditions=true|false (ALPHA - default=false)
DRADeviceTaints=true|false (ALPHA - default=false)
DRAExtendedResource=true|false (ALPHA - default=false)
DRAPartitionableDevices=true|false (ALPHA - default=false)
DRAPrioritizedList=true|false (BETA - default=true)
DRAResourceClaimDeviceStatus=true|false (BETA - default=true)
DRASchedulerFilterTimeout=true|false (BETA - default=true)
DeclarativeValidation=true|false (BETA - default=true)
DeclarativeValidationTakeover=true|false (BETA - default=false)
DeploymentReplicaSetTerminatingReplicas=true|false (ALPHA - default=false)
DetectCacheInconsistency=true|false (BETA - default=true)
DisableCPUQuotaWithExclusiveCPUs=true|false (BETA - default=true)
EnvFiles=true|false (ALPHA - default=false)
EventedPLEG=true|false (ALPHA - default=false)
ExternalServiceAccountTokenSigner=true|false (BETA - default=true)
GracefulNodeShutdown=true|false (BETA - default=true)
GracefulNodeShutdownBasedOnPodPriority=true|false (BETA - default=true)
HPAConfigurableTolerance=true|false (ALPHA - default=false)
HPAScaleToZero=true|false (ALPHA - default=false)
HostnameOverride=true|false (ALPHA - default=false)
ImageMaximumGCAge=true|false (BETA - default=true)
ImageVolume=true|false (BETA - default=false)
InOrderInformers=true|false (BETA - default=true)
InPlacePodVerticalScaling=true|false (BETA - default=true)
InPlacePodVerticalScalingExclusiveCPUs=true|false (ALPHA - default=false)
InPlacePodVerticalScalingExclusiveMemory=true|false (ALPHA - default=false)
InTreePluginPortworxUnregister=true|false (ALPHA - default=false)
InformerResourceVersion=true|false (ALPHA - default=false)
JobManagedBy=true|false (BETA - default=true)
KubeletCrashLoopBackOffMax=true|false (ALPHA - default=false)
KubeletEnsureSecretPulledImages=true|false (ALPHA - default=false)
KubeletFineGrainedAuthz=true|false (BETA - default=true)
KubeletInUserNamespace=true|false (ALPHA - default=false)
KubeletPSI=true|false (BETA - default=true)
KubeletPodResourcesDynamicResources=true|false (BETA - default=true)
KubeletPodResourcesGet=true|false (BETA - default=true)
KubeletSeparateDiskGC=true|false (BETA - default=true)
KubeletServiceAccountTokenForCredentialProviders=true|false (BETA - default=true)
ListFromCacheSnapshot=true|false (BETA - default=true)
LocalStorageCapacityIsolationFSQuotaMonitoring=true|false (BETA - default=false)
LoggingAlphaOptions=true|false (ALPHA - default=false)
LoggingBetaOptions=true|false (BETA - default=true)
MatchLabelKeysInPodTopologySpread=true|false (BETA - default=true)
MatchLabelKeysInPodTopologySpreadSelectorMerge=true|false (BETA - default=true)
MaxUnavailableStatefulSet=true|false (ALPHA - default=false)
MemoryQoS=true|false (ALPHA - default=false)
MutableCSINodeAllocatableCount=true|false (BETA - default=false)
MutatingAdmissionPolicy=true|false (BETA - default=false)
NodeLogQuery=true|false (BETA - default=false)
NominatedNodeNameForExpectation=true|false (ALPHA - default=false)
OpenAPIEnums=true|false (BETA - default=true)
PodAndContainerStatsFromCRI=true|false (ALPHA - default=false)
PodCertificateRequest=true|false (ALPHA - default=false)

PodDeletionCost=true|false (BETA - default=true)
PodLevelResources=true|false (BETA - default=true)
PodLogsQuerySplitStreams=true|false (ALPHA - default=false)
PodObservedGenerationTracking=true|false (BETA - default=true)
PodReadyToStartContainersCondition=true|false (BETA - default=true)
PodTopologyLabelsAdmission=true|false (ALPHA - default=false)
PortForwardWebsockets=true|false (BETA - default=true)
PreferSameTrafficDistribution=true|false (BETA - default=true)
PreventStaticPodAPIReferences=true|false (BETA - default=true)
ProcMountType=true|false (BETA - default=true)
QOSReserved=true|false (ALPHA - default=false)
ReduceDefaultCrashLoopBackOffDecay=true|false (ALPHA - default=false)
RelaxedServiceNameValidation=true|false (ALPHA - default=false)
ReloadKubeletServerCertificateFile=true|false (BETA - default=true)
RemoteRequestHeaderUID=true|false (BETA - default=true)
ResourceHealthStatus=true|false (ALPHA - default=false)
RotateKubeletServerCertificate=true|false (BETA - default=true)
RuntimeClassInImageCriApi=true|false (ALPHA - default=false)
SELinuxChangePolicy=true|false (BETA - default=true)
SELinuxMount=true|false (BETA - default=false)
SELinuxMountReadWriteOncePod=true|false (BETA - default=true)
SchedulerAsyncAPICalls=true|false (BETA - default=true)
SchedulerAsyncPreemption=true|false (BETA - default=true)
SchedulerPopFromBackoffQ=true|false (BETA - default=true)
ServiceAccountNodeAudienceRestriction=true|false (BETA - default=true)
SizeBasedListCostEstimate=true|false (BETA - default=true)
StorageCapacityScoring=true|false (ALPHA - default=false)
StorageVersionAPI=true|false (ALPHA - default=false)
StorageVersionHash=true|false (BETA - default=true)
StorageVersionMigrator=true|false (ALPHA - default=false)
StrictIPCIDRValidation=true|false (ALPHA - default=false)
StructuredAuthenticationConfigurationEgressSelector=true|false (BETA - default=true)
SupplementalGroupsPolicy=true|false (BETA - default=true)
SystemdWatchdog=true|false (BETA - default=true)
TokenRequestServiceAccountUIDValidation=true|false (BETA - default=true)
TopologyManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
TopologyManagerPolicyBetaOptions=true|false (BETA - default=true)
TranslateStreamCloseWebsocketRequests=true|false (BETA - default=true)
UnauthenticatedHTTP2DOSMitigation=true|false (BETA - default=true)
UnknownVersionInteroperabilityProxy=true|false (ALPHA - default=false)
UserNamespacesPodSecurityStandards=true|false (ALPHA - default=false)
UserNamespacesSupport=true|false (BETA - default=true)
WatchCacheInitializationPostStartHook=true|false (BETA - default=false)
WatchList=true|false (BETA - default=true)
WatchListClient=true|false (BETA - default=false)
WindowsCPUAndMemoryAffinity=true|false (ALPHA - default=false)
WindowsGracefulNodeShutdown=true|false (BETA - default=true)
This parameter is ignored if a config file is specified by --config.

--healthz-bind-address ipport Default: 0.0.0.0:10256

The IP address and port for the health check server to serve on, defaulting to "0.0.0.0:10256". This parameter is ignored if a config file is specified by --config.

-h, --help

help for kube-proxy

--hostname-override string

If non-empty, will be used as the name of the Node that kube-proxy is running on. If unset, the node name is assumed to be the same as the node's hostname.

--init-only

If true, perform any initialization steps that must be done with full root privileges, and then exit. After doing this, you can run kube-proxy again with only the CAP_NET_ADMIN capability.

--iptables-localhost-nodeports Default: true

If false, kube-proxy will disable the legacy behavior of allowing NodePort services to be accessed via localhost. (Applies only to iptables mode and IPv4; localhost NodePorts are never allowed with other proxy modes or with IPv6.)

--iptables-masquerade-bit int32 Default: 14

If using the iptables or ipvs proxy mode, the bit of the fwmark space to mark packets requiring SNAT with. Must be within the range [0, 31].

--iptables-min-sync-period duration Default: 1s

The minimum period between iptables rule resyncs (e.g. '5s', '1m', '2h22m'). A value of 0 means every Service or EndpointSlice change will result in an immediate iptables resync.

--iptables-sync-period duration Default: 30s

An interval (e.g. '5s', '1m', '2h22m') indicating how frequently various re-synchronizing and cleanup operations are performed. Must be greater than 0.

--ipvs-exclude-cidrs strings

A comma-separated list of CIDRs which the ipvs proxier should not touch when cleaning up IPVS rules.

--ipvs-min-sync-period duration Default: 1s

The minimum period between IPVS rule resyncs (e.g. '5s', '1m', '2h22m'). A value of 0 means every Service or EndpointSlice change will result in an immediate IPVS resync.

--ipvs-scheduler string

The ipvs scheduler type when proxy mode is ipvs

--ipvs-strict-arp

Enable strict ARP by setting arp_ignore to 1 and arp_announce to 2

--ipvs-sync-period duration Default: 30s

An interval (e.g. '5s', '1m', '2h22m') indicating how frequently various re-synchronizing and cleanup operations are performed. Must be greater than 0.

--ipvs-tcp-timeout duration

The timeout for idle IPVS TCP connections, 0 to leave as-is. (e.g. '5s', '1m', '2h22m').

--ipvs-tcpfin-timeout duration

The timeout for IPVS TCP connections after receiving a FIN packet, 0 to leave as-is. (e.g. '5s', '1m', '2h22m').

--ipvs-udp-timeout duration

The timeout for IPVS UDP packets, 0 to leave as-is. (e.g. '5s', '1m', '2h22m').

--kube-api-burst int32 Default: 10

Burst to use while talking with kubernetes apiserver

--kube-api-content-type string Default: "application/vnd.kubernetes.protobuf"

Content type of requests sent to apiserver.

--kube-api-qps float Default: 5

QPS to use while talking with kubernetes apiserver

--kubeconfig string

Path to kubeconfig file with authorization information (the master location can be overridden by the master flag).

--log-flush-frequency duration Default: 5s

Maximum number of seconds between log flushes

--log-text-info-buffer-size quantity

[Alpha] In text format with split output streams, the info messages can be buffered for a while to increase performance. The default value of zero bytes disables buffering. The size can be specified as number of bytes (512), multiples of 1000 (1K), multiples of 1024 (2Ki), or powers of those (3M, 4G, 5Mi, 6Gi). Enable the LoggingAlphaOptions feature gate to use this.

--log-text-split-stream

[Alpha] In text format, write error messages to stderr and info messages to stdout. The default is to write a single stream to stdout. Enable the LoggingAlphaOptions feature gate to use this.

--log_backtrace_at <a string in the form 'file:N'> Default: :0

when logging hits line file:N, emit a stack trace

--log_dir string

If non-empty, write log files in this directory (no effect when -logtostderr=true)

--log_file string

If non-empty, use this log file (no effect when -logtostderr=true)

--log_file_max_size uint Default: 1800

Defines the maximum size a log file can grow to (no effect when -logtostderr=true). Unit is megabytes. If the value is 0, the maximum file size is unlimited.

--logging-format string Default: "text"

Sets the log format. Permitted formats: "text".

--logtostderr Default: true

log to standard error instead of files

--masquerade-all

SNAT all traffic sent via Service cluster IPs. This may be required with some CNI plugins. Only supported on Linux.

--master string

The address of the Kubernetes API server (overrides any value in kubeconfig)

--metrics-bind-address ipport Default: 127.0.0.1:10249

The IP address and port for the metrics server to serve on, defaulting to "127.0.0.1:10249". (Set to "0.0.0.0:10249" / "[::]:10249" to bind on all interfaces.) Set empty to disable. This parameter is ignored if a config file is specified by --config.

--nodeport-addresses strings

A list of CIDR ranges that contain valid node IPs, or alternatively, the single string 'primary'. If set to a list of CIDRs, connections to NodePort services will only be accepted on node IPs in one of the indicated ranges. If set to 'primary', NodePort services will only be accepted on the node's primary IP(s) according to the Node object. If unset, NodePort connections will be accepted on all local IPs. This parameter is ignored if a config file is specified by --config.

--one_output

If true, only write logs to their native severity level (vs also writing to each lower severity level; no effect when -logtostderr=true)

--oom-score-adj int32 Default: -999

The oom-score-adj value for kube-proxy process. Values must be within the range [-1000, 1000]. This parameter is ignored if a config file is specified by --config.

--pod-bridge-interface string

A bridge interface name. When --detect-local-mode is set to BridgeInterface, kube-proxy will consider traffic to be local if it originates from this bridge.

--pod-interface-name-prefix string

An interface name prefix. When --detect-local-mode is set to InterfaceNamePrefix, kube-proxy will consider traffic to be local if it originates from any interface whose name begins with this prefix.

--profiling

If true enables profiling via web interface on /debug/pprof handler. This parameter is ignored if a config file is specified by --config.

--proxy-mode ProxyMode

Which proxy mode to use: on Linux this can be 'iptables' (default), 'ipvs', or 'nftables'. On Windows the only supported value is 'kernelspace'. This parameter is ignored if a config file is specified by --config.

--show-hidden-metrics-for-version string

The previous version for which you want to show hidden metrics. Only the previous minor version is meaningful, other values will not be allowed. The format is <major>.<minor>, e.g.: '1.16'. The purpose of this format is make sure you have the opportunity to notice if the next release hides additional metrics, rather than being surprised when they are permanently removed in the release after that. This parameter is ignored if a config file is specified by --config.

--skip_headers

If true, avoid header prefixes in the log messages

--skip_log_headers

If true, avoid headers when opening log files (no effect when -logtostderr=true)

--stderrthreshold int Default: 2

logs at or above this threshold go to stderr when writing to files and stderr (no effect when -logtostderr=true or -alsologtostderr=true)

-v, --v int

number for the log level verbosity

--version version[=true]

--version, --version=raw prints version information and quits; --version=vX.Y.Z... sets the reported version

--vmodule pattern=N,...

comma-separated list of pattern=N settings for file-filtered logging (only works for text log format)

--write-config-to string

If set, write the default configuration values to this file and exit.

Image Policy API (v1alpha1)

Resource Types

- ImageReview

`ImageReview`

ImageReview checks if the set of images in a pod are allowed.

Field	Description
`apiVersion` string	`imagepolicy.k8s.io/v1alpha1`
`kind` string	`ImageReview`
`metadata` meta/v1.ObjectMeta	Standard object's metadata. More info: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata Refer to the Kubernetes API documentation for the fields of the `metadata` field.
`spec` **[Required]** ImageReviewSpec	Spec holds information about the pod being evaluated
`status` ImageReviewStatus	Status is filled in by the backend and indicates whether the pod should be allowed.

`ImageReviewContainerSpec`

Appears in:

- ImageReviewSpec

ImageReviewContainerSpec is a description of a container within the pod creation request.

Field	Description
`image` string	This can be in the form image:tag or image@SHA:012345679abcdef.

`ImageReviewSpec`

Appears in:

- ImageReview

ImageReviewSpec is a description of the pod creation request.

Field	Description
`containers` []ImageReviewContainerSpec	Containers is a list of a subset of the information in each container of the Pod being created.
`annotations` map[string]string	Annotations is a list of key-value pairs extracted from the Pod's annotations. It only includes keys which match the pattern `*.image-policy.k8s.io/*`. It is up to each webhook backend to determine how to interpret these annotations, if at all.
`namespace` string	Namespace is the namespace the pod is being created in.

`ImageReviewStatus`

Appears in:

- ImageReview

ImageReviewStatus is the result of the review for the pod creation request.

Field	Description
`allowed` **[Required]** bool	Allowed indicates that all images were allowed to be run.
`reason` string	Reason should be empty unless Allowed is false in which case it may contain a short description of what is wrong. Kubernetes may truncate excessively long errors when displaying to the user.
`auditAnnotations` map[string]string	AuditAnnotations will be added to the attributes object of the admission controller request using 'AddAnnotation'. The keys should be prefix-less (i.e., the admission controller will add an appropriate prefix).

---

kube-apiserver Configuration (v1alpha1)

Package v1alpha1 is the v1alpha1 version of the API.

Resource Types

`TracingConfiguration`

Appears in:

- KubeletConfiguration
- TracingConfiguration

TracingConfiguration provides versioned configuration for OpenTelemetry tracing clients.

Field	Description
`endpoint` `string`	Endpoint of the collector this component will report traces to. The connection is insecure, and does not currently support TLS. Recommended is unset, and endpoint is the otlp grpc default, localhost:4317.
`samplingRatePerMillion` `int32`	SamplingRatePerMillion is the number of samples to collect per million spans. Recommended is unset. If unset, sampler respects its parent span's sampling rate, but otherwise never samples.

`AdmissionConfiguration`

AdmissionConfiguration provides versioned configuration for admission controllers.

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1alpha1`
`kind` string	`AdmissionConfiguration`
`plugins` `[]AdmissionPluginConfiguration`	Plugins allows specifying a configuration per admission control plugin.

`AuthenticationConfiguration`

AuthenticationConfiguration provides versioned configuration for authentication.

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1alpha1`
`kind` string	`AuthenticationConfiguration`
`jwt` **[Required]** `[]JWTAuthenticator`	jwt is a list of authenticator to authenticate Kubernetes users using JWT compliant tokens. The authenticator will attempt to parse a raw ID token, verify it's been signed by the configured issuer. The public key to verify the signature is discovered from the issuer's public endpoint using OIDC discovery. For an incoming token, each JWT authenticator will be attempted in the order in which it is specified in this list. Note however that other authenticators may run before or after the JWT authenticators. The specific position of JWT authenticators in relation to other authenticators is neither defined nor stable across releases. Since each JWT authenticator must have a unique issuer URL, at most one JWT authenticator will attempt to cryptographically validate the token. The minimum valid JWT payload must contain the following claims: { "iss": "https://issuer.example.com", "aud": ["audience"], "exp": 1234567890, "": "username" }
`anonymous` **[Required]** `AnonymousAuthConfig`	If present --anonymous-auth must not be set

`AuthorizationConfiguration`

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1alpha1`
`kind` string	`AuthorizationConfiguration`

Field	Description
`authorizers` **[Required]** []AuthorizerConfiguration	Authorizers is an ordered list of authorizers to authorize requests against. This is similar to the --authorization-modes kube-apiserver flag Must be at least one.

EgressSelectorConfiguration

EgressSelectorConfiguration provides versioned configuration for egress selector clients.

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1alpha1`
`kind` string	`EgressSelectorConfiguration`
`egressSelections` **[Required]** []EgressSelection	connectionServices contains a list of egress selection client configurations

TracingConfiguration

TracingConfiguration provides versioned configuration for tracing clients.

Field	Description
`apiVersion` string	`apiserver.k8s.io/v1alpha1`
`kind` string	`TracingConfiguration`
`TracingConfiguration` **[Required]** TracingConfiguration	(Members of `TracingConfiguration` are embedded into this type.) Embed the component config tracing configuration struct

AdmissionPluginConfiguration

Appears in:

- AdmissionConfiguration

AdmissionPluginConfiguration provides the configuration for a single plug-in.

Field	Description
`name` **[Required]** `string`	Name is the name of the admission controller. It must match the registered admission plugin name.
`path` `string`	Path is the path to a configuration file that contains the plugin's configuration
`configuration` k8s.io/apimachinery/pkg/runtime.Unknown	Configuration is an embedded configuration object to be used as the plugin's configuration. If present, it will be used instead of the path to the configuration file.

AnonymousAuthCondition

Appears in:

- AnonymousAuthConfig

AnonymousAuthCondition describes the condition under which anonymous auth should be enabled.

Field	Description
`path` **[Required]** `string`	Path for which anonymous auth is enabled.

AnonymousAuthConfig

Appears in:

- AuthenticationConfiguration

AnonymousAuthConfig provides the configuration for the anonymous authenticator.

Field	Description
`enabled` **[Required]** `bool`	No description provided.

Field	Description
`conditions` **[Required]** []AnonymousAuthCondition	If set, anonymous auth is only allowed if the request meets one of the conditions.

AudienceMatchPolicyType

(Alias of `string`)

Appears in:

- Issuer

AudienceMatchPolicyType is a set of valid values for issuer.audienceMatchPolicy

AuthorizerConfiguration

Appears in:

- AuthorizationConfiguration

Field	Description
`type` **[Required]** `string`	Type refers to the type of the authorizer "Webhook" is supported in the generic API server Other API servers may support additional authorizer types like Node, RBAC, ABAC, etc.
`name` **[Required]** `string`	Name used to describe the webhook This is explicitly used in monitoring machinery for metrics Note: Names must be DNS1123 labels like `myauthorizername` or subdomains like `myauthorizer.example.domain` Required, with no default
`webhook` **[Required]** WebhookConfiguration	Webhook defines the configuration for a Webhook authorizer Must be defined when Type=Webhook Must not be defined when Type!=Webhook

ClaimMappings

Appears in:

- JWTAuthenticator

ClaimMappings provides the configuration for claim mapping

Field	Description
`username` **[Required]** PrefixedClaimOrExpression	username represents an option for the username attribute. The claim's value must be a singular string. Same as the --oidc-username-claim and --oidc-username-prefix flags. If username.expression is set, the expression must produce a string value. If username.expression uses 'claims.email', then 'claims.email_verified' must be used in username.expression or extra[*].valueExpression or claimValidationRules[*].expression. An example claim validation rule expression that matches the validation automatically applied when username.claim is set to 'email' is 'claims.?email_verified.orValue(true) == true'. By explicitly comparing the value to true, we let type-checking see the result will be a boolean, and to make sure a non-boolean email_verified claim will be caught at runtime. In the flag based approach, the --oidc-username-claim and --oidc-username-prefix are optional. If --oidc-username-claim is not set, the default value is "sub". For the authentication config, there is no defaulting for claim or prefix. The claim and prefix must be set explicitly. For claim, if --oidc-username-claim was not set with legacy flag approach, configure username.claim="sub" in the authentication config. For prefix: (1) --oidc-username-prefix="-", no prefix was added to the username. For the same behavior using authentication config, set username.prefix="" (2) --oidc-username-prefix="" and --oidc-username-claim != "email", prefix was "<value of --oidc-issuer-url>#". For the same behavior using authentication config, set username.prefix="#" (3) --oidc-username-prefix="". For the same behavior using authentication config, set username.prefix=""
`groups` PrefixedClaimOrExpression	groups represents an option for the groups attribute. The claim's value must be a string or string array claim. If groups.claim is set, the prefix must be specified (and can be the empty string). If groups.expression is set, the expression must produce a string or string array value. "", [], and null values are treated as the group mapping not being present.
`uid` ClaimOrExpression	uid represents an option for the uid attribute. Claim must be a singular string claim. If uid.expression is set, the expression must produce a string value.
`extra` []ExtraMapping	extra represents an option for the extra attribute. expression must produce a string or string array value. If the value is empty, the extra mapping will not be present. hard-coded extra key/value - key: "foo" valueExpression: "'bar'" This will result in an extra attribute - foo: ["bar"]

Field	Description
	hard-coded key, value copying claim value • key: "foo" valueExpression: "claims.some_claim" This will result in an extra attribute - foo: [value of some_claim] hard-coded key, value derived from claim value • key: "admin" valueExpression: '(has(claims.is_admin) && claims.is_admin) ? "true":""' This will result in: • if is_admin claim is present and true, extra attribute - admin: ["true"] • if is_admin claim is present and false or is_admin claim is not present, no extra attribute will be added

`ClaimOrExpression`

Appears in:

- ClaimMappings

ClaimOrExpression provides the configuration for a single claim or expression.

Field	Description
`claim` `string`	claim is the JWT claim to use. Either claim or expression must be set. Mutually exclusive with expression.
`expression` `string`	expression represents the expression which will be evaluated by CEL. CEL expressions have access to the contents of the token claims, organized into CEL variable: • 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim.

`ClaimValidationRule`

Appears in:

- JWTAuthenticator

ClaimValidationRule provides the configuration for a single claim validation rule.

Field	Description
`claim` `string`	claim is the name of a required claim. Same as --oidc-required-claim flag. Only string claim keys are supported. Mutually exclusive with expression and message.
`requiredValue` `string`	requiredValue is the value of a required claim. Same as --oidc-required-claim flag. Only string claim values are supported. If claim is set and requiredValue is not set, the claim must be present with a value set to the empty string. Mutually exclusive with expression and message.
`expression` `string`	expression represents the expression which will be evaluated by CEL. Must produce a boolean. CEL expressions have access to the contents of the token claims, organized into CEL variable: • 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Must return true for the validation to pass. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim and requiredValue.
`message` `string`	message customizes the returned error message when expression returns false. message is a literal string. Mutually exclusive with claim and requiredValue.

`Connection`

Appears in:

- EgressSelection

Connection provides the configuration for a single egress selection client.

Field	Description
`proxyProtocol` **[Required]** ProtocolType	Protocol is the protocol used to connect from client to the konnectivity server.
`transport` Transport	Transport defines the transport configurations we use to dial to the konnectivity server. This is required if ProxyProtocol is HTTPConnect or GRPC.

EgressSelection

Appears in:

- EgressSelectorConfiguration

EgressSelection provides the configuration for a single egress selection client.

Field	Description
`name` **[Required]** `string`	name is the name of the egress selection. Currently supported values are "controlplane", "master", "etcd" and "cluster" The "master" egress selector is deprecated in favor of "controlplane"
`connection` **[Required]** Connection	connection is the exact information used to configure the egress selection

EgressSelectorType

(Alias of `string`)

Appears in:

- Issuer

EgressSelectorType is an indicator of which egress selection should be used for sending traffic.

ExtraMapping

Appears in:

- ClaimMappings

ExtraMapping provides the configuration for a single extra mapping.

Field	Description
`key` **[Required]** `string`	key is a string to use as the extra attribute key. key must be a domain-prefix path (e.g. example.org/foo). All characters before the first "/" must be a valid subdomain as defined by RFC 1123. All characters trailing the first "/" must be valid HTTP Path characters as defined by RFC 3986. key must be lowercase. Required to be unique.
`valueExpression` **[Required]** `string`	valueExpression is a CEL expression to extract extra attribute value. valueExpression must produce a string or string array value. "", [], and null values are treated as the extra mapping not being present. Empty string values contained within a string array are filtered out. CEL expressions have access to the contents of the token claims, organized into CEL variable: - 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/

Issuer

Appears in:

- JWTAuthenticator

Issuer provides the configuration for an external provider's specific settings.

Field	Description
`url` **[Required]** `string`	url points to the issuer URL in a format https://url or https://url/path. This must match the "iss" claim in the presented JWT, and the issuer returned from discovery. Same value as the --oidc-issuer-url flag. Discovery information is fetched from "{url}/.well-known/openid-configuration" unless overridden by discoveryURL. Required to be unique across all JWT authenticators. Note that egress selection configuration is not used for this network connection. discoveryURL, if specified, overrides the URL used to fetch discovery information instead of using "{url}/.well-known/openid-configuration". The exact value specified is used, so "/.well-known/openid-configuration" must be included in discoveryURL if needed. The "issuer" field in the fetched discovery information must match the "issuer.url" field in the AuthenticationConfiguration and will be used to validate the "iss" claim in the presented JWT. This is for scenarios where the well-known and jwks endpoints are hosted at a different location than the issuer (such as locally in the cluster). Example: A discovery url that is exposed using kubernetes service 'oidc' in namespace 'oidc-namespace' and discovery information is available at '/.well-known/openid-configuration'. discoveryURL: "https://oidc.oidc-namespace/.well-known/openid-configuration" certificateAuthority is used to verify the TLS connection and the hostname on the leaf certificate must be set to 'oidc.oidc-namespace'. curl https://oidc.oidc-namespace/.well-known/openid-configuration (.discoveryURL field) { issuer: "https://oidc.example.com" (.url field) } discoveryURL must be different from url. Required to be unique across all JWT authenticators. Note that egress selection configuration is not used for this network connection.
`discoveryURL` `string`	
`certificateAuthority` `string`	certificateAuthority contains PEM-encoded certificate authority certificates used to validate the connection when fetching discovery information. If unset, the system verifier is used. Same value as the content of the file referenced by the --oidc-ca-file flag.
`audiences` **[Required]** `[]string`	audiences is the set of acceptable audiences the JWT must be issued to. At least one of the entries must match the "aud" claim in presented JWTs. Same value as the --oidc-client-id flag (though this field supports an array). Required to be non-empty.
`audienceMatchPolicy` `AudienceMatchPolicyType`	audienceMatchPolicy defines how the "audiences" field is used to match the "aud" claim in the presented JWT. Allowed values are: 1. "MatchAny" when multiple audiences are specified and 2. empty (or unset) or "MatchAny" when a single audience is specified. - MatchAny: the "aud" claim in the presented JWT must match at least one of the entries in the "audiences" field. For example, if "audiences" is ["foo", "bar"], the "aud" claim in the presented JWT must contain either "foo" or "bar" (and may contain both). - "": The match policy can be empty (or unset) when a single audience is specified in the "audiences" field. The "aud" claim in the presented JWT must contain the single audience (and may contain others). For more nuanced audience validation, use claimValidationRules. example: claimValidationRule[].expression: 'sets.equivalent(claims.aud, ["bar", "foo", "baz"])' to require an exact match.
`egressSelectorType` `EgressSelectorType`	egressSelectorType is an indicator of which egress selection should be used for sending all traffic related to this issuer (discovery, JWKS, distributed claims, etc). If unspecified, no custom dialer is used. When specified, the valid choices are "controlplane" and "cluster". These correspond to the associated values in the --egress-selector-config-file. - controlplane: for traffic intended to go to the control plane. - cluster: for traffic intended to go to the system being managed by Kubernetes.

`JWTAuthenticator`

Appears in:

- AuthenticationConfiguration

JWTAuthenticator provides the configuration for a single JWT authenticator.

Field	Description
`issuer` **[Required]** `Issuer`	issuer contains the basic OIDC provider connection options.

Field	Description
`claimValidationRules` [][]ClaimValidationRule	claimValidationRules are rules that are applied to validate token claims to authenticate users.
`claimMappings` **[Required]** ClaimMappings	claimMappings points claims of a token to be treated as user attributes.
`userValidationRules` [][]UserValidationRule	userValidationRules are rules that are applied to final user before completing authentication. These allow invariants to be applied to incoming identities such as preventing the use of the system: prefix that is commonly used by Kubernetes components. The validation rules are logically ANDed together and must all return true for the validation to pass.

PrefixedClaimOrExpression

Appears in:

- ClaimMappings

PrefixedClaimOrExpression provides the configuration for a single prefixed claim or expression.

Field	Description
`claim` string	claim is the JWT claim to use. Mutually exclusive with expression.
`prefix` string	prefix is prepended to claim's value to prevent clashes with existing names. prefix needs to be set if claim is set and can be the empty string. Mutually exclusive with expression.
`expression` string	expression represents the expression which will be evaluated by CEL. CEL expressions have access to the contents of the token claims, organized into CEL variable: - 'claims' is a map of claim names to claim values. For example, a variable named 'sub' can be accessed as 'claims.sub'. Nested claims can be accessed using dot notation, e.g. 'claims.foo.bar'. Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/ Mutually exclusive with claim and prefix.

ProtocolType

(Alias of `string`)

Appears in:

- Connection

ProtocolType is a set of valid values for Connection.ProtocolType

TCPTransport

Appears in:

- Transport

TCPTransport provides the information to connect to konnectivity server via TCP

Field	Description
`url` **[Required]** string	URL is the location of the konnectivity server to connect to. As an example it might be "https://127.0.0.1:8131"
`tlsConfig` TLSConfig	TLSConfig is the config needed to use TLS when connecting to konnectivity server

TLSConfig

Appears in:

- TCPTransport

TLSConfig provides the authentication information to connect to konnectivity server Only used with TCPTransport

Field	Description
`caBundle` string	caBundle is the file location of the CA to be used to determine trust with the konnectivity server. Must be absent/empty if TCPTransport.URL is prefixed with http:// If absent while

Field	Description
	TCPTransport.URL is prefixed with https://, default to system trust roots.
`clientKey` `string`	clientKey is the file location of the client key to be used in mtls handshakes with the konnectivity server. Must be absent/empty if TCPTransport.URL is prefixed with http:// Must be configured if TCPTransport.URL is prefixed with https://
`clientCert` `string`	clientCert is the file location of the client certificate to be used in mtls handshakes with the konnectivity server. Must be absent/empty if TCPTransport.URL is prefixed with http:// Must be configured if TCPTransport.URL is prefixed with https://

`Transport`

Appears in:

- Connection

Transport defines the transport configurations we use to dial to the konnectivity server

Field	Description
`tcp` `TCPTransport`	TCP is the TCP configuration for communicating with the konnectivity server via TCP ProxyProtocol of GRPC is not supported with TCP transport at the moment Requires at least one of TCP or UDS to be set
`uds` `UDSTransport`	UDS is the UDS configuration for communicating with the konnectivity server via UDS Requires at least one of TCP or UDS to be set

`UDSTransport`

Appears in:

- Transport

UDSTransport provides the information to connect to konnectivity server via UDS

Field	Description
`udsName` **[Required]** `string`	UDSName is the name of the unix domain socket to connect to konnectivity server This does not use a unix:// prefix. (Eg: /etc/srv/kubernetes/konnectivity-server/konnectivity-server.socket)

`UserValidationRule`

Appears in:

- JWTAuthenticator

UserValidationRule provides the configuration for a single user info validation rule.

Field	Description
`expression` **[Required]** `string`	expression represents the expression which will be evaluated by CEL. Must return true for the validation to pass. CEL expressions have access to the contents of UserInfo, organized into CEL variable: - 'user' - authentication.k8s.io/v1, Kind=UserInfo object Refer to https://github.com/kubernetes/api/blob/release-1.28/authentication/v1/types.go#L105-L122 for the definition. API documentation: https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.28/#userinfo-v1-authentication-k8s-io Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/
`message` `string`	message customizes the returned error message when rule returns false. message is a literal string.

`WebhookConfiguration`

Appears in:

- AuthorizerConfiguration

Field	Description
`authorizedTTL` **[Required]** meta/v1.Duration	The duration to cache 'authorized' responses from the webhook authorizer. Same as setting `--authorization-webhook-cache-authorized-ttl` flag Default: 5m0s
`cacheAuthorizedRequests` `bool`	CacheAuthorizedRequests specifies whether authorized requests should be cached. If set to true, the TTL for cached decisions can be configured via the AuthorizedTTL field. Default: true
`unauthorizedTTL` **[Required]** meta/v1.Duration	The duration to cache 'unauthorized' responses from the webhook authorizer. Same as setting `--authorization-webhook-cache-unauthorized-ttl` flag Default: 30s
`cacheUnauthorizedRequests` `bool`	CacheUnauthorizedRequests specifies whether unauthorized requests should be cached. If set to true, the TTL for cached decisions can be configured via the UnauthorizedTTL field. Default: true
`timeout` **[Required]** meta/v1.Duration	Timeout for the webhook request Maximum allowed value is 30s. Required, no default value.
`subjectAccessReviewVersion` **[Required]** `string`	The API version of the authorization.k8s.io SubjectAccessReview to send to and expect from the webhook. Same as setting `--authorization-webhook-version` flag Valid values: v1beta1, v1 Required, no default value
`matchConditionSubjectAccessReviewVersion` **[Required]** `string`	MatchConditionSubjectAccessReviewVersion specifies the SubjectAccessReview version the CEL expressions are evaluated against Valid values: v1 Required, no default value
`failurePolicy` **[Required]** `string`	Controls the authorization decision when a webhook request fails to complete or returns a malformed response or errors evaluating matchConditions. Valid values: • NoOpinion: continue to subsequent authorizers to see if one of them allows the request • Deny: reject the request without consulting subsequent authorizers Required, with no default.
`connectionInfo` **[Required]** WebhookConnectionInfo	ConnectionInfo defines how we talk to the webhook

`matchConditions` **[Required]**
`[]`WebhookMatchCondition
	matchConditions is a list of conditions that must be met for a request to be sent to this webhook. An empty list of matchConditions matches all requests. There are a maximum of 64 match conditions allowed.
	The exact matching logic is (in order):
	1. If at least one matchCondition evaluates to FALSE, then the webhook is skipped.
	2. If ALL matchConditions evaluate to TRUE, then the webhook is called.
	3. If at least one matchCondition evaluates to an error (but none are FALSE):
	  ◦ If failurePolicy=Deny, then the webhook rejects the request
	  ◦ If failurePolicy=NoOpinion, then the error is ignored and the webhook is skipped

`WebhookConnectionInfo`

Appears in:

- WebhookConfiguration

Field	Description
`type` **[Required]** `string`	Controls how the webhook should communicate with the server. Valid values: • KubeConfigFile: use the file specified in kubeConfigFile to locate the server. • InClusterConfig: use the in-cluster configuration to call the SubjectAccessReview API hosted by kube-apiserver. This mode is not allowed for kube-apiserver.
`kubeConfigFile` **[Required]** `string`	Path to KubeConfigFile for connection info Required, if connectionInfo.Type is KubeConfig

`WebhookMatchCondition`

Appears in:

- WebhookConfiguration

Field	Description
`expression` **[Required]** `string`	expression represents the expression which will be evaluated by CEL. Must evaluate to bool. CEL expressions have access to the contents of the SubjectAccessReview in v1 version. If

Field	Description
	version specified by subjectAccessReviewVersion in the request variable is v1beta1, the contents would be converted to the v1 version before evaluating the CEL expression. • 'resourceAttributes' describes information for a resource access request and is unset for non-resource requests. e.g. has(request.resourceAttributes) && request.resourceAttributes.namespace == 'default' • 'nonResourceAttributes' describes information for a non-resource access request and is unset for resource requests. e.g. has(request.nonResourceAttributes) && request.nonResourceAttributes.path == '/healthz'. • 'user' is the user to test for. e.g. request.user == 'alice' • 'groups' is the groups to test for. e.g. ('group1' in request.groups) • 'extra' corresponds to the user.Info.GetExtra() method from the authenticator. • 'uid' is the information about the requesting user. e.g. request.uid == '1' Documentation on CEL: https://kubernetes.io/docs/reference/using-api/cel/

---

kube-controller-manager

Synopsis

The Kubernetes controller manager is a daemon that embeds the core control loops shipped with Kubernetes. In applications of robotics and automation, a control loop is a non-terminating loop that regulates the state of the system. In Kubernetes, a controller is a control loop that watches the shared state of the cluster through the apiserver and makes changes attempting to move the current state towards the desired state. Examples of controllers that ship with Kubernetes today are the replication controller, endpoints controller, namespace controller, and serviceaccounts controller.

```
kube-controller-manager [flags]
```

Options

--allocate-node-cidrs

Should CIDRs for Pods be allocated and set on the cloud provider. Requires --cluster-cidr.

--allow-metric-labels stringToString Default: []

The map from metric-label to value allow-list of this label. The key's format is <MetricName>,<LabelName>. The value's format is <allowed_value>,<allowed_value>...e.g. metric1,label1='v1,v2,v3', metric1,label2='v1,v2,v3' metric2,label1='v1,v2,v3'.

--allow-metric-labels-manifest string

The path to the manifest file that contains the allow-list mapping. The format of the file is the same as the flag --allow-metric-labels. Note that the flag --allow-metric-labels will override the manifest file.

--attach-detach-reconcile-sync-period duration Default: 1m0s

The reconciler sync wait time between volume attach detach. This duration must be larger than one second, and increasing this value from the default may allow for volumes to be mismatched with pods.

--authentication-kubeconfig string

kubeconfig file pointing at the 'core' kubernetes server with enough rights to create tokenreviews.authentication.k8s.io. This is optional. If empty, all token requests are considered to be anonymous and no client CA is looked up in the cluster.

--authentication-skip-lookup

If false, the authentication-kubeconfig will be used to lookup missing authentication configuration from the cluster.

--authentication-token-webhook-cache-ttl duration Default: 10s

The duration to cache responses from the webhook token authenticator.

--authentication-tolerate-lookup-failure

If true, failures to look up missing authentication configuration from the cluster are not considered fatal. Note that this can result in authentication that treats all requests as anonymous.

--authorization-always-allow-paths strings Default: "/healthz,/readyz,/livez"

A list of HTTP paths to skip during authorization, i.e. these are authorized without contacting the 'core' kubernetes server.

--authorization-kubeconfig string

kubeconfig file pointing at the 'core' kubernetes server with enough rights to create subjectaccessreviews.authorization.k8s.io. This is optional. If empty, all requests not skipped by authorization are forbidden.

--authorization-webhook-cache-authorized-ttl duration Default: 10s

The duration to cache 'authorized' responses from the webhook authorizer.

--authorization-webhook-cache-unauthorized-ttl duration Default: 10s

The duration to cache 'unauthorized' responses from the webhook authorizer.

--bind-address string Default: 0.0.0.0

The IP address on which to listen for the --secure-port port. The associated interface(s) must be reachable by the rest of the cluster, and by CLI/web clients. If blank or an unspecified address (0.0.0.0 or ::), all interfaces and IP address families will be used.

--cert-dir string

The directory where the TLS certs are located. If --tls-cert-file and --tls-private-key-file are provided, this flag will be ignored.

--cidr-allocator-type string Default: "RangeAllocator"

Type of CIDR allocator to use

--client-ca-file string

If set, any request presenting a client certificate signed by one of the authorities in the client-ca-file is authenticated with an identity corresponding to the CommonName of the client certificate.

--cloud-config string

The path to the cloud provider configuration file. Empty string for no configuration file.

--cloud-provider string

The provider for cloud services. Empty string for no provider.

--cluster-cidr string

CIDR Range for Pods in cluster. Only used when --allocate-node-cidrs=true; if false, this option will be ignored.

--cluster-name string Default: "kubernetes"

The instance prefix for the cluster.

--cluster-signing-cert-file string

Filename containing a PEM-encoded X509 CA certificate used to issue cluster-scoped certificates. If specified, no more specific --cluster-signing-* flag may be specified.

--cluster-signing-duration duration Default: 8760h0m0s

The max length of duration signed certificates will be given. Individual CSRs may request shorter certs by setting spec.expirationSeconds.

--cluster-signing-key-file string

Filename containing a PEM-encoded RSA or ECDSA private key used to sign cluster-scoped certificates. If specified, no more specific --cluster-signing-* flag may be specified.

--cluster-signing-kube-apiserver-client-cert-file string

Filename containing a PEM-encoded X509 CA certificate used to issue certificates for the kubernetes.io/kube-apiserver-client signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--cluster-signing-kube-apiserver-client-key-file string

Filename containing a PEM-encoded RSA or ECDSA private key used to sign certificates for the kubernetes.io/kube-apiserver-client signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--cluster-signing-kubelet-client-cert-file string

Filename containing a PEM-encoded X509 CA certificate used to issue certificates for the kubernetes.io/kube-apiserver-client-kubelet signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--cluster-signing-kubelet-client-key-file string

Filename containing a PEM-encoded RSA or ECDSA private key used to sign certificates for the kubernetes.io/kube-apiserver-client-kubelet signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--cluster-signing-kubelet-serving-cert-file string

Filename containing a PEM-encoded X509 CA certificate used to issue certificates for the kubernetes.io/kubelet-serving signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--cluster-signing-kubelet-serving-key-file string

Filename containing a PEM-encoded RSA or ECDSA private key used to sign certificates for the kubernetes.io/kubelet-serving signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--cluster-signing-legacy-unknown-cert-file string

Filename containing a PEM-encoded X509 CA certificate used to issue certificates for the kubernetes.io/legacy-unknown signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--cluster-signing-legacy-unknown-key-file string

Filename containing a PEM-encoded RSA or ECDSA private key used to sign certificates for the kubernetes.io/legacy-unknown signer. If specified, --cluster-signing-{cert,key}-file must not be set.

--concurrent-cron-job-syncs int32 Default: 5

The number of cron job objects that are allowed to sync concurrently. Larger number = more responsive jobs, but more CPU (and network) load

--concurrent-daemonset-syncs int32 Default: 2

The number of daemonset objects that are allowed to sync concurrently. Larger number = more responsive daemonsets, but more CPU (and network) load

--concurrent-deployment-syncs int32 Default: 5

The number of deployment objects that are allowed to sync concurrently. Larger number = more responsive deployments, but more CPU (and network) load

--concurrent-endpoint-syncs int32 Default: 5

The number of endpoint syncing operations that will be done concurrently. Larger number = faster endpoint updating, but more CPU (and network) load

--concurrent-ephemeralvolume-syncs int32 Default: 5

The number of ephemeral volume syncing operations that will be done concurrently. Larger number = faster ephemeral volume updating, but more CPU (and network) load

--concurrent-gc-syncs int32 Default: 20

The number of garbage collector workers that are allowed to sync concurrently.

--concurrent-horizontal-pod-autoscaler-syncs int32 Default: 5

The number of horizontal pod autoscaler objects that are allowed to sync concurrently. Larger number = more responsive horizontal pod autoscaler objects processing, but more CPU (and network) load.

--concurrent-job-syncs int32 Default: 5

The number of job objects that are allowed to sync concurrently. Larger number = more responsive jobs, but more CPU (and network) load

--concurrent-namespace-syncs int32 Default: 10

The number of namespace objects that are allowed to sync concurrently. Larger number = more responsive namespace termination, but more CPU (and network) load

--concurrent-rc-syncs int32 Default: 5

The number of replication controllers that are allowed to sync concurrently. Larger number = more responsive replica management, but more CPU (and network) load

--concurrent-replicaset-syncs int32 Default: 5

The number of replica sets that are allowed to sync concurrently. Larger number = more responsive replica management, but more CPU (and network) load

--concurrent-resource-quota-syncs int32 Default: 5

The number of resource quotas that are allowed to sync concurrently. Larger number = more responsive quota management, but more CPU (and network) load

--concurrent-service-endpoint-syncs int32 Default: 5

The number of service endpoint syncing operations that will be done concurrently. Larger number = faster endpoint slice updating, but more CPU (and network) load. Defaults to 5.

--concurrent-service-syncs int32 Default: 1

The number of services that are allowed to sync concurrently. Larger number = more responsive service management, but more CPU (and network) load

--concurrent-serviceaccount-token-syncs int32 Default: 5

The number of service account token objects that are allowed to sync concurrently. Larger number = more responsive token generation, but more CPU (and network) load

--concurrent-statefulset-syncs int32 Default: 5

The number of statefulset objects that are allowed to sync concurrently. Larger number = more responsive statefulsets, but more CPU (and network) load

--concurrent-ttl-after-finished-syncs int32 Default: 5

The number of ttl-after-finished-controller workers that are allowed to sync concurrently.

--concurrent-validating-admission-policy-status-syncs int32 Default: 5

The number of ValidatingAdmissionPolicyStatusController workers that are allowed to sync concurrently.

--configure-cloud-routes Default: true

Should CIDRs allocated by allocate-node-cidrs be configured on the cloud provider.

--contention-profiling

Enable block profiling, if profiling is enabled

--controller-start-interval duration

Interval between starting controller managers.

--controllers strings Default: "*"

A list of controllers to enable. '*' enables all on-by-default controllers, 'foo' enables the controller named 'foo', '-foo' disables the controller named 'foo'.

All controllers: bootstrap-signer-controller, certificatesigningrequest-approving-controller, certificatesigningrequest-cleaner-controller, certificatesigningrequest-signing-controller, cloud-node-lifecycle-controller, clusterrole-aggregation-controller, cronjob-controller, daemonset-controller, deployment-controller, device-taint-eviction-controller, disruption-controller, endpoints-controller, endpointslice-

controller, endpointslice-mirroring-controller, ephemeral-volume-controller, garbage-collector-controller, horizontal-pod-autoscaler-controller, job-controller, kube-apiserver-serving-clustertrustbundle-publisher-controller, legacy-serviceaccount-token-cleaner-controller, namespace-controller, node-ipam-controller, node-lifecycle-controller, node-route-controller, persistentvolume-attach-detach-controller, persistentvolume-binder-controller, persistentvolume-expander-controller, persistentvolume-protection-controller, persistentvolumeclaim-protection-controller, pod-garbage-collector-controller, podcertificaterequest-cleaner-controller, replicaset-controller, replicationcontroller-controller, resourceclaim-controller, resourcequota-controller, root-ca-certificate-publisher-controller, selinux-warning-controller, service-cidr-controller, service-lb-controller, serviceaccount-controller, serviceaccount-token-controller, statefulset-controller, storage-version-migrator-controller, storageversion-garbage-collector-controller, taint-eviction-controller, token-cleaner-controller, ttl-after-finished-controller, ttl-controller, validatingadmissionpolicy-status-controller, volumeattributesclass-protection-controller
Disabled-by-default controllers: bootstrap-signer-controller, selinux-warning-controller, token-cleaner-controller

--disable-attach-detach-reconcile-sync

Disable volume attach detach reconciler sync. Disabling this may cause volumes to be mismatched with pods. Use wisely.

--disable-force-detach-on-timeout

Prevent force detaching volumes based on maximum unmount time and node status. If this flag is set to true, the non-graceful node shutdown feature must be used to recover from node failure. See https://k8s.io/docs/storage-disable-force-detach-on-timeout/.

--disable-http2-serving

If true, HTTP2 serving will be disabled [default=false]

--disabled-metrics strings

This flag provides an escape hatch for misbehaving metrics. You must provide the fully qualified metric name in order to disable it. Disclaimer: disabling metrics is higher in precedence than showing hidden metrics.

--emulated-version strings

The versions different components emulate their capabilities (APIs, features, ...) of.
If set, the component will emulate the behavior of this version instead of the underlying binary version.
Version format could only be major.minor, for example: '--emulated-version=wardle=1.2,kube=1.31'.
Options are: kube=1.31..1.34(default:1.34)
If the component is not specified, defaults to "kube"

--enable-dynamic-provisioning Default: true

Enable dynamic provisioning for environments that support it.

--enable-garbage-collector Default: true

Enables the generic garbage collector. MUST be synced with the corresponding flag of the kube-apiserver.

--enable-hostpath-provisioner

Enable HostPath PV provisioning when running without a cloud provider. This allows testing and development of provisioning features. HostPath provisioning is not supported in any way, won't work in a multi-node cluster, and should not be used for anything other than testing or development.

--enable-leader-migration

Whether to enable controller leader migration.

--endpoint-updates-batch-period duration

The length of endpoint updates batching period. Processing of pod changes will be delayed by this duration to join them with potential upcoming updates and reduce the overall number of endpoints updates. Larger number = higher endpoint programming latency, but lower number of endpoints revision generated

--endpointslice-updates-batch-period duration

The length of endpoint slice updates batching period. Processing of pod changes will be delayed by this duration to join them with potential upcoming updates and reduce the overall number of endpoints updates. Larger number = higher endpoint programming latency, but lower number of endpoints revision generated

--external-cloud-volume-plugin string

The plugin to use when cloud provider is set to external. Can be empty, should only be set when cloud-provider is external. Currently used to allow node-ipam-controller, persistentvolume-binder-controller, persistentvolume-expander-controller and attach-detach-controller to work for in tree cloud providers.

--feature-gates colonSeparatedMultimapStringString

Comma-separated list of component:key=value pairs that describe feature gates for alpha/experimental features of different components.
If the component is not specified, defaults to "kube". This flag can be repeatedly invoked. For example: --feature-gates 'wardle:featureA=true,wardle:featureB=false' --feature-gates 'kube:featureC=true'Options are:
kube:APIResponseCompression=true|false (BETA - default=true)
kube:APIServerIdentity=true|false (BETA - default=true)
kube:APIServingWithRoutine=true|false (ALPHA - default=false)
kube:AllAlpha=true|false (ALPHA - default=false)
kube:AllBeta=true|false (BETA - default=false)
kube:AllowParsingUserUIDFromCertAuth=true|false (BETA - default=true)

kube:AllowUnsafeMalformedObjectDeletion=true|false (ALPHA - default=false)
kube:CBORServingAndStorage=true|false (ALPHA - default=false)
kube:CPUManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
kube:CPUManagerPolicyBetaOptions=true|false (BETA - default=true)
kube:CSIVolumeHealth=true|false (ALPHA - default=false)
kube:ClearingNominatedNodeNameAfterBinding=true|false (ALPHA - default=false)
kube:ClientsAllowCBOR=true|false (ALPHA - default=false)
kube:ClientsPreferCBOR=true|false (ALPHA - default=false)
kube:CloudControllerManagerWebhook=true|false (ALPHA - default=false)
kube:ClusterTrustBundle=true|false (BETA - default=false)
kube:ClusterTrustBundleProjection=true|false (BETA - default=false)
kube:ComponentFlagz=true|false (ALPHA - default=false)
kube:ComponentStatusz=true|false (ALPHA - default=false)
kube:ConcurrentWatchObjectDecode=true|false (BETA - default=false)
kube:ContainerCheckpoint=true|false (BETA - default=true)
kube:ContainerRestartRules=true|false (ALPHA - default=false)
kube:ContainerStopSignals=true|false (ALPHA - default=false)
kube:ContextualLogging=true|false (BETA - default=true)
kube:CoordinatedLeaderElection=true|false (BETA - default=false)
kube:CrossNamespaceVolumeDataSource=true|false (ALPHA - default=false)
kube:CustomCPUCFSQuotaPeriod=true|false (ALPHA - default=false)
kube:DRAAdminAccess=true|false (BETA - default=true)
kube:DRAConsumableCapacity=true|false (ALPHA - default=false)
kube:DRADeviceBindingConditions=true|false (ALPHA - default=false)
kube:DRADeviceTaints=true|false (ALPHA - default=false)
kube:DRAExtendedResource=true|false (ALPHA - default=false)
kube:DRAPartitionableDevices=true|false (ALPHA - default=false)
kube:DRAPrioritizedList=true|false (BETA - default=true)
kube:DRAResourceClaimDeviceStatus=true|false (BETA - default=true)
kube:DRASchedulerFilterTimeout=true|false (BETA - default=true)
kube:DeclarativeValidation=true|false (BETA - default=true)
kube:DeclarativeValidationTakeover=true|false (BETA - default=false)
kube:DeploymentReplicaSetTerminatingReplicas=true|false (ALPHA - default=false)
kube:DetectCacheInconsistency=true|false (BETA - default=true)
kube:DisableCPUQuotaWithExclusiveCPUs=true|false (BETA - default=true)
kube:EnvFiles=true|false (ALPHA - default=false)
kube:EventedPLEG=true|false (ALPHA - default=false)
kube:ExternalServiceAccountTokenSigner=true|false (BETA - default=true)
kube:GracefulNodeShutdown=true|false (BETA - default=true)
kube:GracefulNodeShutdownBasedOnPodPriority=true|false (BETA - default=true)
kube:HPAConfigurableTolerance=true|false (ALPHA - default=false)
kube:HPAScaleToZero=true|false (ALPHA - default=false)
kube:HostnameOverride=true|false (ALPHA - default=false)
kube:ImageMaximumGCAge=true|false (BETA - default=true)
kube:ImageVolume=true|false (BETA - default=false)
kube:InOrderInformers=true|false (BETA - default=true)
kube:InPlacePodVerticalScaling=true|false (BETA - default=true)
kube:InPlacePodVerticalScalingExclusiveCPUs=true|false (ALPHA - default=false)
kube:InPlacePodVerticalScalingExclusiveMemory=true|false (ALPHA - default=false)
kube:InTreePluginPortworxUnregister=true|false (ALPHA - default=false)
kube:InformerResourceVersion=true|false (ALPHA - default=false)
kube:JobManagedBy=true|false (BETA - default=true)
kube:KubeletCrashLoopBackOffMax=true|false (ALPHA - default=false)
kube:KubeletEnsureSecretPulledImages=true|false (ALPHA - default=false)
kube:KubeletFineGrainedAuthz=true|false (BETA - default=true)
kube:KubeletInUserNamespace=true|false (ALPHA - default=false)
kube:KubeletPSI=true|false (BETA - default=true)
kube:KubeletPodResourcesDynamicResources=true|false (BETA - default=true)
kube:KubeletPodResourcesGet=true|false (BETA - default=true)
kube:KubeletSeparateDiskGC=true|false (BETA - default=true)
kube:KubeletServiceAccountTokenForCredentialProviders=true|false (BETA - default=true)
kube:ListFromCacheSnapshot=true|false (BETA - default=true)

kube:LocalStorageCapacityIsolationFSQuotaMonitoring=true|false (BETA - default=false)
kube:LoggingAlphaOptions=true|false (ALPHA - default=false)
kube:LoggingBetaOptions=true|false (BETA - default=true)
kube:MatchLabelKeysInPodTopologySpread=true|false (BETA - default=true)
kube:MatchLabelKeysInPodTopologySpreadSelectorMerge=true|false (BETA - default=true)
kube:MaxUnavailableStatefulSet=true|false (ALPHA - default=false)
kube:MemoryQoS=true|false (ALPHA - default=false)
kube:MutableCSINodeAllocatableCount=true|false (BETA - default=false)
kube:MutatingAdmissionPolicy=true|false (BETA - default=false)
kube:NodeLogQuery=true|false (BETA - default=false)
kube:NominatedNodeNameForExpectation=true|false (ALPHA - default=false)
kube:OpenAPIEnums=true|false (BETA - default=true)
kube:PodAndContainerStatsFromCRI=true|false (ALPHA - default=false)
kube:PodCertificateRequest=true|false (ALPHA - default=false)
kube:PodDeletionCost=true|false (BETA - default=true)
kube:PodLevelResources=true|false (BETA - default=true)
kube:PodLogsQuerySplitStreams=true|false (ALPHA - default=false)
kube:PodObservedGenerationTracking=true|false (BETA - default=true)
kube:PodReadyToStartContainersCondition=true|false (BETA - default=true)
kube:PodTopologyLabelsAdmission=true|false (ALPHA - default=false)
kube:PortForwardWebsockets=true|false (BETA - default=true)
kube:PreferSameTrafficDistribution=true|false (BETA - default=true)
kube:PreventStaticPodAPIReferences=true|false (BETA - default=true)
kube:ProcMountType=true|false (BETA - default=true)
kube:QOSReserved=true|false (ALPHA - default=false)
kube:ReduceDefaultCrashLoopBackOffDecay=true|false (ALPHA - default=false)
kube:RelaxedServiceNameValidation=true|false (ALPHA - default=false)
kube:ReloadKubeletServerCertificateFile=true|false (BETA - default=true)
kube:RemoteRequestHeaderUID=true|false (BETA - default=true)
kube:ResourceHealthStatus=true|false (ALPHA - default=false)
kube:RotateKubeletServerCertificate=true|false (BETA - default=true)
kube:RuntimeClassInImageCriApi=true|false (ALPHA - default=false)
kube:SELinuxChangePolicy=true|false (BETA - default=true)
kube:SELinuxMount=true|false (BETA - default=false)
kube:SELinuxMountReadWriteOncePod=true|false (BETA - default=true)
kube:SchedulerAsyncAPICalls=true|false (BETA - default=true)
kube:SchedulerAsyncPreemption=true|false (BETA - default=true)
kube:SchedulerPopFromBackoffQ=true|false (BETA - default=true)
kube:ServiceAccountNodeAudienceRestriction=true|false (BETA - default=true)
kube:SizeBasedListCostEstimate=true|false (BETA - default=true)
kube:StorageCapacityScoring=true|false (ALPHA - default=false)
kube:StorageVersionAPI=true|false (ALPHA - default=false)
kube:StorageVersionHash=true|false (BETA - default=true)
kube:StorageVersionMigrator=true|false (ALPHA - default=false)
kube:StrictIPCIDRValidation=true|false (ALPHA - default=false)
kube:StructuredAuthenticationConfigurationEgressSelector=true|false (BETA - default=true)
kube:SupplementalGroupsPolicy=true|false (BETA - default=true)
kube:SystemdWatchdog=true|false (BETA - default=true)
kube:TokenRequestServiceAccountUIDValidation=true|false (BETA - default=true)
kube:TopologyManagerPolicyAlphaOptions=true|false (ALPHA - default=false)
kube:TopologyManagerPolicyBetaOptions=true|false (BETA - default=true)
kube:TranslateStreamCloseWebsocketRequests=true|false (BETA - default=true)
kube:UnauthenticatedHTTP2DOSMitigation=true|false (BETA - default=true)
kube:UnknownVersionInteroperabilityProxy=true|false (ALPHA - default=false)
kube:UserNamespacesPodSecurityStandards=true|false (ALPHA - default=false)
kube:UserNamespacesSupport=true|false (BETA - default=true)
kube:WatchCacheInitializationPostStartHook=true|false (BETA - default=false)
kube:WatchList=true|false (BETA - default=true)
kube:WatchListClient=true|false (BETA - default=true)
kube:WindowsCPUAndMemoryAffinity=true|false (ALPHA - default=false)
kube:WindowsGracefulNodeShutdown=true|false (BETA - default=true)
--flex-volume-plugin-dir string Default: "/usr/libexec/kubernetes/kubelet-plugins/volume/exec/"

Full path of the directory in which the flex volume plugin should search for additional third party volume plugins.

-h, --help

help for kube-controller-manager

--horizontal-pod-autoscaler-cpu-initialization-period duration Default: 5m0s

The period after pod start when CPU samples might be skipped.

--horizontal-pod-autoscaler-downscale-stabilization duration Default: 5m0s

The period for which autoscaler will look backwards and not scale down below any recommendation it made during that period.

--horizontal-pod-autoscaler-initial-readiness-delay duration Default: 30s

The period after pod start during which readiness changes will be treated as initial readiness.

--horizontal-pod-autoscaler-sync-period duration Default: 15s

The period for syncing the number of pods in horizontal pod autoscaler.

--horizontal-pod-autoscaler-tolerance float Default: 0.1

The minimum change (from 1.0) in the desired-to-actual metrics ratio for the horizontal pod autoscaler to consider scaling.

--http2-max-streams-per-connection int

The limit that the server gives to clients for the maximum number of streams in an HTTP/2 connection. Zero means to use golang's default.

--kube-api-burst int32 Default: 30

Burst to use while talking with kubernetes apiserver.

--kube-api-content-type string Default: "application/vnd.kubernetes.protobuf"

Content type of requests sent to apiserver.

--kube-api-qps float Default: 20

QPS to use while talking with kubernetes apiserver.

--kubeconfig string

Path to kubeconfig file with authorization and master location information (the master location can be overridden by the master flag).

--large-cluster-size-threshold int32 Default: 50

Number of nodes from which node-lifecycle-controller treats the cluster as large for the eviction logic purposes. --secondary-node-eviction-rate is implicitly overridden to 0 for clusters this size or smaller. Notice: If nodes reside in multiple zones, this threshold will be considered as zone node size threshold for each zone to determine node eviction rate independently.

--leader-elect Default: true

Start a leader election client and gain leadership before executing the main loop. Enable this when running replicated components for high availability.

--leader-elect-lease-duration duration Default: 15s

The duration that non-leader candidates will wait after observing a leadership renewal until attempting to acquire leadership of a led but unrenewed leader slot. This is effectively the maximum duration that a leader can be stopped before it is replaced by another candidate. This is only applicable if leader election is enabled.

--leader-elect-renew-deadline duration Default: 10s

The interval between attempts by the acting master to renew a leadership slot before it stops leading. This must be less than the lease duration. This is only applicable if leader election is enabled.

--leader-elect-resource-lock string Default: "leases"

The type of resource object that is used for locking during leader election. Supported options are 'leases'.

--leader-elect-resource-name string Default: "kube-controller-manager"

The name of resource object that is used for locking during leader election.

--leader-elect-resource-namespace string Default: "kube-system"

The namespace of resource object that is used for locking during leader election.

--leader-elect-retry-period duration Default: 2s

The duration the clients should wait between attempting acquisition and renewal of a leadership. This is only applicable if leader election is enabled.

--leader-migration-config string

Path to the config file for controller leader migration, or empty to use the value that reflects default configuration of the controller manager. The config file should be of type LeaderMigrationConfiguration, group controllermanager.config.k8s.io, version v1alpha1.

--legacy-service-account-token-clean-up-period duration Default: 8760h0m0s

The period of time since the last usage of an legacy service account token before it can be deleted.

--log-flush-frequency duration Default: 5s

Maximum number of seconds between log flushes

--log-text-info-buffer-size quantity

[Alpha] In text format with split output streams, the info messages can be buffered for a while to increase performance. The default value of zero bytes disables buffering. The size can be specified as number of bytes (512), multiples of 1000 (1K), multiples of 1024 (2Ki), or powers of those (3M, 4G, 5Mi, 6Gi). Enable the LoggingAlphaOptions feature gate to use this.

--log-text-split-stream

[Alpha] In text format, write error messages to stderr and info messages to stdout. The default is to write a single stream to stdout. Enable the LoggingAlphaOptions feature gate to use this.

--logging-format string Default: "text"

Sets the log format. Permitted formats: "text".

--master string

The address of the Kubernetes API server (overrides any value in kubeconfig).

--max-endpoints-per-slice int32 Default: 100

The maximum number of endpoints that will be added to an EndpointSlice. More endpoints per slice will result in less endpoint slices, but larger resources. Defaults to 100.

--min-resync-period duration Default: 12h0m0s

The resync period in reflectors will be random between MinResyncPeriod and 2*MinResyncPeriod.

--mirroring-concurrent-service-endpoint-syncs int32 Default: 5

The number of service endpoint syncing operations that will be done concurrently by the endpointslice-mirroring-controller. Larger number = faster endpoint slice updating, but more CPU (and network) load. Defaults to 5.

--mirroring-endpointslice-updates-batch-period duration

The length of EndpointSlice updates batching period for endpointslice-mirroring-controller. Processing of EndpointSlice changes will be delayed by this duration to join them with potential upcoming updates and reduce the overall number of EndpointSlice updates. Larger number = higher endpoint programming latency, but lower number of endpoints revision generated

--mirroring-max-endpoints-per-subset int32 Default: 1000

The maximum number of endpoints that will be added to an EndpointSlice by the endpointslice-mirroring-controller. More endpoints per slice will result in less endpoint slices, but larger resources. Defaults to 100.

--namespace-sync-period duration Default: 5m0s

The period for syncing namespace life-cycle updates

--node-cidr-mask-size int32

Mask size for node cidr in cluster. Default is 24 for IPv4 and 64 for IPv6.

--node-cidr-mask-size-ipv4 int32

Mask size for IPv4 node cidr in dual-stack cluster. Default is 24.

--node-cidr-mask-size-ipv6 int32

Mask size for IPv6 node cidr in dual-stack cluster. Default is 64.

--node-eviction-rate float Default: 0.1

Number of nodes per second on which pods are deleted in case of node failure when a zone is healthy (see --unhealthy-zone-threshold for definition of healthy/unhealthy). Zone refers to entire cluster in non-multizone clusters.

--node-monitor-grace-period duration Default: 50s

Amount of time which we allow running Node to be unresponsive before marking it unhealthy. Must be N times more than kubelet's nodeStatusUpdateFrequency, where N means number of retries allowed for kubelet to post node status. This value should also be greater than the sum of HTTP2_PING_TIMEOUT_SECONDS and HTTP2_READ_IDLE_TIMEOUT_SECONDS

--node-monitor-period duration Default: 5s

The period for syncing NodeStatus in cloud-node-lifecycle-controller.

--node-startup-grace-period duration Default: 1m0s

Amount of time which we allow starting Node to be unresponsive before marking it unhealthy.

--permit-address-sharing

If true, SO_REUSEADDR will be used when binding the port. This allows binding to wildcard IPs like 0.0.0.0 and specific IPs in parallel, and it avoids waiting for the kernel to release sockets in TIME_WAIT state. [default=false]

--permit-port-sharing

If true, SO_REUSEPORT will be used when binding the port, which allows more than one instance to bind on the same address and port. [default=false]

--profiling Default: true

Enable profiling via web interface host:port/debug/pprof/

--pv-recycler-increment-timeout-nfs int32 Default: 30

the increment of time added per Gi to ActiveDeadlineSeconds for an NFS scrubber pod

--pv-recycler-minimum-timeout-hostpath int32 Default: 60

The minimum ActiveDeadlineSeconds to use for a HostPath Recycler pod. This is for development and testing only and will not work in a multi-node cluster.

--pv-recycler-minimum-timeout-nfs int32 Default: 300

The minimum ActiveDeadlineSeconds to use for an NFS Recycler pod

--pv-recycler-pod-template-filepath-hostpath string

The file path to a pod definition used as a template for HostPath persistent volume recycling. This is for development and testing only and will not work in a multi-node cluster.

--pv-recycler-pod-template-filepath-nfs string

The file path to a pod definition used as a template for NFS persistent volume recycling

--pv-recycler-timeout-increment-hostpath int32 Default: 30

the increment of time added per Gi to ActiveDeadlineSeconds for a HostPath scrubber pod. This is for development and testing only and will not work in a multi-node cluster.

--pvclaimbinder-sync-period duration Default: 15s

The period for syncing persistent volumes and persistent volume claims

--requestheader-allowed-names strings

List of client certificate common names to allow to provide usernames in headers specified by --requestheader-username-headers. If empty, any client certificate validated by the authorities in --requestheader-client-ca-file is allowed.

--requestheader-client-ca-file string

Root certificate bundle to use to verify client certificates on incoming requests before trusting usernames in headers specified by --requestheader-username-headers. WARNING: generally do not depend on authorization being already done for incoming requests.

--requestheader-extra-headers-prefix strings Default: "x-remote-extra-"

List of request header prefixes to inspect. X-Remote-Extra- is suggested.

--requestheader-group-headers strings Default: "x-remote-group"

List of request headers to inspect for groups. X-Remote-Group is suggested.

--requestheader-uid-headers strings

List of request headers to inspect for UIDs. X-Remote-Uid is suggested. Requires the RemoteRequestHeaderUID feature to be enabled.

--requestheader-username-headers strings Default: "x-remote-user"

List of request headers to inspect for usernames. X-Remote-User is common.

--resource-quota-sync-period duration Default: 5m0s

The period for syncing quota usage status in the system

--root-ca-file string

If set, this root certificate authority will be included in service account's token secret. This must be a valid PEM-encoded CA bundle.

--route-reconciliation-period duration Default: 10s

The period for reconciling routes created for Nodes by cloud provider.

--secondary-node-eviction-rate float Default: 0.01

Number of nodes per second on which pods are deleted in case of node failure when a zone is unhealthy (see --unhealthy-zone-threshold for definition of healthy/unhealthy). Zone refers to entire cluster in non-multizone clusters. This value is implicitly overridden to 0 if the cluster size is smaller than --large-cluster-size-threshold.

--secure-port int Default: 10257

The port on which to serve HTTPS with authentication and authorization. If 0, don't serve HTTPS at all.

--service-account-private-key-file string

Enables legacy secret-based tokens when set. Filename containing a PEM-encoded private RSA or ECDSA key used to sign service account tokens.

--service-cluster-ip-range string

CIDR Range for Services in cluster. Only used when --allocate-node-cidrs=true; if false, this option will be ignored.

--show-hidden-metrics-for-version string

The previous version for which you want to show hidden metrics. Only the previous minor version is meaningful, other values will not be allowed. The format is <major>.<minor>, e.g.: '1.16'. The purpose of this format is make sure you have the opportunity to notice if the next release hides additional metrics, rather than being surprised when they are permanently removed in the release after that.

--terminated-pod-gc-threshold int32 Default: 12500

Number of terminated pods that can exist before the terminated pod garbage collector starts deleting terminated pods. If <= 0, the terminated pod garbage collector is disabled.

--tls-cert-file string

File containing the default x509 Certificate for HTTPS. (CA cert, if any, concatenated after server cert). If HTTPS serving is enabled, and --tls-cert-file and --tls-private-key-file are not provided, a self-signed certificate and key are generated for the public address and saved to the directory specified by --cert-dir.

--tls-cipher-suites strings

Comma-separated list of cipher suites for the server. If omitted, the default Go cipher suites will be used.
Preferred values: TLS_AES_128_GCM_SHA256, TLS_AES_256_GCM_SHA384, TLS_CHACHA20_POLY1305_SHA256,

TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305, TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256. Insecure values: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_ECDSA_WITH_RC4_128_SHA, TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256, TLS_ECDHE_RSA_WITH_RC4_128_SHA, TLS_RSA_WITH_3DES_EDE_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA, TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_RSA_WITH_AES_256_CBC_SHA, TLS_RSA_WITH_AES_256_GCM_SHA384, TLS_RSA_WITH_RC4_128_SHA.

--tls-min-version string

Minimum TLS version supported. Possible values: VersionTLS10, VersionTLS11, VersionTLS12, VersionTLS13

--tls-private-key-file string

File containing the default x509 private key matching --tls-cert-file.

--tls-sni-cert-key string

A pair of x509 certificate and private key file paths, optionally suffixed with a list of domain patterns which are fully qualified domain names, possibly with prefixed wildcard segments. The domain patterns also allow IP addresses, but IPs should only be used if the apiserver has visibility to the IP address requested by a client. If no domain patterns are provided, the names of the certificate are extracted. Non-wildcard matches trump over wildcard matches, explicit domain patterns trump over extracted names. For multiple key/certificate pairs, use the --tls-sni-cert-key multiple times. Examples: "example.crt,example.key" or "foo.crt,foo.key:*.foo.com,foo.com".

--unhealthy-zone-threshold float Default: 0.55

Fraction of Nodes in a zone which needs to be not Ready (minimum 3) for zone to be treated as unhealthy.

--use-service-account-credentials

If true, use individual service account credentials for each controller.

-v, --v int

number for the log level verbosity

--version version[=true]

--version, --version=raw prints version information and quits; --version=vX.Y.Z... sets the reported version

--vmodule pattern=N,...

comma-separated list of pattern=N settings for file-filtered logging (only works for text log format)