IBM Cloud Pak AIOps 4.10.0 - PDF 2

© Copyright IBM Corp. 2025.
US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Tables of Contents

APIs 1

Custom resource definitions 1

IBM Cloud Pak for AIOps CRDs 1

AIModelUI-v1 3

AIOpsEdge-v1beta1 5

AIOpsUI-v1 112

AIOpsUIExtension-v1 119

AIManager-v1beta1 119

Algorithm-v1alpha1 123

Algorithm-v1beta1 124

ASM-v1beta1.md 125

BaseUI-v1 129

BundleManifest-v1alpha1 131

BundleManifest-v1beta1 132

ConnectorUI-v1 134

InsightsUI-v1 135

Installation-v1alpha1 140

Tunnel-v1 143

Infrastructure Automation CRDs 147

IAConfig 147

IAManageService 148

IMInstall 154

Accessing APIs 158

Generating API keys for authentication 162

Integrated UI APIs 163

Creating custom home page cards with API 163

Identity Management (IM) APIs 174

Preparing to run API commands 174

OIDC Registration APIs 176

Identity Provider APIs 187

Issue resolution API 222

Probable cause customization API 224

Change risk API 228

Metric API 233

Automation (runbook) API 238

Topology APIs 239

Accessing topology service Swagger UI 240

Enabling access to URL routes 242

Topology API reference 243

Properties 244

Edge labels 247

Edge types 249

Entity types 252

Status (and state) 256

Timestamps 258

Topology examples 259

Virtual machine example 259

OpenStack example 262
Physical device example 270
Infrastructure Automation APIs 274

APIs

IBM Cloud Pak® for AIOps offers a set of tools and APIs that you can use to predict, communicate, and resolve events before they become serious problems for your organization.

Learn how to use the tools and APIs that work with IBM Cloud Pak for AIOps in the following topics:

- Custom resource definitions
- Accessing APIs
- Integrated UI APIs
- User management (IAM) APIs
- licensing APIs
- Issue resolution API
- Probable Cause Customization API
- Change Risk API
- Metric API
- Automation (runbook) API
- Topology APIs
- Infrastructure Management API Guide
- Managed services APIs

Important: You can directly access the Swagger UI for only the Topology API and Metric API.

- For more information about accessing Metric service Swagger UI, see Accessing Metric service Swagger UI.
- For more information about accessing topology service Swagger UI, see Accessing Topology service Swagger UI.

Custom resource definitions

Use the following information to learn about the custom resource definitions (CRDs) used by IBM Cloud Pak for AIOps and Infrastructure Automation.

For more information about API conventions and definitions, see API Conventions in the Kubernetes documentation.

- IBM Cloud Pak for AIOps custom resource definitions
- Infrastructure Automation custom resource definitions

IBM Cloud Pak for AIOps custom resource definitions

Use this information to learn about the custom resource definitions (CRDs) used by IBM Cloud Pak for AIOps.

In the following table, CRDs that do not have a hyperlink must not be manually edited unless otherwise instructed, and are manipulated only by the operator.

Table 1. IBM Cloud Pak for AIOps CRDs

Kind	API version	Operator name	Description
AIManagerMainProd	v1beta1	aimanager-operator	AIManagerMainProd is the Schema for the aimanagermainprods API. For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.
AIManager	v1beta1	aimanager-operator	AIManager Deploys the IBM AIOps AI Manager service components of IBM Cloud Pak for AIOps. For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.
Algorithm	v1alpha1	aimanager-operator	Algorithm is the Schema for the algorithms API
Algorithm	v1beta1	aimanager-operator	Algorithm is the Schema for the algorithms API
AIOpsEdge	v1beta1	aiopsedge-operator	AIOpsEdge is the Schema for the aiopsedges API
AIOpsKafkaTopic	v1beta1	aiopsedge-operator	AIOpsKafkaTopic is the Schema for the aiopskafkatopics API
AutomationAction	v1beta1	aiopsedge-operator	AutomationAction is the Schema for the automationactions API
BundleManifest	v1alpha1	aiopsedge-operator	BundleManifest is the Schema for the bundlemanifests API
BundleManifest	v1beta1	aiopsedge-operator	BundleManifest is the Schema for the bundlemanifests API
ConnectorComponent	v1beta1	aiopsedge-operator	ConnectorComponent is the Schema for the connectorcomponents API
ConnectorConfiguration	v1beta1	aiopsedge-operator	ConnectorConfiguration is the Schema for the connectorconfigurations API
ConnectorSchema	v1beta1	aiopsedge-operator	ConnectorSchema is the Schema for the connectorschemas API
GitApp	v1beta1	aiopsedge-operator	GitApp is the Schema for the GitApp API
MicroEdgeConfiguration	v1beta1	aiopsedge-operator	MicroEdgeConfiguration is the Schema for the microedgeconfigurations API
ASMFormation	v1beta1	asm-operator	Create a formation deployment. Documentation This product is not deployable standalone and it is part of the LICCR already accepted
ASM	v1beta1	asm-operator	(No Description)
AIOpsAnalyticsOrchestrator	v1beta1	ibm-aiops-ir-ai	AIOpsAnalyticsOrchestrator is the Schema for the aiopsanalyticsorchestrators API. This product is not deployable standalone and it is part of the LICCR already accepted
IssueResolutionCore	v1beta1	ibm-aiops-ir-core	IssueResolutionCore installs core components required for the issue resolution capabilities of IBM Cloud Pak for AIOps. This product is not deployable standalone and it is part of the LICCR already accepted.

Kind	API version	Operator name	Description
LifecycleService	v1beta1	ibm-aiops-ir-lifecycle	LifecycleService is the Schema for the lifecycleservices API
LifecycleTrigger	v1beta1	ibm-aiops-ir-lifecycle	LifecycleTrigger is the Schema for the lifecycletriggers API
Installation	v1alpha1	ibm-aiops-orchestrator	Leverage advanced, explainable AI to resolve incidents faster, gain visibility into critical workloads, and intelligently manage key application resources. Learn More. By installing this product you accept the license terms.
Tunnel	v1	ibm-secure-tunnel-operator	Tunnel is the Schema for the tunnels API
AIModelUI	v1	ibm-watson-aiops-ui-operator	For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.
AIOpsUIExtension	v1	ibm-watson-aiops-ui-operator	AIOpsUIExtension is the Schema for the aiopsuiextensions API
AIOpsUI	v1	ibm-watson-aiops-ui-operator	For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.
AIOpsUI is the Schema for the aiopsuis API.			
BaseUI	v1	ibm-watson-aiops-ui-operator	For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.
ConnectorUI	v1	ibm-watson-aiops-ui-operator	For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.
InsightsUI	v1	ibm-watson-aiops-ui-operator	For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.

AIModelUI/v1

For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.

spec

Description: CommonSpecs - specs that are common to determining a deployment.

Type: object

Table 1. Description of AIModelUI/v1 spec

Property	Type	Description

Property	Type	Description
spec	object	CommonSpecs - specs that are common to determining a deployment.
spec.configuration	object	Additional configuration for the UI deployment
spec.configuration.topologyModel	string	TopologyModel - Specify the terminology to use in CP4WAIOps when defining collections of topology resource groups. Valid values are "application" or "service". If this setting is not specified, the default value will be "application".
spec.container	object	ContainerSpec - possible configurations around containers
spec.container.containerPort	integer	(No Description)
spec.container.env	array	(No Description)
spec.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.image	object	ImageSpec - specification around how to pull an image.
spec.image.path	string	(No Description)
spec.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.image.pullSecret	string	(No Description)
spec.license	object	License - license spec determines whether a license has been accepted
spec.license.accept	boolean	License acceptance switch
spec.name	string	Name of the service being deployed
spec.replicas	integer	(No Description)
spec.size	string	(No Description)
spec.tlsSecretName	string	(No Description)
spec.version	string	Version of the csv deploying
spec.zenserviceName	string	name of the zenservice cr that is used for this deployment

status

Description: (No Description)

Type: object

Table 1. Description of AIModelUI/v1 status

Property	Type	Description
status	object	(No Description)
status.conditions	array	(No Description)
status.dependencies	object	Dependencies is a ad hoc field to store known depedency status. This field currently does not store all dependencies, but rather only depedencies that we have a vested interest in knowing the status of.
status.managedResources	array	(No Description)
status.phase	string	(No Description)
status.versions	object	(No Description)
status.versions.reconciled	string	(No Description)

AIOpsEdge/v1beta1

AIOpsEdge is the Schema for the aiopsedges API

spec

Description: AIOpsEdgeSpec defines the desired state of AIOpsEdge

Type: object

Table 1. Description of AIOpsEdge/v1beta1 spec

Property	Type	Description
spec	object	AIOpsEdgeSpec defines the desired state of AIOpsEdge
spec.certificates	object	Certificates contains information required to create and trust certificates for internal TLS communication.
spec.certificates.caConfigMapKey	string	Key in the ConfigMap identified by caConfigMapName that contains root certificates to be trusted for internal TLS.
spec.certificates.caConfigMapName	string	Name of the ConfigMap that contains root certificates to be trusted for internal TLS.
spec.certificates.issuer	string	Name of the cert-manager.io/Issuer that will be used to create TLS certificates.
spec.disableDefaultConnectors	boolean	DisableDefaultConnectors can be enabled to prevent the installation of default connectors
spec.elasticsearchSecret	string	The secret name for ElasticSearch. In future releases, OpenSearch will replace ElasticSearch, which is why this is optional. OpenSearch's secret does not depend on the installation name

Property	Type	Description
spec.kafka	object	The specification of the Kafka and ZooKeeper clusters, and Topic Operator.
spec.kafka.clientsCa	object	Configuration of the clients certificate authority.
spec.kafka.clientsCa.certificateExpirationPolicy	string	How should CA certificate expiration be handled when `generateCertificateAuthority=true`. The default is for a new CA certificate to be generated reusing the existing private key.
spec.kafka.clientsCa.generateCertificateAuthority	boolean	If true then Certificate Authority certificates will be generated automatically. Otherwise the user will need to provide a Secret with the CA certificate. Default is true.
spec.kafka.clientsCa.generateSecretOwnerReference	boolean	If true then the Certificate Authority certificates secrets owner reference is set to the Kafka object. Otherwise no owner reference is set so deleting the Kafka object won't delete the secrets. Default is true.
spec.kafka.clientsCa.renewalDays	integer	The number of days in the certificate renewal period. This is the number of days before the a certificate expires during which renewal actions may be performed. When `generateCertificateAuthority` is true, this will cause the generation of a new certificate. When `generateCertificateAuthority` is true, this will cause extra logging at WARN level about the pending certificate expiry. Default is 30.
spec.kafka.clientsCa.validityDays	integer	The number of days generated certificates should be valid for. The default is 365.
spec.kafka.clusterCa	object	Configuration of the cluster certificate authority.
spec.kafka.clusterCa.certificateExpirationPolicy	string	How should CA certificate expiration be handled when `generateCertificateAuthority=true`. The default is for a new CA certificate to be generated reusing the existing private key.
spec.kafka.clusterCa.generateCertificateAuthority	boolean	If true then Certificate Authority certificates will be generated automatically. Otherwise the user will need to provide a Secret with the CA certificate. Default is true.
spec.kafka.clusterCa.generateSecretOwnerReference	boolean	If true then the Certificate Authority certificates secrets owner reference is set to the Kafka object. Otherwise no owner reference is set so deleting the Kafka object won't delete the secrets. Default is true.
spec.kafka.clusterCa.renewalDays	integer	The number of days in the certificate renewal period. This is the number of days before the a certificate expires during which renewal actions may be performed. When `generateCertificateAuthority` is true, this will cause the generation of a new certificate. When `generateCertificateAuthority` is true, this will cause extra logging at WARN level about the pending certificate expiry. Default is 30.
spec.kafka.clusterCa.validityDays	integer	The number of days generated certificates should be valid for. The default is 365.

Property	Type	Description
spec.kafka.cruiseControl	object	Configuration for Cruise Control deployment. Deploys a Cruise Control instance when specified.
spec.kafka.cruiseControl.brokerCapacity	object	The Cruise Control `brokerCapacity` configuration.
spec.kafka.cruiseControl.brokerCapacity.cpuUtilization	integer	Broker capacity for CPU resource utilization as a percentage (0 - 100).
spec.kafka.cruiseControl.brokerCapacity.disk	string	Broker capacity for disk in bytes, for example, 100Gi.
spec.kafka.cruiseControl.brokerCapacity.inboundNetwork	string	Broker capacity for inbound network throughput in bytes per second, for example, 10000KB/s.
spec.kafka.cruiseControl.brokerCapacity.outboundNetwork	string	Broker capacity for outbound network throughput in bytes per second, for example 10000KB/s.
spec.kafka.cruiseControl.config	object	The Cruise Control configuration. For a full list of configuration options refer to https://github.com/linkedin/cruise-control/wiki/Configurations. Note that properties with the following prefixes cannot be set: bootstrap.servers, client.id, zookeeper., network., security., failed.brokers.zk.path,webserver.http., webserver.api.urlprefix, webserver.session.path, webserver.accesslog., two.step., request.reason.required,metric.reporter.sampler.bootstrap.servers, metric.reporter.topic, partition.metric.sample.store.topic, broker.metric.sample.store.topic,capacity.config.file, self.healing., anomaly.detection., ssl. (with the exception of: ssl.cipher.suites, ssl.protocol, ssl.enabled.protocols, webserver.http.cors.enabled,webserver.http.cors.origin, webserver.http.cors.exposeheaders).
spec.kafka.cruiseControl.image	string	The docker image for the pods.
spec.kafka.cruiseControl.jvmOptions	object	JVM Options for the Cruise Control container.
spec.kafka.cruiseControl.jvmOptions.-XX	object	A map of -XX options to the JVM.
spec.kafka.cruiseControl.jvmOptions.-Xms	string	-Xms option to to the JVM.
spec.kafka.cruiseControl.jvmOptions.-Xmx	string	-Xmx option to to the JVM.

Property	Type	Description
spec.kafka.cruiseControl.jvmOptions.gcLoggingEnabled	boolean	Specifies whether the Garbage Collection logging is enabled. The default is false.
spec.kafka.cruiseControl.jvmOptions.javaSystemProperties	array	A map of additional system properties which will be passed using the `-D` option to the JVM.
spec.kafka.cruiseControl.livenessProbe	object	Pod liveness checking for the Cruise Control container.
spec.kafka.cruiseControl.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.cruiseControl.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.cruiseControl.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.cruiseControl.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.cruiseControl.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.cruiseControl.logging	object	Logging configuration (log4j1) for Cruise Control.
spec.kafka.cruiseControl.logging.loggers	object	A Map from logger name to logger level.
spec.kafka.cruiseControl.logging.name	string	The name of the `ConfigMap` from which to get the logging configuration.
spec.kafka.cruiseControl.logging.type	string	Logging type, must be either 'inline' or 'external'.
spec.kafka.cruiseControl.metrics	object	The Prometheus JMX Exporter configuration. See https://github.com/prometheus/jmx_exporter for details of the structure of this configuration.
spec.kafka.cruiseControl.metricsConfig	object	Metrics configuration.

Property	Type	Description
spec.kafka.cruiseControl.metricsConfig.type	string	Metrics type. Only 'jmxPrometheusExporter' supported currently.
spec.kafka.cruiseControl.metricsConfig.valueFrom	object	ConfigMap where the Prometheus JMX Exporter configuration is stored. For details of the structure of this configuration, see the {JMXExporter}.
spec.kafka.cruiseControl.metricsConfig.valueFrom.configMapKeyRef	object	Reference to the key in the ConfigMap containing the metrics configuration.
spec.kafka.cruiseControl.metricsConfig.valueFrom.configMapKeyRef.key	string	Key corresponds to the JSON schema field "key".
spec.kafka.cruiseControl.metricsConfig.valueFrom.configMapKeyRef.name	string	Name corresponds to the JSON schema field "name".
spec.kafka.cruiseControl.metricsConfig.valueFrom.configMapKeyRef.optional	boolean	Optional corresponds to the JSON schema field "optional".
spec.kafka.cruiseControl.readinessProbe	object	Pod readiness checking for the Cruise Control container.
spec.kafka.cruiseControl.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.cruiseControl.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.cruiseControl.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.cruiseControl.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.cruiseControl.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.

Property	Type	Description
spec.kafka.cruiseControl.resources	object	CPU and memory resources to reserve for the Cruise Control container.
spec.kafka.cruiseControl.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.cruiseControl.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.cruiseControl.template	object	Template to specify how Cruise Control resources, `Deployments` and `Pods`, are generated.
spec.kafka.cruiseControl.template.apiService	object	Template for Cruise Control API `Service`.
spec.kafka.cruiseControl.template.apiService.metadata	object	Metadata applied to the resource.
spec.kafka.cruiseControl.template.apiService.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.cruiseControl.template.apiService.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.cruiseControl.template.cruiseControlContainer	object	Template for the Cruise Control container.
spec.kafka.cruiseControl.template.cruiseControlContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext	object	Security context for the container.
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".

Property	Type	Description
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".

Property	Type	Description
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".

Property	Type	Description
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.cruiseControl.template.cruiseControlContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.cruiseControl.template.deployment	object	Template for Cruise Control **`Deployment`**.
spec.kafka.cruiseControl.template.deployment.metadata	object	Metadata applied to the resource.
spec.kafka.cruiseControl.template.deployment.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.cruiseControl.template.deployment.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.

Property	Type	Description
spec.kafka.cruiseControl.template.pod	object	Template for Cruise Control `Pods`.
spec.kafka.cruiseControl.template.pod.affinity	object	The pod's affinity rules.
spec.kafka.cruiseControl.template.pod.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.cruiseControl.template.pod.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.cruiseControl.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.cruiseControl.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.cruiseControl.template.pod.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.cruiseControl.template.pod.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.cruiseControl.template.pod.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.cruiseControl.template.pod.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.cruiseControl.template.pod.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.cruiseControl.template.pod.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.cruiseControl.template.pod.hostAliases	array	The pod's HostAliases. HostAliases is an optional list of hosts and IPs that will be injected into the pod's hosts file if specified.
spec.kafka.cruiseControl.template.pod.imagePullSecrets	array	List of references to secrets in the same namespace to use for pulling any of the images used by this Pod. When the **`STRIMZI_IMAGE_PULL_SECRETS`** environment variable in Cluster Operator and the **`imagePullSecrets`** option are specified, only the **`imagePullSecrets`** variable is used and the **`STRIMZI_IMAGE_PULL_SECRETS`** variable is ignored.
spec.kafka.cruiseControl.template.pod.metadata	object	Metadata applied to the resource.
spec.kafka.cruiseControl.template.pod.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.cruiseControl.template.pod.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.cruiseControl.template.pod.priorityClassName	string	The name of the priority class used to assign priority to the pods. For more information about priority classes, see {K8sPriorityClass}.

Property	Type	Description
spec.kafka.cruiseControl.template.pod.schedulerName	string	The name of the scheduler used to dispatch this `Pod`. If not specified, the default scheduler will be used.
spec.kafka.cruiseControl.template.pod.securityContext	object	Configures pod-level security attributes and common container settings.
spec.kafka.cruiseControl.template.pod.securityContext.fsGroup	integer	FsGroup corresponds to the JSON schema field "fsGroup".
spec.kafka.cruiseControl.template.pod.securityContext.fsGroupChangePolicy	string	FsGroupChangePolicy corresponds to the JSON schema field "fsGroupChangePolicy".
spec.kafka.cruiseControl.template.pod.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.cruiseControl.template.pod.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.cruiseControl.template.pod.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.cruiseControl.template.pod.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.cruiseControl.template.pod.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.cruiseControl.template.pod.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".

Property	Type	Description
spec.kafka.cruiseControl.template.pod.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.cruiseControl.template.pod.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.cruiseControl.template.pod.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.cruiseControl.template.pod.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.cruiseControl.template.pod.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.cruiseControl.template.pod.securityContext.supplementalGroups	array	SupplementalGroups corresponds to the JSON schema field "supplementalGroups".
spec.kafka.cruiseControl.template.pod.securityContext.sysctls	array	Sysctls corresponds to the JSON schema field "sysctls".
spec.kafka.cruiseControl.template.pod.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.cruiseControl.template.pod.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".

Property	Type	Description
spec.kafka.cruiseControl.template.pod.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.cruiseControl.template.pod.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.cruiseControl.template.pod.terminationGracePeriodSeconds	integer	The grace period is the duration in seconds after the processes running in the pod are sent a termination signal, and the time when the processes are forcibly halted with a kill signal. Set this value to longer than the expected cleanup time for your process. Value must be a non-negative integer. A zero value indicates delete immediately. You might need to increase the grace period for very large Kafka clusters, so that the Kafka brokers have enough time to transfer their work to another broker before they are terminated. Defaults to 30 seconds.
spec.kafka.cruiseControl.template.pod.tolerations	array	The pod's tolerations.
spec.kafka.cruiseControl.template.pod.topologySpreadConstraints	array	The pod's topology spread constraints.
spec.kafka.cruiseControl.template.podDisruptionBudget	object	Template for Cruise Control **`PodDisruptionBudget`**.
spec.kafka.cruiseControl.template.podDisruptionBudget.maxUnavailable	integer	Maximum number of unavailable pods to allow automatic Pod eviction. A Pod eviction is allowed when the **`maxUnavailable`** number of pods or fewer are unavailable after the eviction. Setting this value to 0 prevents all voluntary evictions, so the pods must be evicted manually. Defaults to 1.
spec.kafka.cruiseControl.template.podDisruptionBudget.metadata	object	Metadata to apply to the **`PodDistruptionBugetTemplate`** resource.
spec.kafka.cruiseControl.template.podDisruptionBudget.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.

Property	Type	Description
spec.kafka.cruiseControl.template.podDisruptionBudget.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.cruiseControl.template.tlsSidecarContainer	object	Template for the Cruise Control TLS sidecar container.
spec.kafka.cruiseControl.template.tlsSidecarContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext	object	Security context for the container.
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".

Property	Type	Description
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".

Property	Type	Description
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".

Property	Type	Description
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.cruiseControl.template.tlsSidecarContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.cruiseControl.tlsSidecar	object	TLS sidecar configuration.
spec.kafka.cruiseControl.tlsSidecar.image	string	The docker image for the container.
spec.kafka.cruiseControl.tlsSidecar.livenessProbe	object	Pod liveness checking.
spec.kafka.cruiseControl.tlsSidecar.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.cruiseControl.tlsSidecar.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.cruiseControl.tlsSidecar.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.cruiseControl.tlsSidecar.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.cruiseControl.tlsSidecar.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.

Property	Type	Description
spec.kafka.cruiseControl.tlsSidecar.logLevel	string	The log level for the TLS sidecar. Default value is `notice`.
spec.kafka.cruiseControl.tlsSidecar.readinessProbe	object	Pod readiness checking.
spec.kafka.cruiseControl.tlsSidecar.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.cruiseControl.tlsSidecar.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.cruiseControl.tlsSidecar.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.cruiseControl.tlsSidecar.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.cruiseControl.tlsSidecar.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.cruiseControl.tlsSidecar.resources	object	CPU and memory resources to reserve.
spec.kafka.cruiseControl.tlsSidecar.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.cruiseControl.tlsSidecar.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.entityOperator	object	Configuration of the Entity Operator.
spec.kafka.entityOperator.affinity	object	The pod's affinity rules.

Property	Type	Description
spec.kafka.entityOperator.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.entityOperator.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.entityOperator.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.entityOperator.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.entityOperator.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.entityOperator.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.template	object	Template for Entity Operator resources. The template allows users to specify how is the **`Deployment`** and **`Pods`** generated.
spec.kafka.entityOperator.template.deployment	object	Template for Entity Operator **`Deployment`**.
spec.kafka.entityOperator.template.deployment.metadata	object	Metadata applied to the resource.
spec.kafka.entityOperator.template.deployment.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.entityOperator.template.deployment.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.entityOperator.template.pod	object	Template for Entity Operator **`Pods`**.
spec.kafka.entityOperator.template.pod.affinity	object	The pod's affinity rules.
spec.kafka.entityOperator.template.pod.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.entityOperator.template.pod.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.entityOperator.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.entityOperator.template.pod.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.entityOperator.template.pod.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.template.pod.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.template.pod.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.entityOperator.template.pod.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.template.pod.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.entityOperator.template.pod.hostAliases	array	The pod's HostAliases. HostAliases is an optional list of hosts and IPs that will be injected into the pod's hosts file if specified.

Property	Type	Description
spec.kafka.entityOperator.template.pod.imagePullSecrets	array	List of references to secrets in the same namespace to use for pulling any of the images used by this Pod. When the **`STRIMZI_IMAGE_PULL_SECRETS`** environment variable in Cluster Operator and the **`imagePullSecrets`** option are specified, only the **`imagePullSecrets`** variable is used and the **`STRIMZI_IMAGE_PULL_SECRETS`** variable is ignored.
spec.kafka.entityOperator.template.pod.metadata	object	Metadata applied to the resource.
spec.kafka.entityOperator.template.pod.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.entityOperator.template.pod.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.entityOperator.template.pod.priorityClassName	string	The name of the priority class used to assign priority to the pods. For more information about priority classes, see {K8sPriorityClass}.
spec.kafka.entityOperator.template.pod.schedulerName	string	The name of the scheduler used to dispatch this **`Pod`**. If not specified, the default scheduler will be used.
spec.kafka.entityOperator.template.pod.securityContext	object	Configures pod-level security attributes and common container settings.
spec.kafka.entityOperator.template.pod.securityContext.fsGroup	integer	FsGroup corresponds to the JSON schema field "fsGroup".
spec.kafka.entityOperator.template.pod.securityContext.fsGroupChangePolicy	string	FsGroupChangePolicy corresponds to the JSON schema field "fsGroupChangePolicy".
spec.kafka.entityOperator.template.pod.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.entityOperator.template.pod.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".

Property	Type	Description
spec.kafka.entityOperator.template.pod.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.entityOperator.template.pod.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.entityOperator.template.pod.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.entityOperator.template.pod.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.entityOperator.template.pod.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.entityOperator.template.pod.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.entityOperator.template.pod.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.entityOperator.template.pod.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.entityOperator.template.pod.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".

Property	Type	Description
spec.kafka.entityOperator.template.pod.securityContext.supplementalGroups	array	SupplementalGroups corresponds to the JSON schema field "supplementalGroups".
spec.kafka.entityOperator.template.pod.securityContext.sysctls	array	Sysctls corresponds to the JSON schema field "sysctls".
spec.kafka.entityOperator.template.pod.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.entityOperator.template.pod.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.entityOperator.template.pod.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.entityOperator.template.pod.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.entityOperator.template.pod.terminationGracePeriodSeconds	integer	The grace period is the duration in seconds after the processes running in the pod are sent a termination signal, and the time when the processes are forcibly halted with a kill signal. Set this value to longer than the expected cleanup time for your process. Value must be a non-negative integer. A zero value indicates delete immediately. You might need to increase the grace period for very large Kafka clusters, so that the Kafka brokers have enough time to transfer their work to another broker before they are terminated. Defaults to 30 seconds.
spec.kafka.entityOperator.template.pod.tolerations	array	The pod's tolerations.
spec.kafka.entityOperator.template.pod.topologySpreadConstraints	array	The pod's topology spread constraints.

Property	Type	Description
spec.kafka.entityOperator.template.tlsSidecarContainer	object	Template for the Entity Operator TLS sidecar container.
spec.kafka.entityOperator.template.tlsSidecarContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext	object	Security context for the container.
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".

Property	Type	Description
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".

Property	Type	Description
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".

Property	Type	Description
spec.kafka.entityOperator.template.tlsSidecarContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.entityOperator.template.topicOperatorContainer	object	Template for the Entity Topic Operator container.
spec.kafka.entityOperator.template.topicOperatorContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext	object	Security context for the container.
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".

Property	Type	Description
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".

Property	Type	Description
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".

Property	Type	Description
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.entityOperator.template.topicOperatorContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.entityOperator.template.userOperatorContainer	object	Template for the Entity User Operator container.
spec.kafka.entityOperator.template.userOperatorContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.entityOperator.template.userOperatorContainer.securityContext	object	Security context for the container.
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".

Property	Type	Description
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".

Property	Type	Description
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".

Property	Type	Description
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.entityOperator.template.userOperatorContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.entityOperator.tlsSidecar	object	TLS sidecar configuration.
spec.kafka.entityOperator.tlsSidecar.image	string	The docker image for the container.
spec.kafka.entityOperator.tlsSidecar.livenessProbe	object	Pod liveness checking.
spec.kafka.entityOperator.tlsSidecar.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.

Property	Type	Description
spec.kafka.entityOperator.tlsSidecar.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.logLevel	string	The log level for the TLS sidecar. Default value is `notice`.
spec.kafka.entityOperator.tlsSidecar.readinessProbe	object	Pod readiness checking.
spec.kafka.entityOperator.tlsSidecar.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.entityOperator.tlsSidecar.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.entityOperator.tlsSidecar.resources	object	CPU and memory resources to reserve.

Property	Type	Description
spec.kafka.entityOperator.tlsSidecar.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.entityOperator.tlsSidecar.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.entityOperator.tolerations	array	The pod's tolerations.
spec.kafka.entityOperator.topicOperator	object	Configuration of the Topic Operator.
spec.kafka.entityOperator.topicOperator.image	string	The image to use for the Topic Operator.
spec.kafka.entityOperator.topicOperator.jvmOptions	object	JVM Options for pods.
spec.kafka.entityOperator.topicOperator.jvmOptions.-XX	object	A map of -XX options to the JVM.
spec.kafka.entityOperator.topicOperator.jvmOptions.-Xms	string	-Xms option to to the JVM.
spec.kafka.entityOperator.topicOperator.jvmOptions.-Xmx	string	-Xmx option to to the JVM.
spec.kafka.entityOperator.topicOperator.jvmOptions.gcLoggingEnabled	boolean	Specifies whether the Garbage Collection logging is enabled. The default is false.
spec.kafka.entityOperator.topicOperator.jvmOptions.javaSystemProperties	array	A map of additional system properties which will be passed using the `-D` option to the JVM.
spec.kafka.entityOperator.topicOperator.livenessProbe	object	Pod liveness checking.

Property	Type	Description
spec.kafka.entityOperator.topicOperator.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.entityOperator.topicOperator.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.entityOperator.topicOperator.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.entityOperator.topicOperator.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.entityOperator.topicOperator.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.entityOperator.topicOperator.logging	object	Logging configuration.
spec.kafka.entityOperator.topicOperator.logging.loggers	object	A Map from logger name to logger level.
spec.kafka.entityOperator.topicOperator.logging.name	string	The name of the **`ConfigMap`** from which to get the logging configuration.
spec.kafka.entityOperator.topicOperator.logging.type	string	Logging type, must be either 'inline' or 'external'.
spec.kafka.entityOperator.topicOperator.readinessProbe	object	Pod readiness checking.
spec.kafka.entityOperator.topicOperator.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.

Property	Type	Description
spec.kafka.entityOperator.topicOperator.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.entityOperator.topicOperator.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.entityOperator.topicOperator.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.entityOperator.topicOperator.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.entityOperator.topicOperator.reconciliationIntervalSeconds	integer	Interval between periodic reconciliations.
spec.kafka.entityOperator.topicOperator.resources	object	CPU and memory resources to reserve.
spec.kafka.entityOperator.topicOperator.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.entityOperator.topicOperator.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.entityOperator.topicOperator.topicMetadataMaxAttempts	integer	The number of attempts at getting topic metadata.
spec.kafka.entityOperator.topicOperator.watchedNamespace	string	The namespace the Topic Operator should watch.

Property	Type	Description
spec.kafka.entityOperator.topicOperator.zookeeperSessionTimeoutSeconds	integer	Timeout for the ZooKeeper session.
spec.kafka.entityOperator.userOperator	object	Configuration of the User Operator.
spec.kafka.entityOperator.userOperator.image	string	The image to use for the User Operator.
spec.kafka.entityOperator.userOperator.jvmOptions	object	JVM Options for pods.
spec.kafka.entityOperator.userOperator.jvmOptions.-XX	object	A map of -XX options to the JVM.
spec.kafka.entityOperator.userOperator.jvmOptions.-Xms	string	-Xms option to to the JVM.
spec.kafka.entityOperator.userOperator.jvmOptions.-Xmx	string	-Xmx option to to the JVM.
spec.kafka.entityOperator.userOperator.jvmOptions.gcLoggingEnabled	boolean	Specifies whether the Garbage Collection logging is enabled. The default is false.
spec.kafka.entityOperator.userOperator.jvmOptions.javaSystemProperties	array	A map of additional system properties which will be passed using the `-D` option to the JVM.
spec.kafka.entityOperator.userOperator.livenessProbe	object	Pod liveness checking.
spec.kafka.entityOperator.userOperator.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.

Property	Type	Description
spec.kafka.entityOperator.userOperator.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.entityOperator.userOperator.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.entityOperator.userOperator.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.entityOperator.userOperator.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.entityOperator.userOperator.logging	object	Logging configuration.
spec.kafka.entityOperator.userOperator.logging.loggers	object	A Map from logger name to logger level.
spec.kafka.entityOperator.userOperator.logging.name	string	The name of the `ConfigMap` from which to get the logging configuration.
spec.kafka.entityOperator.userOperator.logging.type	string	Logging type, must be either 'inline' or 'external'.
spec.kafka.entityOperator.userOperator.readinessProbe	object	Pod readiness checking.
spec.kafka.entityOperator.userOperator.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.entityOperator.userOperator.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.

Property	Type	Description
spec.kafka.entityOperator.userOperator.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.entityOperator.userOperator.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.entityOperator.userOperator.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.entityOperator.userOperator.reconciliationIntervalSeconds	integer	Interval between periodic reconciliations.
spec.kafka.entityOperator.userOperator.resources	object	CPU and memory resources to reserve.
spec.kafka.entityOperator.userOperator.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.entityOperator.userOperator.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.entityOperator.userOperator.secretPrefix	string	The prefix that will be added to the KafkaUser name to be used as the Secret name.
spec.kafka.entityOperator.userOperator.watchedNamespace	string	The namespace the User Operator should watch.
spec.kafka.entityOperator.userOperator.zookeeperSessionTimeoutSeconds	integer	Timeout for the ZooKeeper session.
spec.kafka.jmxTrans	object	Configuration for JmxTrans. When the property is present a JmxTrans deployment is created for gathering JMX metrics from each Kafka broker. For more information see https://github.com/jmxtrans/jmxtrans[JmxTrans GitHub].

Property	Type	Description
spec.kafka.jmxTrans.image	string	The image to use for the JmxTrans.
spec.kafka.jmxTrans.kafkaQueries	array	Queries to send to the Kafka brokers to define what data should be read from each broker. For more information on these properties see, xref:type-JmxTransQueryTemplate-reference[`JmxTransQueryTemplate` schema reference].
spec.kafka.jmxTrans.logLevel	string	Sets the logging level of the JmxTrans deployment.For more information see, https://github.com/jmxtrans/jmxtrans-agent/wiki/Troubleshooting[JmxTrans Logging Level].
spec.kafka.jmxTrans.outputDefinitions	array	Defines the output hosts that will be referenced later on. For more information on these properties see, xref:type-JmxTransOutputDefinitionTemplate-reference[`JmxTransOutputDefinitionTemplate` schema reference].
spec.kafka.jmxTrans.resources	object	CPU and memory resources to reserve.
spec.kafka.jmxTrans.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.jmxTrans.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.jmxTrans.template	object	Template for JmxTrans resources.
spec.kafka.jmxTrans.template.contain er	object	Template for JmxTrans container.
spec.kafka.jmxTrans.template.contain er.env	array	Environment variables which should be applied to the container.
spec.kafka.jmxTrans.template.contain er.securityContext	object	Security context for the container.
spec.kafka.jmxTrans.template.contain er.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.jmxTrans.template.contain er.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".

Property	Type	Description
spec.kafka.jmxTrans.template.container.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.jmxTrans.template.container.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.jmxTrans.template.container.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.jmxTrans.template.container.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.jmxTrans.template.container.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.jmxTrans.template.container.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.jmxTrans.template.container.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.jmxTrans.template.container.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.jmxTrans.template.container.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".

Property	Type	Description
spec.kafka.jmxTrans.template.container.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.jmxTrans.template.container.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.jmxTrans.template.container.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.jmxTrans.template.container.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.jmxTrans.template.container.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.jmxTrans.template.container.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.jmxTrans.template.container.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.jmxTrans.template.container.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.jmxTrans.template.container.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".

Property	Type	Description
spec.kafka.jmxTrans.template.container.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.jmxTrans.template.container.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.jmxTrans.template.deployment	object	Template for JmxTrans **`Deployment`**.
spec.kafka.jmxTrans.template.deployment.metadata	object	Metadata applied to the resource.
spec.kafka.jmxTrans.template.deployment.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.jmxTrans.template.deployment.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.jmxTrans.template.pod	object	Template for JmxTrans **`Pods`**.
spec.kafka.jmxTrans.template.pod.affinity	object	The pod's affinity rules.
spec.kafka.jmxTrans.template.pod.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.jmxTrans.template.pod.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.jmxTrans.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.jmxTrans.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.jmxTrans.template.pod.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.jmxTrans.template.pod.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.jmxTrans.template.pod.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.jmxTrans.template.pod.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.jmxTrans.template.pod.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.jmxTrans.template.pod.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.jmxTrans.template.pod.hostAliases	array	The pod's HostAliases. HostAliases is an optional list of hosts and IPs that will be injected into the pod's hosts file if specified.
spec.kafka.jmxTrans.template.pod.imagePullSecrets	array	List of references to secrets in the same namespace to use for pulling any of the images used by this Pod. When the **`STRIMZI_IMAGE_PULL_SECRETS`** environment variable in Cluster Operator and the **`imagePullSecrets`** option are specified, only the **`imagePullSecrets`** variable is used and the **`STRIMZI_IMAGE_PULL_SECRETS`** variable is ignored.
spec.kafka.jmxTrans.template.pod.metadata	object	Metadata applied to the resource.
spec.kafka.jmxTrans.template.pod.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.jmxTrans.template.pod.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.jmxTrans.template.pod.priorityClassName	string	The name of the priority class used to assign priority to the pods. For more information about priority classes, see {K8sPriorityClass}.
spec.kafka.jmxTrans.template.pod.schedulerName	string	The name of the scheduler used to dispatch this **`Pod`**. If not specified, the default scheduler will be used.
spec.kafka.jmxTrans.template.pod.securityContext	object	Configures pod-level security attributes and common container settings.
spec.kafka.jmxTrans.template.pod.securityContext.fsGroup	integer	FsGroup corresponds to the JSON schema field "fsGroup".
spec.kafka.jmxTrans.template.pod.securityContext.fsGroupChangePolicy	string	FsGroupChangePolicy corresponds to the JSON schema field "fsGroupChangePolicy".

Property	Type	Description
spec.kafka.jmxTrans.template.pod.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.jmxTrans.template.pod.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.jmxTrans.template.pod.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.jmxTrans.template.pod.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.jmxTrans.template.pod.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.jmxTrans.template.pod.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.jmxTrans.template.pod.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.jmxTrans.template.pod.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.jmxTrans.template.pod.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.jmxTrans.template.pod.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".

Property	Type	Description
spec.kafka.jmxTrans.template.pod.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.jmxTrans.template.pod.securityContext.supplementalGroups	array	SupplementalGroups corresponds to the JSON schema field "supplementalGroups".
spec.kafka.jmxTrans.template.pod.securityContext.sysctls	array	Sysctls corresponds to the JSON schema field "sysctls".
spec.kafka.jmxTrans.template.pod.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.jmxTrans.template.pod.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.jmxTrans.template.pod.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.jmxTrans.template.pod.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.jmxTrans.template.pod.terminationGracePeriodSeconds	integer	The grace period is the duration in seconds after the processes running in the pod are sent a termination signal, and the time when the processes are forcibly halted with a kill signal. Set this value to longer than the expected cleanup time for your process. Value must be a non-negative integer. A zero value indicates delete immediately. You might need to increase the grace period for very large Kafka clusters, so that the Kafka brokers have enough time to transfer their work to another broker before they are terminated. Defaults to 30 seconds.
spec.kafka.jmxTrans.template.pod.tolerations	array	The pod's tolerations.

Property	Type	Description
spec.kafka.jmxTrans.template.pod.topologySpreadConstraints	array	The pod's topology spread constraints.
spec.kafka.kafka	object	Configuration of the Kafka cluster.
spec.kafka.kafka.affinity	object	The pod's affinity rules.
spec.kafka.kafka.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.kafka.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.kafka.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.kafka.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".

Property	Type	Description
spec.kafka.kafka.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.authorization	object	Authorization configuration for Kafka brokers.
spec.kafka.kafka.authorization.allowOnError	boolean	Defines whether a Kafka client should be allowed or denied by default when the authorizer fails to query the Open Policy Agent, for example, when it is temporarily unavailable). Defaults to **`false`** - all actions will be denied.
spec.kafka.kafka.authorization.clientId	string	OAuth Client ID which the Kafka client can use to authenticate against the OAuth server and use the token endpoint URI.
spec.kafka.kafka.authorization.delegateToKafkaAcls	boolean	Whether authorization decision should be delegated to the 'Simple' authorizer if DENIED by Keycloak Authorization Services policies. Default value is **`false`**.
spec.kafka.kafka.authorization.disableTlsHostnameVerification	boolean	Enable or disable TLS hostname verification. Default value is **`false`**.
spec.kafka.kafka.authorization.expireAfterMs	integer	The expiration of the records kept in the local cache to avoid querying the Open Policy Agent for every request. Defines how often the cached authorization decisions are reloaded from the Open Policy Agent server. In milliseconds. Defaults to **`3600000`**.
spec.kafka.kafka.authorization.grantsRefreshPeriodSeconds	integer	The time between two consecutive grants refresh runs in seconds. The default value is 60.
spec.kafka.kafka.authorization.grantsRefreshPoolSize	integer	The number of threads to use to refresh grants for active sessions. The more threads, the more parallelism, so the sooner the job completes. However, using more threads places a heavier load on the authorization server. The default value is 5.
spec.kafka.kafka.authorization.initialCacheCapacity	integer	Initial capacity of the local cache used by the authorizer to avoid querying the Open Policy Agent for every request Defaults to **`5000`**.
spec.kafka.kafka.authorization.maximumCacheSize	integer	Maximum capacity of the local cache used by the authorizer to avoid querying the Open Policy Agent for every request. Defaults to **`50000`**.

Property	Type	Description
spec.kafka.kafka.authorization.superUsers	array	List of super users. Should contain list of user principals which should get unlimited access rights.
spec.kafka.kafka.authorization.tlsTrustedCertificates	array	Trusted certificates for TLS connection to the OAuth server.
spec.kafka.kafka.authorization.tokenEndpointUri	string	Authorization server token endpoint URI.
spec.kafka.kafka.authorization.type	string	Authorization type. Currently, the supported types are **`simple`**, **`keycloak`**, and **`opa`**. **`simple`** authorization type uses Kafka's **`kafka.security.authorizer.AclAuthorizer`** class for authorization. **`keycloak`** authorization type uses Keycloak Authorization Services for authorization. **`opa`** authorization type uses Open Policy Agent based authorization.
spec.kafka.kafka.authorization.url	string	The URL used to connect to the Open Policy Agent server. The URL has to include the policy which will be queried by the authorizer. This option is required.
spec.kafka.kafka.brokerRackInitImage	string	The image of the init container used for initializing the **`broker.rack`**.
spec.kafka.kafka.config	object	Kafka broker config properties with the following prefixes cannot be set: listeners, advertised., broker., listener., host.name, port, inter.broker.listener.name, sasl., ssl., security., password., principal.builder.class, log.dir, zookeeper.connect, zookeeper.set.acl, zookeeper.ssl, zookeeper.clientCnxnSocket, authorizer., super.user, cruise.control.metrics.topic, cruise.control.metrics.reporter.bootstrap.servers (with the exception of: zookeeper.connection.timeout.ms, ssl.cipher.suites, ssl.protocol, ssl.enabled.protocols,cruise.control.metrics.topic.num.partitions, cruise.control.metrics.topic.replication.factor, cruise.control.metrics.topic.retention.ms,cruise.control.metrics.topic.auto.create.retries, cruise.control.metrics.topic.auto.create.timeout.ms,cruise.control.metrics.topic.min.insync.replicas).
spec.kafka.kafka.image	string	The docker image for the pods. The default value depends on the configured **`Kafka.spec.kafka.version`**.
spec.kafka.kafka.jmxOptions	object	JMX Options for Kafka brokers.
spec.kafka.kafka.jmxOptions.authentication	object	Authentication configuration for connecting to the Kafka JMX port.
spec.kafka.kafka.jmxOptions.authentication.type	string	Authentication type. Currently the only supported types are **`password`**.**`password`** type creates a username and protected port with no TLS.
spec.kafka.kafka.jvmOptions	object	JVM Options for pods.

Property	Type	Description
spec.kafka.kafka.jvmOptions.-XX	object	A map of -XX options to the JVM.
spec.kafka.kafka.jvmOptions.-Xms	string	-Xms option to to the JVM.
spec.kafka.kafka.jvmOptions.-Xmx	string	-Xmx option to to the JVM.
spec.kafka.kafka.jvmOptions.gcLoggingEnabled	boolean	Specifies whether the Garbage Collection logging is enabled. The default is false.
spec.kafka.kafka.jvmOptions.javaSystemProperties	array	A map of additional system properties which will be passed using the `-D` option to the JVM.
spec.kafka.kafka.listeners	array	Listeners corresponds to the JSON schema field "listeners".
spec.kafka.kafka.livenessProbe	object	Pod liveness checking.
spec.kafka.kafka.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.kafka.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.kafka.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.kafka.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.kafka.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.kafka.logging	object	Logging configuration for Kafka.
spec.kafka.kafka.logging.loggers	object	A Map from logger name to logger level.
spec.kafka.kafka.logging.name	string	The name of the `ConfigMap` from which to get the logging configuration.
spec.kafka.kafka.logging.type	string	Logging type, must be either 'inline' or 'external'.
spec.kafka.kafka.metrics	object	The Prometheus JMX Exporter configuration. See https://github.com/prometheus/jmx_exporter for details of the structure of this configuration.

Property	Type	Description
spec.kafka.kafka.metricsConfig	object	Metrics configuration.
spec.kafka.kafka.metricsConfig.type	string	Metrics type. Only 'jmxPrometheusExporter' supported currently.
spec.kafka.kafka.metricsConfig.valueFrom	object	ConfigMap where the Prometheus JMX Exporter configuration is stored. For details of the structure of this configuration, see the {JMXExporter}.
spec.kafka.kafka.metricsConfig.valueFrom.configMapKeyRef	object	Reference to the key in the ConfigMap containing the metrics configuration.
spec.kafka.kafka.metricsConfig.valueFrom.configMapKeyRef.key	string	Key corresponds to the JSON schema field "key".
spec.kafka.kafka.metricsConfig.valueFrom.configMapKeyRef.name	string	Name corresponds to the JSON schema field "name".
spec.kafka.kafka.metricsConfig.valueFrom.configMapKeyRef.optional	boolean	Optional corresponds to the JSON schema field "optional".
spec.kafka.kafka.rack	object	Configuration of the `broker.rack` broker config.
spec.kafka.kafka.rack.topologyKey	string	A key that matches labels assigned to the Kubernetes cluster nodes. The value of the label is used to set the broker's `broker.rack` config and `client.rack` in Kafka Connect.
spec.kafka.kafka.readinessProbe	object	Pod readiness checking.
spec.kafka.kafka.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.kafka.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.kafka.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.

Property	Type	Description
spec.kafka.kafka.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.kafka.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.kafka.replicas	integer	The number of pods in the cluster.
spec.kafka.kafka.resources	object	CPU and memory resources to reserve.
spec.kafka.kafka.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.kafka.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.kafka.storage	object	Storage configuration (disk). Cannot be updated.
spec.kafka.kafka.storage.class	string	The storage class to use for dynamic volume allocation.
spec.kafka.kafka.storage.deleteClaim	boolean	Specifies if the persistent volume claim has to be deleted when the cluster is un-deployed.
spec.kafka.kafka.storage.id	integer	Storage identification number. It is mandatory only for storage volumes defined in a storage of type 'jbod'.
spec.kafka.kafka.storage.overrides	array	Overrides for individual brokers. The **`overrides`** field allows to specify a different configuration for different brokers.
spec.kafka.kafka.storage.selector	object	Specifies a specific persistent volume to use. It contains key:value pairs representing labels for selecting such a volume.
spec.kafka.kafka.storage.size	string	When type=persistent-claim, defines the size of the persistent volume claim (i.e 1Gi). Mandatory when type=persistent-claim.
spec.kafka.kafka.storage.sizeLimit	string	When type=ephemeral, defines the total amount of local storage required for this EmptyDir volume (for example 1Gi).
spec.kafka.kafka.storage.type	string	Storage type, must be either 'ephemeral', 'persistent-claim', or 'jbod'.
spec.kafka.kafka.storage.volumes	array	List of volumes as Storage objects representing the JBOD disks array.
spec.kafka.kafka.template	object	Template for Kafka cluster resources. The template allows users to specify how are the **`StatefulSet`**, **`Pods`** and **`Services`** generated.
spec.kafka.kafka.template.bootstrapService	object	Template for Kafka bootstrap **`Service`**.

Property	Type	Description
spec.kafka.kafka.template.bootstrapService.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.bootstrapService.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.bootstrapService.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.brokersService	object	Template for Kafka broker **`Service`**.
spec.kafka.kafka.template.brokersService.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.brokersService.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.brokersService.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.clusterCaCert	object	Template for Secret with Kafka Cluster certificate public key.
spec.kafka.kafka.template.clusterCaCert.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.clusterCaCert.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.clusterCaCert.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.externalBootstrapIngress	object	Template for Kafka external bootstrap **`Ingress`**.

Property	Type	Description
spec.kafka.kafka.template.externalBootstrapIngress.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.externalBootstrapIngress.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.externalBootstrapIngress.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.externalBootstrapRoute	object	Template for Kafka external bootstrap **`Route`**.
spec.kafka.kafka.template.externalBootstrapRoute.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.externalBootstrapRoute.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.externalBootstrapRoute.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.externalBootstrapService	object	Template for Kafka external bootstrap **`Service`**.
spec.kafka.kafka.template.externalBootstrapService.externalTrafficPolicy	string	Specifies whether the service routes external traffic to node-local or cluster-wide endpoints. **`Cluster`** may cause a second hop to another node and obscures the client source IP. **`Local`** avoids a second hop for LoadBalancer and Nodeport type services and preserves the client source IP (when supported by the infrastructure). If unspecified, Kubernetes will use **`Cluster`** as the default.
spec.kafka.kafka.template.externalBootstrapService.loadBalancerSourceRanges	array	A list of CIDR ranges (for example **`10.0.0.0/8`** or **`130.211.204.1/32`**) from which clients can connect to load balancer type listeners. If supported by the platform, traffic through the loadbalancer is restricted to the specified CIDR ranges. This field is applicable only for loadbalancer type services and is ignored if the cloud provider does not support the feature. For more information, see https://v1-17.docs.kubernetes.io/docs/tasks/access-application-cluster/configure-cloud-provider-firewall/.

Property	Type	Description
spec.kafka.kafka.template.externalBootstrapService.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.externalBootstrapService.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.kafka.template.externalBootstrapService.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.kafka.template.initContainer	object	Template for the Kafka init container.
spec.kafka.kafka.template.initContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.kafka.template.initContainer.securityContext	object	Security context for the container.
spec.kafka.kafka.template.initContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.kafka.template.initContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.kafka.template.initContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.kafka.template.initContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.kafka.template.initContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".

Property	Type	Description
spec.kafka.kafka.template.initContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.kafka.template.initContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.kafka.template.initContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.kafka.template.initContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.kafka.template.initContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.kafka.template.initContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.kafka.template.initContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.kafka.template.initContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.kafka.template.initContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".

Property	Type	Description
spec.kafka.kafka.template.initContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.kafka.template.initContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.kafka.template.initContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.kafka.template.initContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafka.template.initContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.kafka.template.initContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.kafka.template.initContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.kafka.template.initContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.kafka.template.kafkaContainer	object	Template for the Kafka broker container.

Property	Type	Description
spec.kafka.kafka.template.kafkaContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.kafka.template.kafkaContainer.securityContext	object	Security context for the container.
spec.kafka.kafka.template.kafkaContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.kafka.template.kafkaContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.kafka.template.kafkaContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.kafka.template.kafkaContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.kafka.template.kafkaContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.kafka.template.kafkaContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.kafka.template.kafkaContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.kafka.template.kafkaContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".

Property	Type	Description
spec.kafka.kafka.template.kafkaContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.kafka.template.kafkaContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.kafka.template.kafkaContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.kafka.template.kafkaContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.kafka.template.kafkaContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.kafka.template.kafkaContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafka.template.kafkaContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.kafka.template.kafkaContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.kafka.template.kafkaContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".

Property	Type	Description
spec.kafka.kafka.template.kafkaContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafka.template.kafkaContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.kafka.template.kafkaContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.kafka.template.kafkaContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.kafka.template.kafkaContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.kafka.template.perPodIngress	object	Template for Kafka per-pod **`Ingress`** used for access from outside of Kubernetes.
spec.kafka.kafka.template.perPodIngress.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.perPodIngress.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.perPodIngress.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.perPodRoute	object	Template for Kafka per-pod **`Routes`** used for access from outside of OpenShift.

Property	Type	Description
spec.kafka.kafka.template.perPodRoute.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.perPodRoute.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.perPodRoute.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.perPodService	object	Template for Kafka per-pod **`Services`** used for access from outside of Kubernetes.
spec.kafka.kafka.template.perPodService.externalTrafficPolicy	string	Specifies whether the service routes external traffic to node-local or cluster-wide endpoints. **`Cluster`** may cause a second hop to another node and obscures the client source IP. **`Local`** avoids a second hop for LoadBalancer and Nodeport type services and preserves the client source IP (when supported by the infrastructure). If unspecified, Kubernetes will use **`Cluster`** as the default.
spec.kafka.kafka.template.perPodService.loadBalancerSourceRanges	array	A list of CIDR ranges (for example **`10.0.0.0/8`** or **`130.211.204.1/32`**) from which clients can connect to load balancer type listeners. If supported by the platform, traffic through the loadbalancer is restricted to the specified CIDR ranges. This field is applicable only for loadbalancer type services and is ignored if the cloud provider does not support the feature. For more information, see https://v1-17.docs.kubernetes.io/docs/tasks/access-application-cluster/configure-cloud-provider-firewall/.
spec.kafka.kafka.template.perPodService.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.perPodService.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.perPodService.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.persistentVolumeClaim	object	Template for all Kafka **`PersistentVolumeClaims`**.
spec.kafka.kafka.template.persistentVolumeClaim.metadata	object	Metadata applied to the resource.

Property	Type	Description
spec.kafka.kafka.template.persistentVolumeClaim.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.kafka.template.persistentVolumeClaim.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.kafka.template.pod	object	Template for Kafka `Pods`.
spec.kafka.kafka.template.pod.affinity	object	The pod's affinity rules.
spec.kafka.kafka.template.pod.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.kafka.template.pod.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.kafka.template.pod.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".

Property	Type	Description
spec.kafka.kafka.template.pod.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.template.pod.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.template.pod.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.kafka.template.pod.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.template.pod.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafka.template.pod.hostAliases	array	The pod's HostAliases. HostAliases is an optional list of hosts and IPs that will be injected into the pod's hosts file if specified.
spec.kafka.kafka.template.pod.imagePullSecrets	array	List of references to secrets in the same namespace to use for pulling any of the images used by this Pod. When the **`STRIMZI_IMAGE_PULL_SECRETS`** environment variable in Cluster Operator and the **`imagePullSecrets`** option are specified, only the **`imagePullSecrets`** variable is used and the **`STRIMZI_IMAGE_PULL_SECRETS`** variable is ignored.
spec.kafka.kafka.template.pod.metadata	object	Metadata applied to the resource.
spec.kafka.kafka.template.pod.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.

Property	Type	Description
spec.kafka.kafka.template.pod.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.kafka.template.pod.priorityClassName	string	The name of the priority class used to assign priority to the pods. For more information about priority classes, see {K8sPriorityClass}.
spec.kafka.kafka.template.pod.schedulerName	string	The name of the scheduler used to dispatch this `Pod`. If not specified, the default scheduler will be used.
spec.kafka.kafka.template.pod.securityContext	object	Configures pod-level security attributes and common container settings.
spec.kafka.kafka.template.pod.securityContext.fsGroup	integer	FsGroup corresponds to the JSON schema field "fsGroup".
spec.kafka.kafka.template.pod.securityContext.fsGroupChangePolicy	string	FsGroupChangePolicy corresponds to the JSON schema field "fsGroupChangePolicy".
spec.kafka.kafka.template.pod.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.kafka.template.pod.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.kafka.template.pod.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.kafka.template.pod.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.kafka.template.pod.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.kafka.template.pod.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".

Property	Type	Description
spec.kafka.kafka.template.pod.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafka.template.pod.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.kafka.template.pod.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.kafka.template.pod.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.kafka.template.pod.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafka.template.pod.securityContext.supplementalGroups	array	SupplementalGroups corresponds to the JSON schema field "supplementalGroups".
spec.kafka.kafka.template.pod.securityContext.sysctls	array	Sysctls corresponds to the JSON schema field "sysctls".
spec.kafka.kafka.template.pod.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.kafka.template.pod.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.kafka.template.pod.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".

Property	Type	Description
spec.kafka.kafka.template.pod.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.kafka.template.pod.terminationGracePeriodSeconds	integer	The grace period is the duration in seconds after the processes running in the pod are sent a termination signal, and the time when the processes are forcibly halted with a kill signal. Set this value to longer than the expected cleanup time for your process. Value must be a non-negative integer. A zero value indicates delete immediately. You might need to increase the grace period for very large Kafka clusters, so that the Kafka brokers have enough time to transfer their work to another broker before they are terminated. Defaults to 30 seconds.
spec.kafka.kafka.template.pod.tolerations	array	The pod's tolerations.
spec.kafka.kafka.template.pod.topologySpreadConstraints	array	The pod's topology spread constraints.
spec.kafka.kafka.template.podDisruptionBudget	object	Template for Kafka `PodDisruptionBudget`.
spec.kafka.kafka.template.podDisruptionBudget.maxUnavailable	integer	Maximum number of unavailable pods to allow automatic Pod eviction. A Pod eviction is allowed when the `maxUnavailable` number of pods or fewer are unavailable after the eviction. Setting this value to 0 prevents all voluntary evictions, so the pods must be evicted manually. Defaults to 1.
spec.kafka.kafka.template.podDisruptionBudget.metadata	object	Metadata to apply to the `PodDistruptionBugetTemplate` resource.
spec.kafka.kafka.template.podDisruptionBudget.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.kafka.template.podDisruptionBudget.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as `StatefulSets`, `Deployments`, `Pods`, and `Services`.
spec.kafka.kafka.template.statefulset	object	Template for Kafka `StatefulSet`.
spec.kafka.kafka.template.statefulset.metadata	object	Metadata applied to the resource.

Property	Type	Description
spec.kafka.kafka.template.statefulset.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.statefulset.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafka.template.statefulset.podManagementPolicy	string	PodManagementPolicy which will be used for this StatefulSet. Valid values are **`Parallel`** and **`OrderedReady`**. Defaults to **`Parallel`**.
spec.kafka.kafka.template.tlsSidecarContainer	object	Template for the Kafka broker TLS sidecar container.
spec.kafka.kafka.template.tlsSidecarContainer.env	array	Environment variables which should be applied to the container.
spec.kafka.kafka.template.tlsSidecarContainer.securityContext	object	Security context for the container.
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".

Property	Type	Description
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".

Property	Type	Description
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.kafka.template.tlsSidecarContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".

Property	Type	Description
spec.kafka.kafka.tlsSidecar	object	TLS sidecar configuration.
spec.kafka.kafka.tlsSidecar.image	string	The docker image for the container.
spec.kafka.kafka.tlsSidecar.livenessProbe	object	Pod liveness checking.
spec.kafka.kafka.tlsSidecar.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.kafka.tlsSidecar.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.kafka.tlsSidecar.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.kafka.tlsSidecar.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.kafka.tlsSidecar.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.kafka.tlsSidecar.logLevel	string	The log level for the TLS sidecar. Default value is `notice`.
spec.kafka.kafka.tlsSidecar.readinessProbe	object	Pod readiness checking.
spec.kafka.kafka.tlsSidecar.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.kafka.tlsSidecar.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.kafka.tlsSidecar.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.

Property	Type	Description
spec.kafka.kafka.tlsSidecar.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.kafka.tlsSidecar.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.kafka.tlsSidecar.resources	object	CPU and memory resources to reserve.
spec.kafka.kafka.tlsSidecar.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.kafka.tlsSidecar.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.kafka.tolerations	array	The pod's tolerations.
spec.kafka.kafka.version	string	The kafka broker version. Defaults to {DefaultKafkaVersion}. Consult the user documentation to understand the process required to upgrade or downgrade the version.
spec.kafka.kafkaExporter	object	Configuration of the Kafka Exporter. Kafka Exporter can provide additional metrics, for example lag of consumer group at topic/partition.
spec.kafka.kafkaExporter.enableSaramaLogging	boolean	Enable Sarama logging, a Go client library used by the Kafka Exporter.
spec.kafka.kafkaExporter.groupRegex	string	Regular expression to specify which consumer groups to collect. Default value is `.*`.
spec.kafka.kafkaExporter.image	string	The docker image for the pods.
spec.kafka.kafkaExporter.livenessProbe	object	Pod liveness check.
spec.kafka.kafkaExporter.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.kafkaExporter.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.kafkaExporter.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.

Property	Type	Description
spec.kafka.kafkaExporter.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.kafkaExporter.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.kafkaExporter.logging	string	Only log messages with the given severity or above. Valid levels: [`debug`, `info`, `warn`, `error`, `fatal`]. Default log level is `info`.
spec.kafka.kafkaExporter.readinessProbe	object	Pod readiness check.
spec.kafka.kafkaExporter.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.kafkaExporter.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.kafkaExporter.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.kafkaExporter.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.kafkaExporter.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.kafkaExporter.resources	object	CPU and memory resources to reserve.
spec.kafka.kafkaExporter.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.kafkaExporter.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.kafkaExporter.template	object	Customization of deployment templates and pods.
spec.kafka.kafkaExporter.template.container	object	Template for the Kafka Exporter container.

Property	Type	Description
spec.kafka.kafkaExporter.template.container.env	array	Environment variables which should be applied to the container.
spec.kafka.kafkaExporter.template.container.securityContext	object	Security context for the container.
spec.kafka.kafkaExporter.template.container.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.kafkaExporter.template.container.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.kafkaExporter.template.container.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.kafkaExporter.template.container.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.kafkaExporter.template.container.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.kafkaExporter.template.container.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.kafkaExporter.template.container.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.kafkaExporter.template.container.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".

Property	Type	Description
spec.kafka.kafkaExporter.template.container.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.kafkaExporter.template.container.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.kafkaExporter.template.container.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.kafkaExporter.template.container.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.kafkaExporter.template.container.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.kafkaExporter.template.container.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafkaExporter.template.container.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.kafkaExporter.template.container.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.kafkaExporter.template.container.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".

Property	Type	Description
spec.kafka.kafkaExporter.template.container.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafkaExporter.template.container.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.kafkaExporter.template.container.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.kafkaExporter.template.container.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.kafkaExporter.template.container.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.kafkaExporter.template.deployment	object	Template for Kafka Exporter **`Deployment`**.
spec.kafka.kafkaExporter.template.deployment.metadata	object	Metadata applied to the resource.
spec.kafka.kafkaExporter.template.deployment.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafkaExporter.template.deployment.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafkaExporter.template.pod	object	Template for Kafka Exporter **`Pods`**.

Property	Type	Description
spec.kafka.kafkaExporter.template.pod.affinity	object	The pod's affinity rules.
spec.kafka.kafkaExporter.template.pod.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.kafkaExporter.template.pod.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafkaExporter.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafkaExporter.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.kafkaExporter.template.pod.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.kafkaExporter.template.pod.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafkaExporter.template.pod.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.kafkaExporter.template.pod.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.kafkaExporter.template.pod.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafkaExporter.template.pod.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.kafkaExporter.template.pod.hostAliases	array	The pod's HostAliases. HostAliases is an optional list of hosts and IPs that will be injected into the pod's hosts file if specified.
spec.kafka.kafkaExporter.template.pod.imagePullSecrets	array	List of references to secrets in the same namespace to use for pulling any of the images used by this Pod. When the **`STRIMZI_IMAGE_PULL_SECRETS`** environment variable in Cluster Operator and the **`imagePullSecrets`** option are specified, only the **`imagePullSecrets`** variable is used and the **`STRIMZI_IMAGE_PULL_SECRETS`** variable is ignored.
spec.kafka.kafkaExporter.template.pod.metadata	object	Metadata applied to the resource.
spec.kafka.kafkaExporter.template.pod.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafkaExporter.template.pod.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafkaExporter.template.pod.priorityClassName	string	The name of the priority class used to assign priority to the pods. For more information about priority classes, see {K8sPriorityClass}.
spec.kafka.kafkaExporter.template.pod.schedulerName	string	The name of the scheduler used to dispatch this **`Pod`**. If not specified, the default scheduler will be used.

Property	Type	Description
spec.kafka.kafkaExporter.template.pod.securityContext	object	Configures pod-level security attributes and common container settings.
spec.kafka.kafkaExporter.template.pod.securityContext.fsGroup	integer	FsGroup corresponds to the JSON schema field "fsGroup".
spec.kafka.kafkaExporter.template.pod.securityContext.fsGroupChangePolicy	string	FsGroupChangePolicy corresponds to the JSON schema field "fsGroupChangePolicy".
spec.kafka.kafkaExporter.template.pod.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.kafkaExporter.template.pod.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.kafkaExporter.template.pod.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.kafkaExporter.template.pod.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.kafkaExporter.template.pod.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.kafkaExporter.template.pod.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.kafkaExporter.template.pod.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".

Property	Type	Description
spec.kafka.kafkaExporter.template.pod.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.kafkaExporter.template.pod.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.kafkaExporter.template.pod.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.kafkaExporter.template.pod.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.kafkaExporter.template.pod.securityContext.supplementalGroups	array	SupplementalGroups corresponds to the JSON schema field "supplementalGroups".
spec.kafka.kafkaExporter.template.pod.securityContext.sysctls	array	Sysctls corresponds to the JSON schema field "sysctls".
spec.kafka.kafkaExporter.template.pod.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.kafkaExporter.template.pod.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.kafkaExporter.template.pod.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".

Property	Type	Description
spec.kafka.kafkaExporter.template.pod.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.kafkaExporter.template.pod.terminationGracePeriodSeconds	integer	The grace period is the duration in seconds after the processes running in the pod are sent a termination signal, and the time when the processes are forcibly halted with a kill signal. Set this value to longer than the expected cleanup time for your process. Value must be a non-negative integer. A zero value indicates delete immediately. You might need to increase the grace period for very large Kafka clusters, so that the Kafka brokers have enough time to transfer their work to another broker before they are terminated. Defaults to 30 seconds.
spec.kafka.kafkaExporter.template.pod.tolerations	array	The pod's tolerations.
spec.kafka.kafkaExporter.template.pod.topologySpreadConstraints	array	The pod's topology spread constraints.
spec.kafka.kafkaExporter.template.service	object	Template for Kafka Exporter **`Service`**.
spec.kafka.kafkaExporter.template.service.metadata	object	Metadata applied to the resource.
spec.kafka.kafkaExporter.template.service.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafkaExporter.template.service.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.kafkaExporter.topicRegex	string	Regular expression to specify which topics to collect. Default value is **`.*`**.
spec.kafka.maintenanceTimeWindows	array	A list of time windows for maintenance tasks (that is, certificates renewal). Each time window is defined by a cron expression.
spec.kafka.topicOperator	object	Configuration of the Topic Operator.
spec.kafka.topicOperator.affinity	object	Pod affinity rules.

Property	Type	Description
spec.kafka.topicOperator.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.topicOperator.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.topicOperator.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.topicOperator.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.topicOperator.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.topicOperator.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.topicOperator.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.topicOperator.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.topicOperator.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.topicOperator.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.topicOperator.image	string	The image to use for the Topic Operator.
spec.kafka.topicOperator.jvmOptions	object	JVM Options for pods.
spec.kafka.topicOperator.jvmOptions.-XX	object	A map of -XX options to the JVM.
spec.kafka.topicOperator.jvmOptions.-Xms	string	-Xms option to to the JVM.
spec.kafka.topicOperator.jvmOptions.-Xmx	string	-Xmx option to to the JVM.
spec.kafka.topicOperator.jvmOptions.gcLoggingEnabled	boolean	Specifies whether the Garbage Collection logging is enabled. The default is false.
spec.kafka.topicOperator.jvmOptions.javaSystemProperties	array	A map of additional system properties which will be passed using the `-D` option to the JVM.
spec.kafka.topicOperator.livenessProbe	object	Pod liveness checking.
spec.kafka.topicOperator.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.topicOperator.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.topicOperator.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.topicOperator.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.

Property	Type	Description
spec.kafka.topicOperator.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.topicOperator.logging	object	Logging configuration.
spec.kafka.topicOperator.logging.loggers	object	A Map from logger name to logger level.
spec.kafka.topicOperator.logging.name	string	The name of the `ConfigMap` from which to get the logging configuration.
spec.kafka.topicOperator.logging.type	string	Logging type, must be either 'inline' or 'external'.
spec.kafka.topicOperator.readinessProbe	object	Pod readiness checking.
spec.kafka.topicOperator.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.topicOperator.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.topicOperator.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.topicOperator.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.topicOperator.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.topicOperator.reconciliationIntervalSeconds	integer	Interval between periodic reconciliations.
spec.kafka.topicOperator.resources	object	CPU and memory resources to reserve.
spec.kafka.topicOperator.resources.limits	object	Limits corresponds to the JSON schema field "limits".

Property	Type	Description
spec.kafka.topicOperator.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.topicOperator.tlsSidecar	object	TLS sidecar configuration.
spec.kafka.topicOperator.tlsSidecar.image	string	The docker image for the container.
spec.kafka.topicOperator.tlsSidecar.livenessProbe	object	Pod liveness checking.
spec.kafka.topicOperator.tlsSidecar.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.topicOperator.tlsSidecar.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.topicOperator.tlsSidecar.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.topicOperator.tlsSidecar.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.topicOperator.tlsSidecar.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.topicOperator.tlsSidecar.logLevel	string	The log level for the TLS sidecar. Default value is `notice`.
spec.kafka.topicOperator.tlsSidecar.readinessProbe	object	Pod readiness checking.
spec.kafka.topicOperator.tlsSidecar.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.

Property	Type	Description
spec.kafka.topicOperator.tlsSidecar.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.topicOperator.tlsSidecar.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.topicOperator.tlsSidecar.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.topicOperator.tlsSidecar.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.topicOperator.tlsSidecar.resources	object	CPU and memory resources to reserve.
spec.kafka.topicOperator.tlsSidecar.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.topicOperator.tlsSidecar.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.topicOperator.topicMetadataMaxAttempts	integer	The number of attempts at getting topic metadata.
spec.kafka.topicOperator.watchedNamespace	string	The namespace the Topic Operator should watch.
spec.kafka.topicOperator.zookeeperSessionTimeoutSeconds	integer	Timeout for the ZooKeeper session.
spec.kafka.zookeeper	object	Configuration of the ZooKeeper cluster.
spec.kafka.zookeeper.affinity	object	The pod's affinity rules.
spec.kafka.zookeeper.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".

Property	Type	Description
spec.kafka.zookeeper.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.zookeeper.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.zookeeper.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.zookeeper.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.zookeeper.config	object	The ZooKeeper broker config. Properties with the following prefixes cannot be set: server., dataDir, dataLogDir, clientPort, authProvider, quorum.auth, requireClientAuthScheme, snapshot.trust.empty, standaloneEnabled, reconfigEnabled, 4lw.commands.whitelist, secureClientPort, ssl., serverCnxnFactory, sslQuorum (with the exception of: ssl.protocol, ssl.quorum.protocol, ssl.enabledProtocols, ssl.quorum.enabledProtocols, ssl.ciphersuites, ssl.quorum.ciphersuites, ssl.hostnameVerification, ssl.quorum.hostnameVerification).
spec.kafka.zookeeper.image	string	The docker image for the pods.
spec.kafka.zookeeper.jvmOptions	object	JVM Options for pods.
spec.kafka.zookeeper.jvmOptions.-XX	object	A map of -XX options to the JVM.
spec.kafka.zookeeper.jvmOptions.-Xms	string	-Xms option to to the JVM.
spec.kafka.zookeeper.jvmOptions.-Xmx	string	-Xmx option to to the JVM.
spec.kafka.zookeeper.jvmOptions.gcLoggingEnabled	boolean	Specifies whether the Garbage Collection logging is enabled. The default is false.
spec.kafka.zookeeper.jvmOptions.javaSystemProperties	array	A map of additional system properties which will be passed using the `-D` option to the JVM.
spec.kafka.zookeeper.livenessProbe	object	Pod liveness checking.
spec.kafka.zookeeper.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.zookeeper.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.zookeeper.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.

Property	Type	Description
spec.kafka.zookeeper.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.zookeeper.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.zookeeper.logging	object	Logging configuration for ZooKeeper.
spec.kafka.zookeeper.logging.loggers	object	A Map from logger name to logger level.
spec.kafka.zookeeper.logging.name	string	The name of the `ConfigMap` from which to get the logging configuration.
spec.kafka.zookeeper.logging.type	string	Logging type, must be either 'inline' or 'external'.
spec.kafka.zookeeper.metrics	object	The Prometheus JMX Exporter configuration. See https://github.com/prometheus/jmx_exporter for details of the structure of this configuration.
spec.kafka.zookeeper.metricsConfig	object	Metrics configuration.
spec.kafka.zookeeper.metricsConfig.type	string	Metrics type. Only 'jmxPrometheusExporter' supported currently.
spec.kafka.zookeeper.metricsConfig.valueFrom	object	ConfigMap where the Prometheus JMX Exporter configuration is stored. For details of the structure of this configuration, see the {JMXExporter}.
spec.kafka.zookeeper.metricsConfig.valueFrom.configMapKeyRef	object	Reference to the key in the ConfigMap containing the metrics configuration.
spec.kafka.zookeeper.metricsConfig.valueFrom.configMapKeyRef.key	string	Key corresponds to the JSON schema field "key".
spec.kafka.zookeeper.metricsConfig.valueFrom.configMapKeyRef.name	string	Name corresponds to the JSON schema field "name".

Property	Type	Description
spec.kafka.zookeeper.metricsConfig.valueFrom.configMapKeyRef.optional	boolean	Optional corresponds to the JSON schema field "optional".
spec.kafka.zookeeper.readinessProbe	object	Pod readiness checking.
spec.kafka.zookeeper.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.zookeeper.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.zookeeper.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.zookeeper.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.zookeeper.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.zookeeper.replicas	integer	The number of pods in the cluster.
spec.kafka.zookeeper.resources	object	CPU and memory resources to reserve.
spec.kafka.zookeeper.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.zookeeper.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.zookeeper.storage	object	Storage configuration (disk). Cannot be updated.
spec.kafka.zookeeper.storage.class	string	The storage class to use for dynamic volume allocation.
spec.kafka.zookeeper.storage.deleteClaim	boolean	Specifies if the persistent volume claim has to be deleted when the cluster is un-deployed.
spec.kafka.zookeeper.storage.id	integer	Storage identification number. It is mandatory only for storage volumes defined in a storage of type 'jbod'.

Property	Type	Description
spec.kafka.zookeeper.storage.overrides	array	Overrides for individual brokers. The **`overrides`** field allows to specify a different configuration for different brokers.
spec.kafka.zookeeper.storage.selector	object	Specifies a specific persistent volume to use. It contains key:value pairs representing labels for selecting such a volume.
spec.kafka.zookeeper.storage.size	string	When type=persistent-claim, defines the size of the persistent volume claim (i.e 1Gi). Mandatory when type=persistent-claim.
spec.kafka.zookeeper.storage.sizeLimit	string	When type=ephemeral, defines the total amount of local storage required for this EmptyDir volume (for example 1Gi).
spec.kafka.zookeeper.storage.type	string	Storage type, must be either 'ephemeral' or 'persistent-claim'.
spec.kafka.zookeeper.template	object	Template for ZooKeeper cluster resources. The template allows users to specify how are the **`StatefulSet`**, **`Pods`** and **`Services`** generated.
spec.kafka.zookeeper.template.clientService	object	Template for ZooKeeper client **`Service`**.
spec.kafka.zookeeper.template.clientService.metadata	object	Metadata applied to the resource.
spec.kafka.zookeeper.template.clientService.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.zookeeper.template.clientService.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.zookeeper.template.nodesService	object	Template for ZooKeeper nodes **`Service`**.
spec.kafka.zookeeper.template.nodesService.metadata	object	Metadata applied to the resource.
spec.kafka.zookeeper.template.nodesService.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.zookeeper.template.nodesService.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.

Property	Type	Description
spec.kafka.zookeeper.template.persistentVolumeClaim	object	Template for all ZooKeeper **`PersistentVolumeClaims`**.
spec.kafka.zookeeper.template.persistentVolumeClaim.metadata	object	Metadata applied to the resource.
spec.kafka.zookeeper.template.persistentVolumeClaim.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.zookeeper.template.persistentVolumeClaim.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.zookeeper.template.pod	object	Template for ZooKeeper **`Pods`**.
spec.kafka.zookeeper.template.pod.affinity	object	The pod's affinity rules.
spec.kafka.zookeeper.template.pod.affinity.nodeAffinity	object	NodeAffinity corresponds to the JSON schema field "nodeAffinity".
spec.kafka.zookeeper.template.pod.affinity.nodeAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution	object	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".

Property	Type	Description
spec.kafka.zookeeper.template.pod.affinity.nodeAffinity.requiredDuringSchedulingIgnoredDuringExecution.nodeSelectorTerms	array	NodeSelectorTerms corresponds to the JSON schema field "nodeSelectorTerms".
spec.kafka.zookeeper.template.pod.affinity.podAffinity	object	PodAffinity corresponds to the JSON schema field "podAffinity".
spec.kafka.zookeeper.template.pod.affinity.podAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.template.pod.affinity.podAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.template.pod.affinity.podAntiAffinity	object	PodAntiAffinity corresponds to the JSON schema field "podAntiAffinity".
spec.kafka.zookeeper.template.pod.affinity.podAntiAffinity.preferredDuringSchedulingIgnoredDuringExecution	array	PreferredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "preferredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.template.pod.affinity.podAntiAffinity.requiredDuringSchedulingIgnoredDuringExecution	array	RequiredDuringSchedulingIgnoredDuringExecution corresponds to the JSON schema field "requiredDuringSchedulingIgnoredDuringExecution".
spec.kafka.zookeeper.template.pod.hostAliases	array	The pod's HostAliases. HostAliases is an optional list of hosts and IPs that will be injected into the pod's hosts file if specified.

Property	Type	Description
spec.kafka.zookeeper.template.pod.imagePullSecrets	array	List of references to secrets in the same namespace to use for pulling any of the images used by this Pod. When the **`STRIMZI_IMAGE_PULL_SECRETS`** environment variable in Cluster Operator and the **`imagePullSecrets`** option are specified, only the **`imagePullSecrets`** variable is used and the **`STRIMZI_IMAGE_PULL_SECRETS`** variable is ignored.
spec.kafka.zookeeper.template.pod.metadata	object	Metadata applied to the resource.
spec.kafka.zookeeper.template.pod.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.zookeeper.template.pod.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **`StatefulSets`**, **`Deployments`**, **`Pods`**, and **`Services`**.
spec.kafka.zookeeper.template.pod.priorityClassName	string	The name of the priority class used to assign priority to the pods. For more information about priority classes, see {K8sPriorityClass}.
spec.kafka.zookeeper.template.pod.schedulerName	string	The name of the scheduler used to dispatch this **`Pod`**. If not specified, the default scheduler will be used.
spec.kafka.zookeeper.template.pod.securityContext	object	Configures pod-level security attributes and common container settings.
spec.kafka.zookeeper.template.pod.securityContext.fsGroup	integer	FsGroup corresponds to the JSON schema field "fsGroup".
spec.kafka.zookeeper.template.pod.securityContext.fsGroupChangePolicy	string	FsGroupChangePolicy corresponds to the JSON schema field "fsGroupChangePolicy".
spec.kafka.zookeeper.template.pod.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".
spec.kafka.zookeeper.template.pod.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".

Property	Type	Description
spec.kafka.zookeeper.template.pod.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.zookeeper.template.pod.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.zookeeper.template.pod.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.zookeeper.template.pod.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.zookeeper.template.pod.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.zookeeper.template.pod.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.zookeeper.template.pod.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".
spec.kafka.zookeeper.template.pod.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.zookeeper.template.pod.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.zookeeper.template.pod.securityContext.supplementalGroups	array	SupplementalGroups corresponds to the JSON schema field "supplementalGroups".

Property	Type	Description
spec.kafka.zookeeper.template.pod.securityContext.sysctls	array	Sysctls corresponds to the JSON schema field "sysctls".
spec.kafka.zookeeper.template.pod.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.zookeeper.template.pod.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.zookeeper.template.pod.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.zookeeper.template.pod.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.zookeeper.template.pod.terminationGracePeriodSeconds	integer	The grace period is the duration in seconds after the processes running in the pod are sent a termination signal, and the time when the processes are forcibly halted with a kill signal. Set this value to longer than the expected cleanup time for your process. Value must be a non-negative integer. A zero value indicates delete immediately. You might need to increase the grace period for very large Kafka clusters, so that the Kafka brokers have enough time to transfer their work to another broker before they are terminated. Defaults to 30 seconds.
spec.kafka.zookeeper.template.pod.tolerations	array	The pod's tolerations.
spec.kafka.zookeeper.template.pod.topologySpreadConstraints	array	The pod's topology spread constraints.
spec.kafka.zookeeper.template.podDisruptionBudget	object	Template for ZooKeeper `PodDisruptionBudget`.

Property	Type	Description
spec.kafka.zookeeper.template.podDisruptionBudget.maxUnavailable	integer	Maximum number of unavailable pods to allow automatic Pod eviction. A Pod eviction is allowed when the **maxUnavailable** number of pods or fewer are unavailable after the eviction. Setting this value to 0 prevents all voluntary evictions, so the pods must be evicted manually. Defaults to 1.
spec.kafka.zookeeper.template.podDisruptionBudget.metadata	object	Metadata to apply to the **PodDistruptionBugetTemplate** resource.
spec.kafka.zookeeper.template.podDisruptionBudget.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **StatefulSets**, **Deployments**, **Pods**, and **Services**.
spec.kafka.zookeeper.template.podDisruptionBudget.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **StatefulSets**, **Deployments**, **Pods**, and **Services**.
spec.kafka.zookeeper.template.statefulset	object	Template for ZooKeeper **StatefulSet**.
spec.kafka.zookeeper.template.statefulset.metadata	object	Metadata applied to the resource.
spec.kafka.zookeeper.template.statefulset.metadata.annotations	object	Annotations added to the resource template. Can be applied to different resources such as **StatefulSets**, **Deployments**, **Pods**, and **Services**.
spec.kafka.zookeeper.template.statefulset.metadata.labels	object	Labels added to the resource template. Can be applied to different resources such as **StatefulSets**, **Deployments**, **Pods**, and **Services**.
spec.kafka.zookeeper.template.statefulset.podManagementPolicy	string	PodManagementPolicy which will be used for this StatefulSet. Valid values are **Parallel** and **OrderedReady**. Defaults to **Parallel**.
spec.kafka.zookeeper.template.tlsSidecarContainer	object	Template for the Zookeeper server TLS sidecar container. The TLS sidecar is not used anymore and this option will be ignored.
spec.kafka.zookeeper.template.tlsSidecarContainer.env	array	Environment variables which should be applied to the container.

Property	Type	Description
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext	object	Security context for the container.
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".

Property	Type	Description
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".

Property	Type	Description
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.zookeeper.template.tlsSidecarContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.zookeeper.template.zookeeperContainer	object	Template for the ZooKeeper container.
spec.kafka.zookeeper.template.zookeeperContainer.env	array	Environment variables which should be applied to the container.

Property	Type	Description
spec.kafka.zookeeper.template.zookeeperContainer.securityContext	object	Security context for the container.
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.allowPrivilegeEscalation	boolean	AllowPrivilegeEscalation corresponds to the JSON schema field "allowPrivilegeEscalation".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.capabilities	object	Capabilities corresponds to the JSON schema field "capabilities".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.capabilities.add	array	Add corresponds to the JSON schema field "add".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.capabilities.drop	array	Drop corresponds to the JSON schema field "drop".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.privileged	boolean	Privileged corresponds to the JSON schema field "privileged".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.procMount	string	ProcMount corresponds to the JSON schema field "procMount".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.readOnlyRootFilesystem	boolean	ReadOnlyRootFilesystem corresponds to the JSON schema field "readOnlyRootFilesystem".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.runAsGroup	integer	RunAsGroup corresponds to the JSON schema field "runAsGroup".

Property	Type	Description
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.runAsNonRoot	boolean	RunAsNonRoot corresponds to the JSON schema field "runAsNonRoot".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.runAsUser	integer	RunAsUser corresponds to the JSON schema field "runAsUser".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seLinuxOptions	object	SeLinuxOptions corresponds to the JSON schema field "seLinuxOptions".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seLinuxOptions.level	string	Level corresponds to the JSON schema field "level".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seLinuxOptions.role	string	Role corresponds to the JSON schema field "role".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seLinuxOptions.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seLinuxOptions.user	string	User corresponds to the JSON schema field "user".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seccompProfile	object	SeccompProfile corresponds to the JSON schema field "seccompProfile".

Property	Type	Description
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seccompProfile.localhostProfile	string	LocalhostProfile corresponds to the JSON schema field "localhostProfile".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.seccompProfile.type	string	Type corresponds to the JSON schema field "type".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.windowsOptions	object	WindowsOptions corresponds to the JSON schema field "windowsOptions".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.windowsOptions.gmsaCredentialSpec	string	GmsaCredentialSpec corresponds to the JSON schema field "gmsaCredentialSpec".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.windowsOptions.gmsaCredentialSpecName	string	GmsaCredentialSpecName corresponds to the JSON schema field "gmsaCredentialSpecName".
spec.kafka.zookeeper.template.zookeeperContainer.securityContext.windowsOptions.runAsUserName	string	RunAsUserName corresponds to the JSON schema field "runAsUserName".
spec.kafka.zookeeper.tlsSidecar	object	TLS sidecar configuration. The TLS sidecar is not used anymore and this option will be ignored.
spec.kafka.zookeeper.tlsSidecar.image	string	The docker image for the container.
spec.kafka.zookeeper.tlsSidecar.livenessProbe	object	Pod liveness checking.

Property	Type	Description
spec.kafka.zookeeper.tlsSidecar.livenessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.zookeeper.tlsSidecar.livenessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.zookeeper.tlsSidecar.livenessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.zookeeper.tlsSidecar.livenessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.zookeeper.tlsSidecar.livenessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.
spec.kafka.zookeeper.tlsSidecar.logLevel	string	The log level for the TLS sidecar. Default value is `notice`.
spec.kafka.zookeeper.tlsSidecar.readinessProbe	object	Pod readiness checking.
spec.kafka.zookeeper.tlsSidecar.readinessProbe.failureThreshold	integer	Minimum consecutive failures for the probe to be considered failed after having succeeded. Defaults to 3. Minimum value is 1.
spec.kafka.zookeeper.tlsSidecar.readinessProbe.initialDelaySeconds	integer	The initial delay before first the health is first checked. Default to 15 seconds. Minimum value is 0.
spec.kafka.zookeeper.tlsSidecar.readinessProbe.periodSeconds	integer	How often (in seconds) to perform the probe. Default to 10 seconds. Minimum value is 1.
spec.kafka.zookeeper.tlsSidecar.readinessProbe.successThreshold	integer	Minimum consecutive successes for the probe to be considered successful after having failed. Defaults to 1. Must be 1 for liveness. Minimum value is 1.
spec.kafka.zookeeper.tlsSidecar.readinessProbe.timeoutSeconds	integer	The timeout for each attempted health check. Default to 5 seconds. Minimum value is 1.

Property	Type	Description
spec.kafka.zookeeper.tlsSidecar.resources	object	CPU and memory resources to reserve.
spec.kafka.zookeeper.tlsSidecar.resources.limits	object	Limits corresponds to the JSON schema field "limits".
spec.kafka.zookeeper.tlsSidecar.resources.requests	object	Requests corresponds to the JSON schema field "requests".
spec.kafka.zookeeper.tolerations	array	The pod's tolerations.
spec.kafkaAccessSecret	string	KafkaAccessSecret when set will configure the application to use the provided kafka credentials instead of deploying its own.
spec.license	object	LicenseAccept is used to accept the license agreement
spec.license.accept	boolean	(No Description)
spec.postgres	object	Postgres property
spec.postgres.accessSecret	string	Access secret name that contains the Postgres service name and port, username, password, and CA certificate.
spec.size	string	(No Description)
spec.storageClass	string	StorageClass is the storage class used for PVCs. If not set it will fall back to the cluster default when available.

status

Description: CommonStatus that can be used with StatusUpdater

Type: object

Table 1. Description of AIOpsEdge/v1beta1 status

Property	Type	Description
status	object	CommonStatus that can be used with StatusUpdater
status.conditions	array	(No Description)
status.observedGeneration	integer	(No Description)
status.phase	string	(No Description)
status.resources	object	Resources contained in status

AIOpsUI/v1

For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms. AIOpsUI is the Schema for the aiopsuis API.

spec

Description: (No Description)

Type: object

Table 1. Description of AIOpsUI/v1 spec

Property	Type	Description
spec	object	(No Description)
spec.configuration	object	Additional configuration for the UI deployment
spec.configuration.topologyModel	string	TopologyModel - Specify the terminology to use in CP4WAIOps when defining collections of topology resource groups. Valid values are "application" or "service". If this setting is not specified, the default value will be "application".
spec.container	object	ContainerSpec - possible configurations around containers
spec.container.applicationUiApi	object	(No Description)
spec.container.applicationUiApi.container	object	ContainerSpec - possible configurations around containers
spec.container.applicationUiApi.container.containerPort	integer	(No Description)
spec.container.applicationUiApi.container.env	array	(No Description)
spec.container.applicationUiApi.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.applicationUiApi.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.applicationUiApi.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.applicationUiApi.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.applicationUiApi.image	object	ImageSpec - specification around how to pull an image.
spec.container.applicationUiApi.image.path	string	(No Description)

Property	Type	Description
spec.container.applicationUiApi.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.applicationUiApi.image.pullSecret	string	(No Description)
spec.container.applicationUiApi.tlsSecretName	string	(No Description)
spec.container.containerPort	integer	(No Description)
spec.container.env	array	(No Description)
spec.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiApi	object	(No Description)
spec.container.uiApi.container	object	ContainerSpec - possible configurations around containers
spec.container.uiApi.container.containerPort	integer	(No Description)
spec.container.uiApi.container.env	array	(No Description)
spec.container.uiApi.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.uiApi.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.uiApi.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/

Property	Type	Description
spec.container.uiApi.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiApi.image	object	ImageSpec - specification around how to pull an image.
spec.container.uiApi.image.path	string	(No Description)
spec.container.uiApi.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.uiApi.image.pullSecret	string	(No Description)
spec.container.uiApi.tlsSecretName	string	(No Description)
spec.container.uiBundleApi	object	(No Description)
spec.container.uiBundleApi.container	object	ContainerSpec - possible configurations around containers
spec.container.uiBundleApi.container.containerPort	integer	(No Description)
spec.container.uiBundleApi.container.env	array	(No Description)
spec.container.uiBundleApi.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.uiBundleApi.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.uiBundleApi.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiBundleApi.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/

Property	Type	Description
spec.container.uiBundleApi.image	object	ImageSpec - specification around how to pull an image.
spec.container.uiBundleApi.image.path	string	(No Description)
spec.container.uiBundleApi.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.uiBundleApi.image.pullSecret	string	(No Description)
spec.container.uiBundleApi.tlsSecretName	string	(No Description)
spec.container.uiContent	object	(No Description)
spec.container.uiContent.container	object	ContainerSpec - possible configurations around containers
spec.container.uiContent.container.containerPort	integer	(No Description)
spec.container.uiContent.container.env	array	(No Description)
spec.container.uiContent.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.uiContent.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims,that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.uiContent.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiContent.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiContent.image	object	ImageSpec - specification around how to pull an image.
spec.container.uiContent.image.path	string	(No Description)

Property	Type	Description
spec.container.uiContent.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.uiContent.image.pullSecret	string	(No Description)
spec.container.uiContent.tlsSecretName	string	(No Description)
spec.container.uiServer	object	(No Description)
spec.container.uiServer.container	object	ContainerSpec - possible configurations around containers
spec.container.uiServer.container.containerPort	integer	(No Description)
spec.container.uiServer.container.env	array	(No Description)
spec.container.uiServer.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.uiServer.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container.This is an alpha field and requires enabling the DynamicResourceAllocation feature gate.This field is immutable. It can only be set for containers.
spec.container.uiServer.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiServer.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiServer.image	object	ImageSpec - specification around how to pull an image.
spec.container.uiServer.image.path	string	(No Description)
spec.container.uiServer.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.uiServer.image.pullSecret	string	(No Description)

Property	Type	Description
spec.container.uiServer.tlsSecretName	string	(No Description)
spec.elasticSearch	object	Contact details for ElasticSearch
spec.elasticSearch.bindingSecret	string	Data store binding secret
spec.image	object	ImageSpec - specification around how to pull an image.
spec.image.path	string	(No Description)
spec.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.image.pullSecret	string	(No Description)
spec.license	object	License - license spec determines whether a license has been accepted
spec.license.accept	boolean	License acceptance switch
spec.name	string	Name of the service being deployed
spec.redis	object	Name of the service being deployed
spec.redis.bindingSecret	string	Data store binding secret
spec.replicas	integer	(No Description)
spec.size	string	(No Description)
spec.tlsSecretName	string	(No Description)
spec.version	string	Version of the csv deploying
spec.zenserviceName	string	name of the zenservice cr that is used for this deployment

status

Description: (No Description)

Type: object

Table 1. Description of AIOpsUI/v1 status

Property	Type	Description
status	object	(No Description)
status.conditions	array	(No Description)
status.dependencies	object	Dependencies is a ad hoc field to store known depedency status. This field currently does not store all dependencies, but rather only depedencies that we have a vested interest in knowing the status of.
status.managedResources	array	(No Description)
status.phase	string	(No Description)
status.versions	object	(No Description)

Property	Type	Description
status.versions.reconciled	string	(No Description)

AIOpsUIExtension/v1

AIOpsUIExtension is the Schema for the aiopsuiextensions API

spec

Description: AIOpsUIExtensionSpec defines the desired state of AIOpsUIExtension

Type: object

Table 1. Description of AIOpsUIExtension/v1 spec

Property	Type	Description
spec	object	AIOpsUIExtensionSpec defines the desired state of AIOpsUIExtension
spec.frameSrcUrls	string	FrameSourceUrls contains any frame-srcs that will be applied to our Akora UI server CSP.
spec.menuRoutes	(No Type)	MenuRoutes contains menu configurations, that will be mapped to the formats required by Zen and Akora.
spec.routes	(No Type)	Routes contains page configuration and bundle configurations.

status

Description: (No Description)

Type: object

Table 1. Description of AIOpsUIExtension/v1 status

Property	Type	Description
status	object	(No Description)
status.conditions	array	(No Description)
status.dependencies	object	Dependencies is a ad hoc field to store known depedency status. This field currently does not store all dependencies, but rather only depedencies that we have a vested interest in knowing the status of.
status.managedResources	array	(No Description)
status.phase	string	(No Description)
status.versions	object	(No Description)
status.versions.reconciled	string	(No Description)

AIManager/v1beta1

AIManager Deploys the IBM AIOps AI Manager service components of IBM Cloud Pak for AIOps. For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.

spec

Description: AIManagerSpec defines the desired state of AIManager

Type: object

Table 1. Description of AIManager/v1beta1 spec

Property	Type	Description
spec	object	AIManagerSpec defines the desired state of AIManager
spec.aimanagerValues	object	Plumbing Values Override (hidden)
spec.cartridge	object	Cartridge Name
spec.cartridge.name	string	Cartridge Name
spec.certificates	object	Certificates contains information required to create and trust certificates for internal TLS communication.
spec.certificates.caConfigMapKey	string	Key in the ConfigMap identified by caConfigMapName that contains root certificates to be trusted for internal TLS.
spec.certificates.caConfigMapName	string	Name of the ConfigMap that contains root certificates to be trusted for internal TLS.
spec.certificates.issuer	string	Name of the cert-manager.io/Issuer that will be used to create TLS certificates.
spec.chatOpsDisabled	boolean	ChatOpsDisabled (hidden)
spec.chatOpsSlackDisabled	boolean	ChatOpsSlackDisabled (hidden)
spec.chatOpsTeamDisabled	boolean	ChatOpsTeamDisabled (hidden)
spec.chatopsAuthentication	object	ChatopsAuthentication (hidden)
spec.chatopsAuthentication.configmapName	string	(No Description)
spec.chatopsAuthentication.secretName	string	(No Description)
spec.controllerDisabled	boolean	ControllerDisabled (hidden)
spec.cronjobConfig	object	Cronjob Configuration Properties
spec.cronjobConfig.unitCount	integer	Unit Count
spec.debug	object	Debug Options (hidden)
spec.debug.devPullSecrets	array	(No Description)
spec.debug.disableAIManager	boolean	(No Description)
spec.debug.disableElastic	boolean	(No Description)
spec.debug.disableKafka	boolean	(No Description)

Property	Type	Description
spec.debug.disableModeltrain	boolean	Note: this element is no longer used and should be removed in the next major version of this API
spec.debug.disableZen	boolean	(No Description)
spec.debug.enableCustomSCC	boolean	(No Description)
spec.debug.useDockerIOForKafka	boolean	(No Description)
spec.directFlink	boolean	(No Description)
spec.elasticsearch	object	Elastic Search
spec.elasticsearch.elasticSuperuserSecretName	string	ElasticUserSecretName
spec.elasticsearch.elasticUserSecretName	string	ElasticUserSecretName
spec.eventProcessor	object	Event Processor
spec.eventProcessor.name	string	Event Processor Name
spec.existingPullSecret	object	Existing Pull Secret
spec.existingPullSecret.name	string	Secret Name
spec.flink	object	Flink
spec.flink.fdSecretName	string	Flink Deployment Secret Name
spec.flink.secretName	string	Flink Secret Name
spec.integratorProxy	object	Integrator Proxy Config Properties
spec.integratorProxy.enabled	boolean	A proxy outgoing Chatops routes through.
spec.integratorProxy.proxyRoute	string	(No Description)
spec.kafka	object	Kafka
spec.kafka.secretName	string	Kafka Secret Name
spec.license	object	Please accept the licenses to allow for installation
spec.license.accept	boolean	I have read and accept the license: https://ibm.biz/cp4aiops-4100-license
spec.logAnomaly	object	LogAnomalyDetectorConfig (hidden)
spec.logAnomaly.limits	object	(No Description)
spec.logAnomaly.limits.cpu	string	(No Description)
spec.logAnomaly.limits.memory	string	(No Description)
spec.logAnomaly.replicas	integer	(No Description)
spec.logAnomaly.requests	object	(No Description)

Property	Type	Description
spec.logAnomaly.requests.cpu	string	(No Description)
spec.logAnomaly.requests.memory	string	(No Description)
spec.logAnomaly.serverPools	integer	(No Description)
spec.logAnomalyDisabled	boolean	LogAnomalyDisabled (hidden)
spec.minio	object	Minio Properties
spec.minio.customSizing	object	MinioCustomResources (hidden)
spec.minio.customSizing.limits	object	(No Description)
spec.minio.customSizing.limits.cpu	string	(No Description)
spec.minio.customSizing.limits.memory	string	(No Description)
spec.minio.customSizing.replicas	integer	(No Description)
spec.minio.customSizing.requests	object	(No Description)
spec.minio.customSizing.requests.cpu	string	(No Description)
spec.minio.customSizing.requests.memory	string	(No Description)
spec.minio.customSizing.serverPools	integer	(No Description)
spec.minio.storage	object	The Minio storage options
spec.minio.storage.storageClassName	string	The StorageClassName
spec.minio.storage.storageSize	string	The Size of Storage (format example "100Gi")
spec.modelTrain	object	Model Train Properties
spec.modelTrain.maxLearners	integer	Maximum number of training to run at once
spec.modelTrain.pollInterval	integer	Poll Interval for the Queue
spec.modelTrain.sizeLimit	integer	Size Limit for the Queue
spec.postgresDB	object	Postgres Database Instance Properties
spec.postgresDB.edbSecretName	string	EDB Postgresql Database Connection Secret Name
spec.postgresDB.secretName	string	Postgresql Database Connection Secret Name
spec.redis	object	Redis
spec.redis.bindingSecretName	string	Redis Binding Secret Name

Property	Type	Description
spec.repo	string	The private registry to pull images from instead of cp.icr.io. Leave blank if a private registry is not used. All images must be pushed to repo via mirroring.
spec.size	string	Preconfigured HA Sizing Options
spec.storage	object	The storage options
spec.storage.enabled	boolean	Enable Storage
spec.storage.storageClassName	string	The StorageClassName
spec.ticketAnalysisDisabled	boolean	TicketAnalysisDisabled (hidden)
spec.version	string	Version
spec.zen	object	The Zen options
spec.zen.deploy	boolean	Flag to deploy our own cpd in the same namespace as aimanager
spec.zen.ignoreReady	boolean	Flag to deploy zen customization even if not in ready state
spec.zen.instanceId	string	ID of Zen Service Instance
spec.zen.serviceInstanceName	string	Name of the ZenService Instance object
spec.zen.serviceNamespace	string	Namespace of the ZenService Instance
spec.zen.storage	object	The Zen storage options
spec.zen.storage.storageClassName	string	The StorageClassName

status

Description: AIManagerStatus defines the observed state of AIManager

Type: object

Table 1. Description of AIManager/v1beta1 status

Property	Type	Description
status	object	AIManagerStatus defines the observed state of AIManager
status.message	string	Status information from AIManager
status.phase	string	Current phase of AIManager.
status.versions	object	Information on versions for AI Manager
status.versions.available	object	Available properties
status.versions.available.versions	array	Possible available versions for AI Manager
status.versions.reconciled	string	Current version of the reconciled instance

Algorithm/v1alpha1

Algorithm is the Schema for the algorithms API

spec

Description: AlgorithmSpec defines the desired state of Algorithm

Type: object

Table 1. Description of Algorithm/v1alpha1 spec

Property	Type	Description
spec	object	AlgorithmSpec defines the desired state of Algorithm
spec.description	string	(No Description)
spec.enabled	boolean	(No Description)
spec.manifestBase64	string	(No Description)
spec.name	string	(No Description)
spec.registered	boolean	(No Description)
spec.runtime	string	(No Description)
spec.usedStatus	array	(No Description)
spec.version	string	(No Description)

status

Description: AlgorithmStatus defines the observed state of Algorithm

Type: object

Table 1. Description of Algorithm/v1alpha1 status

Property	Type	Description
status	object	AlgorithmStatus defines the observed state of Algorithm

Algorithm/v1beta1

Algorithm is the Schema for the algorithms API

spec

Description: AlgorithmSpec defines the desired state of Algorithm

Type: object

Table 1. Description of Algorithm/v1beta1 spec

Property	Type	Description
spec	object	AlgorithmSpec defines the desired state of Algorithm
spec.description	string	(No Description)
spec.enabled	boolean	(No Description)
spec.manifestBase64	string	(No Description)
spec.name	string	(No Description)
spec.registered	boolean	(No Description)

Property	Type	Description
spec.runtime	string	(No Description)
spec.usedStatus	array	(No Description)
spec.version	string	(No Description)

status

Description: AlgorithmStatus defines the observed state of Algorithm

Type: object

Table 1. Description of Algorithm/v1beta1 status

Property	Type	Description
status	object	AlgorithmStatus defines the observed state of Algorithm

ASM/v1beta1

spec

Description: ASMSpec defines the desired state of ASM

Type: object

Table 1. Description of ASM/v1beta1 spec

Property	Type	Description
spec	object	ASMSpec defines the desired state of ASM
spec.advanced	object	Advanced image properties
spec.advanced.imagePullPolicy	string	The default pull policy is IfNotPresent, which causes the Kubelet to skip pulling an image that already exists.
spec.advanced.imagePullRepository	string	Docker registry that all component images are pulled from
spec.asm	object	Topology components
spec.asm.aaionap	boolean	Do you want to deploy AAI ONAP Service
spec.asm.almObserver	boolean	Do you want to deploy ALM Observer
spec.asm.ansibleawxObserver	boolean	Do you want to deploy Ansible AWX Observer
spec.asm.appDisco	object	Application Discovery properties
spec.asm.appDisco.caCertProperty	string	Cert Property
spec.asm.appDisco.caCertSecret	string	Cert Secret
spec.asm.appDisco.caKeyProperty	string	Key Property
spec.asm.appDisco.certSecret	string	DB2 Certificate Secret

Property	Type	Description
spec.asm.appDisco.dbsecret	string	DB2 secret
spec.asm.appDisco.dburl	string	DB2 host to connect
spec.asm.appDisco.enabled	boolean	Enable Application Discovery and Application Discovery Observer
spec.asm.appDisco.scaleDS	integer	(No Description)
spec.asm.appDisco.scaleSSS	integer	(No Description)
spec.asm.appDisco.secure	boolean	Secure DB connection
spec.asm.appDisco.tlsSecret	string	(No Description)
spec.asm.appdynamicsObserver	boolean	Do you want to deploy AppDynamics Observer
spec.asm.awsObserver	boolean	Do you want to deploy AWS Observer
spec.asm.azureObserver	boolean	Do you want to deploy Microsoft Azure Observer
spec.asm.bigcloudfabricObserver	boolean	Do you want to deploy BigCloud Fabric Observer
spec.asm.bigfixinventoryObserver	boolean	Do you want to deploy BigFix Inventory Observer
spec.asm.cienablueplanetObserver	boolean	Do you want to deploy Ciena Blue Planet Observer
spec.asm.ciscoaciObserver	boolean	Do you want to deploy Cisco ACI Observer
spec.asm.contrailObserver	boolean	Do you want to deploy Juniper Contrail Observer
spec.asm.datadogObserver	boolean	Do you want to deploy Datadog Observer
spec.asm.dnsObserver	boolean	Do you want to deploy DNS Observer
spec.asm.dockerObserver	boolean	Do you want to deploy Docker Observer
spec.asm.dynatraceObserver	boolean	Do you want to deploy Dynatrace Observer
spec.asm.externalRisksObserver	boolean	Do you want to deploy External Risks Observer
spec.asm.fileObserver	boolean	Do you want to deploy File Observer
spec.asm.gitlabObserver	boolean	Do you want to deploy GitLab Observer
spec.asm.googlecloudObserver	boolean	Do you want to deploy Google Cloud Platform Observer
spec.asm.hpnfvdObserver	boolean	Do you want to deploy HPNFVD Observer
spec.asm.iafCartridgeRequirementsName	string	IAF CartridgeRequirements Instance Name for connection to IAF ElasticSearch
spec.asm.iafElasticBindingSecret	string	IAF ElasticSearch Binding Secret for connection to IAF ElasticSearch

Property	Type	Description
spec.asm.iafKafkaBindingSecret	string	IAF Kafka Binding Secret for connection to IAF Kafka
spec.asm.ibmcloudObserver	boolean	Do you want to deploy IBM Cloud Observer (default = false)
spec.asm.instanaObserver	boolean	Do you want to deploy Instana Observer
spec.asm.itnmObserver	boolean	Do you want to deploy ITNM Observer
spec.asm.jenkinsObserver	boolean	Do you want to deploy Jenkins Observer
spec.asm.junipercsoObserver	boolean	Do you want to deploy Juniper Networks CSO Observer
spec.asm.kubernetesObserver	boolean	Do you want to deploy Kubernetes Observer (default = true)?
spec.asm.netDisco	boolean	Do you want to deploy Network Discovery and Network Discovery Observer
spec.asm.newrelicObserver	boolean	Do you want to deploy New Relic Observer
spec.asm.openstackObserver	boolean	Do you want to deploy OpenStack Observer
spec.asm.rancherObserver	boolean	Do you want to deploy Rancher Observer
spec.asm.restObserver	boolean	Do you want to deploy REST Observer
spec.asm.sdconapObserver	boolean	Do you want to deploy SDC ONAP Observer
spec.asm.servicenowObserver	boolean	Do you want to deploy ServiceNow Observer
spec.asm.sevoneObserver	boolean	Do you want to deploy SevOne Observer
spec.asm.taddmObserver	boolean	Do you want to deploy TADDM Observer
spec.asm.userServiceUrl	string	CEM User Service Url connection
spec.asm.viptelaObserver	boolean	Do you want to deploy Viptela Observer
spec.asm.vmvcenterObserver	boolean	Do you want to deploy VMware VCenter Observer
spec.asm.vmwarensxObserver	boolean	Do you want to deploy VMware NSX Observer
spec.asm.zabbixObserver	boolean	Do you want to deploy Zabbix Observer
spec.clusterDomain	string	Use the fully qualified domain name (FQDN) to specify the cluster domain: apps. ...com
spec.deploymentType	string	Deployment type (small or large)
spec.entitlementSecret	string	Entitlment secret to pull images
spec.helmValuesASM	object	(Not shown within UI)
spec.ingress	object	Vanilla Kube (Not shown within UI)
spec.ingress.port	string	(No Description)
spec.ingress.prefixWithReleaseName	boolean	(No Description)
spec.ingress.tlsSecret	string	(No Description)

Property	Type	Description
spec.license	object	License properties
spec.license.accept	boolean	If you have purchased IBM Cloud Pak for AIOps, you must read and accept this license: https://ibm.biz/cp4aiops-480-license.
spec.persistence	object	Persistence properties
spec.persistence.enabled	boolean	Enable persistence storage
spec.persistence.storageClass	string	Storage Class
spec.persistence.storageSizeCassandraBackup	string	Storage Size Cassandra Backup
spec.persistence.storageSizeCassandraData	string	Storage Size Cassandra Data
spec.persistence.storageSizeElastic	string	Storage Size Elasticsearch
spec.persistence.storageSizeFileObserver	string	Storage size File Observer .
spec.persistence.storageSizeKafka	string	Storage Size Kafka
spec.persistence.storageSizeSdconapObserver	string	Storage Size SDCONAP Observer
spec.persistence.storageSizeZookeeper	string	Storage Size Zookeeper
spec.version	string	Version
spec.zen	object	The Zen Parameters
spec.zen.deploy	boolean	Flag to deploy our own cpd in the same namespace as aimanager
spec.zen.ignoreReady	boolean	Flag to deploy zen customization even if not in ready state
spec.zen.instanceId	string	ID of Zen Service Instance
spec.zen.serviceInstanceName	string	Name of the ZenService Instance object
spec.zen.serviceNamespace	string	Namespace of the ZenService Instance
spec.zen.storage	object	The Zen storage options
spec.zen.storage.storageClassName	string	The StorageClassName

status

Description: (No Description)

Type: object

Table 1. Description of ASM/v1beta1 status

Property	Type	Description
status	object	(No Description)
status.conditions	array	(No Description)
status.managedResources	array	Resources managed by Netcool Agile Service Manager

Property	Type	Description
status.message	string	(No Description)
status.phase	string	Current phase of installed full Cloud deployment.
status.postgreSQL	object	PostgreSQL contains information about PostgreSQL status
status.postgreSQL.databases	array	Databases contains information about individual databases
status.postgreSQL.time	string	Time at which the status was last checked
status.versions	object	Information on versions for Agile Service Manager
status.versions.available	object	(No Description)
status.versions.available.versions	array	Possible available versions for Agile Service Manager
status.versions.reconciled	string	Current version of the reconciled instance

BaseUI/v1

For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.

spec

Description: (No Description)

Type: object

Table 1. Description of BaseUI/v1 spec

Property	Type	Description
spec	object	(No Description)
spec.bedrockVersion	string	(No Description)
spec.certificates	object	Certificates contains information required to create and trust certificates for internal TLS communication.
spec.certificates.caConfigMapKey	string	Key in the ConfigMap identified by caConfigMapName that contains root certificates to be trusted for internal TLS.
spec.certificates.caConfigMapName	string	Name of the ConfigMap that contains root certificates to be trusted for internal TLS.
spec.certificates.issuer	string	Name of the cert-manager.io/Issuer that will be used to create TLS certificates.
spec.configuration	object	Additional configuration for the UI deployment
spec.configuration.topologyModel	string	TopologyModel - Specify the terminology to use in CP4WAIOps when defining collections of topology resource groups. Valid values are "application" or "service". If this setting is not specified, the default value will be "application".

Property	Type	Description
spec.container	object	ContainerSpec - possible configurations around containers
spec.container.containerPort	integer	(No Description)
spec.container.env	array	(No Description)
spec.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.image	object	ImageSpec - specification around how to pull an image.
spec.image.path	string	(No Description)
spec.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.image.pullSecret	string	(No Description)
spec.license	object	License - license spec determines whether a license has been accepted
spec.license.accept	boolean	License acceptance switch
spec.modules	object	Modules is a list of module settings for the operator. This value will be default for 4.6.x and will eventually be required.
spec.modules.groupProfiles	object	(No Description)
spec.modules.groupProfiles.enabled	boolean	Enabled toggles the module as enabled or disabled, by default modules are enabled
spec.modules.logs	object	(No Description)
spec.modules.logs.enabled	boolean	Enabled toggles the module as enabled or disabled, by default modules are enabled
spec.name	string	Name of the service being deployed
spec.replicas	integer	(No Description)
spec.size	string	(No Description)
spec.tlsSecretName	string	(No Description)
spec.version	string	Version of the csv deploying
spec.zenserviceName	string	name of the zenservice cr that is used for this deployment

status

Description: (No Description)

Type: object

Table 1. Description of BaseUI/v1 status

Property	Type	Description
status	object	(No Description)
status.conditions	array	(No Description)
status.managedResources	array	(No Description)
status.phase	string	(No Description)
status.versions	object	(No Description)
status.versions.reconciled	string	(No Description)

BundleManifest/v1alpha1

BundleManifest is the Schema for the bundlemanifests API

spec

Description: BundleManifestSpec defines the desired state of BundleManifest

Type: object

Table 1. Description of BundleManifest/v1alpha1 spec

Property	Type	Description
spec	object	BundleManifestSpec defines the desired state of BundleManifest
spec.instanced	object	GitAppSpec defines the desired state of GitApp
spec.instanced.authSecret	object	AuthSecret is the name of the secret containing credentials for Basic-Auth in the username and password keys
spec.instanced.authSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.instanced.branch	string	Branch to checkout
spec.instanced.certSecret	object	CertSecret is the name of the secret containing the ca.crt needed to access the repo if the origin uses a self-signed certificate
spec.instanced.certSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.instanced.components	array	(No Description)

Property	Type	Description
spec.instanced.repo	string	Repo containing deployment files
spec.prereqs	object	GitAppSpec defines the desired state of GitApp
spec.prereqs.authSecret	object	AuthSecret is the name of the secret containing credentials for Basic-Auth in the username and password keys
spec.prereqs.authSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.prereqs.branch	string	Branch to checkout
spec.prereqs.certSecret	object	CertSecret is the name of the secret containing the ca.crt needed to access the repo if the origin uses a self-signed certificate
spec.prereqs.certSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.prereqs.components	array	(No Description)
spec.prereqs.repo	string	Repo containing deployment files

status

Description: CommonStatus that can be used with StatusUpdater

Type: object

Table 1. Description of BundleManifest/v1alpha1 status

Property	Type	Description
status	object	CommonStatus that can be used with StatusUpdater
status.conditions	array	(No Description)
status.observedGeneration	integer	(No Description)
status.phase	string	(No Description)
status.resources	object	Resources contained in status

BundleManifest/v1beta1

BundleManifest is the Schema for the bundlemanifests API

spec

Description: BundleManifestSpec defines the desired state of BundleManifest

Type: object

Table 1. Description of BundleManifest/v1beta1 spec

Property	Type	Description
spec	object	BundleManifestSpec defines the desired state of BundleManifest
spec.instanced	object	GitAppSpec defines the desired state of GitApp
spec.instanced.authSecret	object	AuthSecret is the name of the secret containing credentials for Basic-Auth in the username and password keys
spec.instanced.authSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.instanced.branch	string	Branch to checkout
spec.instanced.certSecret	object	CertSecret is the name of the secret containing the ca.crt needed to access the repo if the origin uses a self-signed certificate
spec.instanced.certSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.instanced.components	array	(No Description)
spec.instanced.repo	string	Repo containing deployment files
spec.prereqs	object	GitAppSpec defines the desired state of GitApp
spec.prereqs.authSecret	object	AuthSecret is the name of the secret containing credentials for Basic-Auth in the username and password keys
spec.prereqs.authSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.prereqs.branch	string	Branch to checkout
spec.prereqs.certSecret	object	CertSecret is the name of the secret containing the ca.crt needed to access the repo if the origin uses a self-signed certificate
spec.prereqs.certSecret.name	string	Name of the referent. More info: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Add other useful fields. apiVersion, kind, uid?
spec.prereqs.components	array	(No Description)
spec.prereqs.repo	string	Repo containing deployment files

status

Description: CommonStatus that can be used with StatusUpdater

Type: object

Table 1. Description of BundleManifest/v1beta1 status

Property	Type	Description
status	object	CommonStatus that can be used with StatusUpdater
status.conditions	array	(No Description)
status.observedGeneration	integer	(No Description)
status.phase	string	(No Description)
status.resources	object	Resources contained in status

ConnectorUI/v1

For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.

spec

Description: CommonSpecs - specs that are common to determining a deployment.

Type: object

Table 1. Description of ConnectorUI/v1 spec

Property	Type	Description
spec	object	CommonSpecs - specs that are common to determining a deployment.
spec.configuration	object	Additional configuration for the UI deployment
spec.configuration.topologyModel	string	TopologyModel - Specify the terminology to use in CP4WAIOps when defining collections of topology resource groups. Valid values are "application" or "service". If this setting is not specified, the default value will be "application".
spec.container	object	ContainerSpec - possible configurations around containers
spec.container.containerPort	integer	(No Description)
spec.container.env	array	(No Description)
spec.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/

Property	Type	Description
spec.image	object	ImageSpec - specification around how to pull an image.
spec.image.path	string	(No Description)
spec.image.pull Policy	string	PullPolicy describes a policy for if/when to pull a container image
spec.image.pull Secret	string	(No Description)
spec.license	object	License - license spec determines whether a license has been accepted
spec.license.acc ept	boolean	License acceptance switch
spec.name	string	Name of the service being deployed
spec.replicas	integer	(No Description)
spec.size	string	(No Description)
spec.tlsSecretN ame	string	(No Description)
spec.version	string	Version of the csv deploying
spec.zenservice Name	string	name of the zenservice cr that is used for this deployment

status

Description: (No Description)

Type: object

Table 1. Description of ConnectorUI/v1 status

Property	Type	Description
status	object	(No Description)
status.condition s	array	(No Description)
status.depende ncies	object	Dependencies is a ad hoc field to store known depedency status. This field currently does not store all dependencies, but rather only depedencies that we have a vested interest in knowing the status of.
status.managed Resources	array	(No Description)
status.phase	string	(No Description)
status.versions	object	(No Description)
status.versions.r econciled	string	(No Description)

InsightsUI/v1

For documentation regarding install parameters, see IBM Documentation. By installing this product you accept the license terms.

spec

Description: (No Description)

Type: object

Table 1. Description of InsightsUI/v1 spec

Property	Type	Description
spec	object	(No Description)
spec.asm	object	Contact details for ASM
spec.asm.bindingSecret	string	Data store binding secret
spec.configuration	object	Additional configuration for the UI deployment
spec.configuration.topologyModel	string	TopologyModel - Specify the terminology to use in CP4WAIOps when defining collections of topology resource groups. Valid values are "application" or "service". If this setting is not specified, the default value will be "application".
spec.container	object	ContainerSpec - possible configurations around containers
spec.container.containerPort	integer	(No Description)
spec.container.datarouter	object	(No Description)
spec.container.datarouter.containerPort	integer	(No Description)
spec.container.datarouter.env	array	(No Description)
spec.container.datarouter.image	object	ImageSpec - specification around how to pull an image.
spec.container.datarouter.image.path	string	(No Description)
spec.container.datarouter.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.datarouter.image.pullSecret	string	(No Description)
spec.container.datarouter.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.datarouter.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.datarouter.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/

Property	Type	Description
spec.container.datarouter.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.datarouter.tlsSecretName	string	(No Description)
spec.container.env	array	(No Description)
spec.container.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.ui	object	(No Description)
spec.container.ui.containerPort	integer	(No Description)
spec.container.ui.env	array	(No Description)
spec.container.ui.image	object	ImageSpec - specification around how to pull an image.
spec.container.ui.image.path	string	(No Description)
spec.container.ui.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.ui.image.pullSecret	string	(No Description)
spec.container.ui.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.ui.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.

Property	Type	Description
spec.container.ui.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.ui.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.ui.tlsSecretName	string	(No Description)
spec.container.uiApi	object	(No Description)
spec.container.uiApi.containerPort	integer	(No Description)
spec.container.uiApi.env	array	(No Description)
spec.container.uiApi.image	object	ImageSpec - specification around how to pull an image.
spec.container.uiApi.image.path	string	(No Description)
spec.container.uiApi.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.container.uiApi.image.pullSecret	string	(No Description)
spec.container.uiApi.resources	object	ResourceRequirements describes the compute resource requirements.
spec.container.uiApi.resources.claims	array	Claims lists the names of resources, defined in spec.resourceClaims, that are used by this container. This is an alpha field and requires enabling the DynamicResourceAllocation feature gate. This field is immutable. It can only be set for containers.
spec.container.uiApi.resources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiApi.resources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. Requests cannot exceed Limits. More info: https://kubernetes.io/docs/concepts/configuration/manage-resources-containers/
spec.container.uiApi.tlsSecretName	string	(No Description)
spec.demodatasets	array	(No Description)

Property	Type	Description
spec.elasticSearch	object	Contact details for ElasticSearch
spec.elasticSearch.bindingSecret	string	Data store binding secret
spec.image	object	ImageSpec - specification around how to pull an image.
spec.image.path	string	(No Description)
spec.image.pullPolicy	string	PullPolicy describes a policy for if/when to pull a container image
spec.image.pullSecret	string	(No Description)
spec.kafka	object	Contact details for Kafka
spec.kafka.bindingSecret	string	Data store binding secret
spec.license	object	License - license spec determines whether a license has been accepted
spec.license.accept	boolean	License acceptance switch
spec.name	string	Name of the service being deployed
spec.postgres	object	Contact details for postgres
spec.postgres.bindingSecret	string	Data store binding secret
spec.postgres.quiesce	boolean	Quiesce set this to true to stop writes to postgres
spec.replicas	integer	(No Description)
spec.size	string	(No Description)
spec.tlsSecretName	string	(No Description)
spec.version	string	Version of the csv deploying
spec.zenserviceName	string	name of the zenservice cr that is used for this deployment

status

Description: (No Description)

Type: object

Table 1. Description of InsightsUI/v1 status

Property	Type	Description
status	object	(No Description)
status.conditions	array	(No Description)
status.dependencies	object	Dependencies is a ad hoc field to store known depedency status. This field currently does not store all dependencies, but rather only depedencies that we have a vested interest in knowing the status of.
status.managedResources	array	(No Description)

Property	Type	Description
status.phase	string	(No Description)
status.versions	object	(No Description)
status.versions.reconciled	string	(No Description)

Installation/v1alpha1

Leverage advanced, explainable AI to resolve incidents faster, gain visibility into critical workloads, and intelligently manage key application resources. Learn More. By installing this product you accept the license terms.

spec

Description: For detailed information about the following settings, see the IBM Documentation at https://ibm.biz/cp4aiops_410_install

Type: object

Table 1. Description of Installation/v1alpha1 spec

Property	Type	Description
spec	object	For detailed information about the following settings, see the IBM Documentation at https://ibm.biz/cp4aiops_410_install
spec.automationFoundation	object	AutomationFoundation contains parameters that can be used to customize how the IBM Automation Foundation objects are configured.
spec.automationFoundation.automationBase	string	AutomationBase is the name of the IBM Automation Foundation "AutomationBase" CR instance that the Cloud Pak should use. If a name is not specified, and no "AutomationBase" instances exist in the namespace, the Cloud Pak will attempt to provision one.
spec.automationFoundation.automationUIConfig	string	AutomationUIConfig is the name of the IBM Automation Foundation "AutomationUIConfig" CR instance that the Cloud Pak should use. If a name is not specified, and no "AutomationUIConfig" instances exist in the namespace, the Cloud Pak will attempt to provision one.
spec.automationFoundation.enabled	boolean	The Enabled flag controls whether the Cloud Pak will register with IBM Automation Foundation. The default value is `true`. When the flag is set to `false`, the Cloud Pak will not create any Automation Foundation objects, including those referenced in other parts of the AutomationFoundation spec.
spec.enableConnectionModule	boolean	Use the Secure Tunnel to safely connect to remote data sources.
spec.groupProfiles	object	GroupProfiles contains settings for group profile features.
spec.groupProfiles.enabled	boolean	Enabled, if set to true, turns on the group profiles features.
spec.imagePullSecret	string	Select the image pull secret to use for the Cloud Pak. If no secret is selected, the global pull secret will be used.

Property	Type	Description
spec.installPlanApproval	string	Approval is the user approval policy for an InstallPlan. It must be one of "Automatic" or "Manual".
spec.license	object	License acceptance is required to create resources.
spec.license.accept	boolean	I accept the product license after reading it at https://ibm.biz/cp4aiops-4100-license
spec.pakModules	array	Pak Modules are modular pieces that may be enabled for deployment
spec.proxyServer	string	If the target cluster is air-gapped, specify the address and port number of an HTTP proxy that will provide access to the Slack or MS Teams service used for ChatOps, for example, proxy.company.com:8080.
spec.resourceOverrides	string	Specify the name of a configmap that finely tunes the capabilities and capacity of an installation.
spec.size	string	Select a size to determine the amount of virtual resources requested and the level of high availability support.
spec.storage	object	CP4AIOps storage configuration.
spec.storage.data-aiops-ibm-elasticsearch-es-server-all	object	Override data-aiops-ibm-elasticsearch-es-server-all PVC properties. If in a multizone environment, PVC on the cluster will have a different suffix than `all`.
spec.storage.data-aiops-ibm-elasticsearch-es-server-all.size	string	Specify the size of the PVC for each replica with units.
spec.storage.data-aiops-ibm-elasticsearch-es-server-all.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.
spec.storage.data-aiops-topology-cassandra	object	Override data-aiops-topology-cassandra PVC properties.
spec.storage.data-aiops-topology-cassandra.size	string	Specify the size of the PVC for each replica with units.
spec.storage.data-aiops-topology-cassandra.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.
spec.storage.data-c-example-couchdbcluster-m	object	Override data-c-example-couchdbcluster-m PVC properties.
spec.storage.data-c-example-couchdbcluster-m.size	string	Specify the size of the PVC for each replica with units.
spec.storage.data-c-example-couchdbcluster-m.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.

Property	Type	Description
spec.storage.data-iaf-system-kafka	object	Override data-iaf-system-kafka PVC properties.
spec.storage.data-iaf-system-kafka.size	string	Specify the size of the PVC for each replica with units.
spec.storage.data-iaf-system-kafka.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.
spec.storage.data-ibm-cp-aiops-redis-server	object	Override data-ibm-cp-aiops-redis-server PVC properties.
spec.storage.data-ibm-cp-aiops-redis-server.size	string	Specify the size of the PVC for each replica with units.
spec.storage.data-ibm-cp-aiops-redis-server.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.
spec.storage.export-aimanager-ibm-minio	object	Override export-aimanager-ibm-minio PVC properties.
spec.storage.export-aimanager-ibm-minio.size	string	Specify the size of the PVC for each replica with units.
spec.storage.export-aimanager-ibm-minio.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.
spec.storage.ibm-cp-aiops-edb-postgres	object	Override ibm-cp-aiops-edb-postgres PVC properties. Based on installation name used, the prefix `ibm-cp-aiops` in the PVC name will match the installation name.
spec.storage.ibm-cp-aiops-edb-postgres.size	string	Specify the size of the PVC for each replica with units.
spec.storage.ibm-cp-aiops-edb-postgres.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.
spec.storage.zen-metastore-edb	object	Override zen-metastore-edb PVC properties.
spec.storage.zen-metastore-edb.size	string	Specify the size of the PVC for each replica with units.
spec.storage.zen-metastore-edb.storageClass	string	Specify the storage class for the PVC. The Installation storage classes will be used by default if no value is provided.

Property	Type	Description
spec.storageClass	string	Select a file storage class that supports ReadWriteMany access mode.
spec.storageClassLargeBlock	string	Select a storage class with a large block size (for example, 64k).
spec.topologyModel	string	Specify the terminology to use when defining collections of resource groups. You can select application or service.
spec.zones	array	Define the names of zones for a multi-zone cluster.

status

Description: InstallationStatus defines the observed state of Installation

Type: object

Table 1. Description of Installation/v1alpha1 status

Property	Type	Description
status	object	InstallationStatus defines the observed state of Installation
status.componentnamespace	string	(No Description)
status.componentstatus	object	(No Description)
status.customProfileConfigmap	string	(No Description)
status.customProfileValidationStatus	string	(No Description)
status.failureMessage	string	(No Description)
status.failureReason	string	(No Description)
status.imagePullSecret	string	(No Description)
status.licenseacceptance	string	LicenseAcceptanceStatus describes the status of the license acceptance
status.locations	object	(No Description)
status.locations.cloudPakUiUrl	string	Launch the Cloud Pak
status.locations.csAdminHubUrl	string	Manage services and processes
status.phase	string	InstallationPhase describes the phase of the overall Installation
status.size	string	(No Description)
status.storageclass	string	(No Description)
status.storageclasslargeblock	string	(No Description)

Tunnel/v1

Tunnel is the Schema for the tunnels API

spec

Description: SecureTunnelSpec defines the desired state of Tunnel

Type: object

Table 1. Description of Tunnel/v1 spec

Property	Type	Description
spec	object	SecureTunnelSpec defines the desired state of Tunnel
spec.ApiServerURL	string	(No Description)
spec.apiReplicas	integer	(No Description)
spec.apiServerURL	string	(No Description)
spec.applicationMappingPortRang	string	(No Description)
spec.auditReceiver	object	(No Description)
spec.auditReceiver.mongoDB	array	(No Description)
spec.auditReceiver.syslog	array	(No Description)
spec.auditReceivers	array	(No Description)
spec.commonServiceNamespace	string	(No Description)
spec.controllerReplicas	integer	(No Description)
spec.defaultTunnelWorkerResources	object	(No Description)
spec.defaultTunnelWorkerResources.limits	object	(No Description)
spec.defaultTunnelWorkerResources.limits.cpu	string	(No Description)
spec.defaultTunnelWorkerResources.limits.memory	string	(No Description)
spec.defaultTunnelWorkerResources.requests	object	(No Description)

Property	Type	Description
spec.defaultTunnelWorkerResources.requests.cpu	string	(No Description)
spec.defaultTunnelWorkerResources.requests.memory	string	(No Description)
spec.iamEndpoint	string	(No Description)
spec.imageDigest	string	(No Description)
spec.imageRepo	string	(No Description)
spec.imageTag	string	(No Description)
spec.ingressClass	string	(No Description)
spec.ingressDomain	string	(No Description)
spec.runAsStandAlone	boolean	(No Description)
spec.enableEgressNetworkPolicy	boolean	(No Description)
spec.enableAdminLogin	boolean	(No Description)
spec.connectionTLSCertDurationTime	string	(No Description)
spec.inventory	object	(No Description)
spec.inventory.inventoryPort	integer	(No Description)
spec.inventory.inventoryService	string	(No Description)
spec.inventory.namespace	string	(No Description)
spec.inventory.searchApiSecrets	string	(No Description)
spec.jwtPublicURL	string	(No Description)
spec.kubernetesEvn	boolean	(No Description)
spec.license	object	Opt-in license acceptance is required to create resources
spec.license.accept	boolean	(No Description)
spec.noManagementUI	boolean	(No Description)
spec.pullSecret	string	(No Description)
spec.runType	string	(No Description)

Property	Type	Description
spec.showInConsole	boolean	(No Description)
spec.showInContainerLog	boolean	(No Description)
spec.size	string	(No Description)
spec.tunnelWorkerResources	object	////
spec.tunnelWorkerResources.limits	object	Limits describes the maximum amount of compute resources allowed. More info: https://kubernetes.io/docs/concepts/configuration/manage-compute-resources-container/
spec.tunnelWorkerResources.requests	object	Requests describes the minimum amount of compute resources required. If Requests is omitted for a container, it defaults to Limits if that is explicitly specified, otherwise to an implementation-defined value. More info: https://kubernetes.io/docs/concepts/configuration/manage-compute-resources-container/
spec.uiImageDigest	string	(No Description)
spec.uiImageRepo	string	(No Description)
spec.uiImageTag	string	(No Description)
spec.uiReplicas	integer	(No Description)
spec.useTag	boolean	(No Description)
spec.version	string	(No Description)
spec.workerReplicas	integer	(No Description)
spec.zenCoreCertificateName	string	(No Description)

status

Description: SecureTunnelStatus defines the observed state of Tunnel

Type: object

Table 1. Description of Tunnel/v1 status

Property	Type	Description
status	object	SecureTunnelStatus defines the observed state of Tunnel
status.conditions	array	(No Description)
status.versions	object	INSERT ADDITIONAL STATUS FIELD - define observed state of cluster Important: Run "operator-sdk generate k8s" to regenerate code after modifying this file Add custom validation using kubebuilder tags: https://book-v1.book.kubebuilder.io/beyond_basics/generating_crd.html
status.versions.description	string	(No Description)

Property	Type	Description
status.versions.reconciled	string	(No Description)
status.versions.status	string	(No Description)
status.versions.tenantID	string	(No Description)

Infrastructure Automation custom resource definitions

Use this information to learn about the custom resource definitions (CRDs) used by Infrastructure Automation.

In the following table, CRDs that do not have a hyperlink must not be manually edited unless otherwise instructed, and are manipulated only by the operator.

Table 1. Infrastructure Automation CRDs

Kind	API version	Operator name	Description
IAConfig	v1alpha1	ibm-infrastructure-automation-operator	Build, deploy, and manage VMs and Kubernetes clusters. Enable self-service orchestration of resources across clouds. Learn More.
ManageService	v1alpha1	ibm-management-cam-install	ManageService is the Schema for the manageservices API
ManageIQ	v1alpha1	ibm-management-im-install	ManageIQ is the Schema for the manageiqs API
IMInstall	v1alpha1	ibm-management-im-install	IMInstall is the Schema for the iminstalls API

IAConfig/v1alpha1

Build, deploy, and manage VMs and Kubernetes clusters. Enable self-service orchestration of resources across clouds. Learn More.

spec

Description: IAConfigSpec defines the desired state of IAConfig

Type: object

Table 1. Description of IAConfig/v1alpha1 spec

Property	Type	Description

Property	Type	Description
spec	object	IAConfigSpec defines the desired state of IAConfig
spec.imagePullSecret	string	Select a secret to serve as the ImagePullSecret
spec.infraAutoComposableComponents	array	Enable Infrastructure management (im) and/or Manage services (cam).
spec.license	object	License acceptance is required to create resources.
spec.license.accept	boolean	I accept the product license after reading it at https://ibm.biz/cp4aiops-4100-license
spec.skipAutomationBaseStatus	boolean	This field is reserved for support person to use.
spec.skipAutomationUiConfigStatus	boolean	This field is reserved for support person to use.
spec.storageClass	string	Select a storage class that supports ReadWriteMany access mode.
spec.storageClassLargeBlock	string	Select a storage class with a large block size (for example, 64k).

Important: The following fields are deprecated and should not be used in Infrastructure Automation 4.10.0:

- skipAutomationBaseStatus
- skipAutomationUiConfigStatus

status

Description: IAConfigStatus defines the observed state of IAConfig

Type: object

Table 2. Description of IAConfig/v1alpha1 status

Property	Type	Description
status	object	IAConfigStatus defines the observed state of IAConfig
status.failureMessage	string	INSERT ADDITIONAL STATUS FIELD - define observed state of cluster Important: Run "make" to regenerate code after modifying this file
status.phase	string	InstallationPhase describes the phase of the overall Installation
status.storageclass	string	(No Description)
status.storageclasslargeblock	string	(No Description)
status.version	string	(No Description)

ManageService/v1alpha1

ManageService is the Schema for the manageservices API.

spec

Description: ManageServiceSpec defines the desired state of ManageService.

Type: object

Table 1. Description of IAManageService spec

Property	Type	Description	Default
camAPI	object	Configuration for Cloud Automation Manager API microservices	`{"camComposerApi":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"camComposerUi":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"camIaas":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"camMongo":{"resources":{"limits":{"cpu":"500m","memory":"4Gi"},"requests":{"cpu":"100m","memory":"2Gi"}}},"camOrchestration":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"camPortalUi":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"camProviderTerraformApi":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}},"terraformSSHDir":{"terraformSSHDirSecret":""}},"camSecret":{"secretName":"cam-api-secret"},"camServiceLibraryUi":{"resources":{"limits":{"cpu":"300m","memory":"512Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"camServiceLibraryUiAPi":{"resources":{"limits":{"cpu":"100m","memory":"512Mi"},"requests":{"cpu":"50m","memory":"256Mi"}}},"camTenantApi":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"camUiBasic":{"resources":{"limits":{"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}},"certificate":{"certName":"cert"},"replicaCount":1}`
camAPI.replicaCount	int	Number of replicas for cam microservices	`1`

Property	Type	Description	Default
camController.replicaCount	int	Number of replicas for Controller	`1`
camLoggingPolicies	object	Configuration for the logging level used with Cloud Automation Manager microservices	`{"logLevel":"info"}`
camLoggingPolicies.logLevel	string	Log Level for logging	`"info"`
camMongoPV	object	MongoDB Persistant Storage	`{"name":"cam-mongo-pv","persistence":{"accessMode":"ReadWriteOnce","enabled":true,"existingClaimName":"","size":"15Gi","storageClassName":"","useDynamicProvisioning":false}}`
camMongoPV.persistence.accessMode	string	Access Mode for Storage	`"ReadWriteOnce"`
camMongoPV.persistence.enabled	bool	Enable persistance	`true`
camMongoPV.persistence.existingClaimName	string	Specify the name of the Existing Claim to be used by your application, empty string means don't use an existClaim	`""`
camMongoPV.persistence.size	string	MongoDB PVC size (in Gi)	`"15Gi"`
camMongoPV.persistence.storageClassName	string	Specify the name of the StorageClass empty string means don't use a StorageClass	`""`
camMongoPV.persistence.useDynamicProvisioning	bool	useDynamicProvisioning for persistence	`false`
camProviderTerraform	object	Configuration for Cloud Automation Manager Provider Terraform	`{"isolateRuntime":false,"replicaCount":1,"runtime":{"replicaCount":0},"terraformSSHDir":{"terraformSSHDirSecret":""}}`

Property	Type	Description	Default
camProviderTerraform.isolateRuntime	bool	Select to enable isolation mode for Provider Terraform runtime. In this mode, the runtime runs in separate pods and have read only access to the cam-terraform-pv.	`false`
camProviderTerraform.replicaCount	int	Number of replicas for camProviderTerraformApi	`1`
camProviderTerraform.runtime	object	Configuration for Cloud Automation Manager Provider Terraform runtime	`{"replicaCount":0}`
camProviderTerraform.runtime.replicaCount	int	Number of replicas for Provider Terraform Runtime. This value is only applicable when camProviderTerraform.isolateRuntime is enabled	`0`
camProviderTerraform.terraformSSHDir	object	Cloud Automation Manager Provider Terraform SSH Directory	`{"terraformSSHDirSecret":""}`
camProviderTerraform.terraformSSHDir.terraformSSHDirSecret	string	Secret name for Cloud Automation Manager Provider Terraform SSH Directory Contents	`""`

Property	Type	Description	Default
camProxy	object	CAM Proxy configuration	`{"customNginxSSLCertificate":{"customNginxSSLCertName":""},"limits":{"cpu":"500m","memory":"1500Mi"},"replicaCount":1,"requests":{"cpu":"100m","memory":"256Mi"},"resources":null}`
camProxy.customNginxSSLCertificate.customNginxSSLCertName	string	Nginx SSL Custom Certificate Secret Name	`""`
camProxy.limits.cpu	string	The CPU limit	`"500m"`
camProxy.limits.memory	string	The memory limit	`"1500Mi"`
camProxy.replicaCount	int	Number of replicas for Proxy	`1`
camProxy.requests.cpu	string	The requested CPU	`"100m"`
camProxy.requests.memory	string	The requested memory	`"256Mi"`
camUI	object	UI Configuration	`{"camUISecret":{"secretName":"cam-ui-secret","sessionKey":"opsConsole.sid","sessionTTL":"43200"},"replicaCount":1}`
camUI.camUISecret	object	User Interface Secret	`{"secretName":"cam-ui-secret","sessionKey":"opsConsole.sid","sessionTTL":"43200"}`
camUI.camUISecret.secretName	string	User interface Secret Name	`"cam-ui-secret"`
camUI.camUISecret.sessionKey	string	Session Key	`"opsConsole.sid"`
camUI.camUISecret.sessionTTL	string	Session timeout in seconds	`"43200"`
camUI.replicaCount	int	Number of replicas for UI	`1`
database	object	Optionally use an external MongoDB	`{"bundled":true}`
database.bundled	bool	Use internal MongoDB. Set to false for external. For more information, see Using an external MongoDB in the IBM Knowledge Center	`true`
license	object	License acceptance	`{"accept":true}`

Property	Type	Description	Default
license.accept	bool	Accept the license to use the product"	`true`
proxy	object	Configuration for a proxy used with Cloud Automation Manager. Supply details in the secure values secret.	`{"useProxy":false}`
proxy.useProxy	bool	Select to use a proxy with CAM and fill out the fields in the Secure Values secret (see the Knowledge Center install instructions for more details)	`false`
resources	object	Amount of memory and CPU to assign to Cloud Automation Manager microservices	`{"limits": {"cpu":"500m","memory":"1500Mi"},"requests":{"cpu":"100m","memory":"256Mi"}}`
resources.limits.cpu	string	The memory limit	`"500m"`
resources.limits.memory	string	The CPU limit	`"1500Mi"`
resources.requests.cpu	string	The requested CPU	`"100m"`
resources.requests.memory	string	The requested memory	`"256Mi"`
roks	bool	Select to indicate it is Red Hat OpenShift on IBM Cloud System	`false`
roksRegion	string	Red Hat OpenShift on IBM Cloud Storage Region selection	`""`

Property	Type	Description	Default
roksZone	string	Red Hat OpenShift on IBM Cloud Storage Zone selection	""

IMInstall/v1alpha1

IMInstall is the Schema for the iminstalls API

spec

Description: IMInstallSpec defines the desired state of IMInstall

Type: object

Table 1. Description of IMInstall/v1alpha1 spec

Property	Type	Description
spec	object	IMInstallSpec defines the desired state of IMInstall
spec.appAnnotations	object	Optional Annotations to apply to the Httpd, Kafka, Memcached, Orchestrator and PostgresQL Pods
spec.appName	string	Application name used for deployed objects (default: iminstall)
spec.applicationDomain	string	Domain name for the external route. Used for external authentication configuration
spec.applicationImage	string	Image string used for the infrastructure management application deployment. By default it is based on the olm.relatedImage.application annotation
spec.backupLabelName	string	This label will be applied to essential resources that need to be backed up (default: manageiq.org/backup)
spec.baseWorkerImage	string	Image string used for the base worker deployments By default this is determined by the orchestrator pod
spec.databaseRegion	string	Database region number (default: 0)
spec.databaseSecret	string	Secret containing the database access information, content generated if not provided (default: postgresql-secrets)
spec.databaseVolumeCapacity	string	Database volume size (default: 50Gi)
spec.deployMessagingService	boolean	Deprecated: Flag to indicate if Kafka and Zookeeper should be deployed (default: false)
spec.enableApplicationLocalLogin	boolean	Flag to allow logging into the application without SSO (default: false)
spec.enableSSO	boolean	Flag to enable SSO in the application (default: false)
spec.enforceWorkerResourceConstraints	boolean	Flag to trigger worker resource constraint enforcement (default: true)

Property	Type	Description
spec.httpdAuthConfig	string	Secret containing the httpd configuration files Mutually exclusive with the OIDCClientSecret and OIDCProviderURL if using openid-connect
spec.httpdAuthenticationType	string	Type of httpd authentication (default: internal) Options: internal, external, active-directory, saml, openid-connect Note: external, active-directory, and saml require an httpd container with elevated privileges
spec.httpdCpuLimit	string	Httpd deployment CPU limit (default: 1000m)
spec.httpdCpuRequest	string	Httpd deployment CPU request (default: 50m)
spec.httpdImage	string	Image string used for the httpd deployment (default: os.Getenv("HTTPD_IMAGE"))
spec.httpdImageNamespace	string	Image namespace used for the httpd deployment (default: docker-na-public.artifactory.swg-devops.com/hyc-cp4mcm-team-docker-local/infra-mgmt) Note: the exact image will be determined by the authentication method selected
spec.httpdImageTag	string	Image tag used for the httpd deployment (default: 2-el9)
spec.httpdMemoryLimit	string	Httpd deployment memory limit (default: 200Mi)
spec.httpdMemoryRequest	string	Httpd deployment memory request (default: 100Mi)
spec.imagePullSecret	string	Secret containing the image registry authentication information needed for the manageiq images
spec.initialAdminGroupName	string	Group name to create with the super admin role. This can be used to seed a group when using external authentication
spec.internalCertificatesSecret	string	Secret containing all of the necessary certificates to secure communication between pods (default: internal-certificates-secret)
spec.kafkaCpuRequest	string	Kafka deployment CPU request (default: 250m)
spec.kafkaCpulimit	string	Kafka deployment CPU limit (default: 250m)
spec.kafkaImage	string	Image string used for the kafka deployment (default: os.Getenv("KAFKA_IMAGE"))
spec.kafkaImageName	string	Image used for the kafka deployment (default: docker.io/bitnami/kafka)
spec.kafkaImageTag	string	Image tag used for the kafka deployment (default: latest)
spec.kafkaMemoryLimit	string	Kafka deployment memory limit (default: 1024Mi)
spec.kafkaMemoryRequest	string	Kafka deployment memory request (default: 256Mi)
spec.kafkaSecret	string	Secret containing the kafka access information, content generated if not provided (default: kafka-secrets)
spec.kafkaVolumeCapacity	string	Kafka volume size (default: 1Gi)
spec.license	object	Opt-in license acceptance is required to create resources
spec.license.accept	boolean	Accept is an opt-in license acceptance required to deploy resources

Property	Type	Description
spec.memcachedCpuLimit	string	Memcached deployment CPU limit (default: 1000m)
spec.memcachedCpuRequest	string	Memcached deployment CPU request (default: 50m)
spec.memcachedImage	string	Image string used for the memcached deployment (default: :)
spec.memcachedImageName	string	Image used for the memcached deployment (default: manageiq/memcached)
spec.memcachedImageTag	string	Image tag used for the memcached deployment (default: 1-el9)
spec.memcachedMaxConnection	string	Memcached max simultaneous connections (default: 1024)
spec.memcachedMaxMemory	string	Memcached item memory in megabytes (default: 64)
spec.memcachedMemoryLimit	string	Memcached deployment memory limit (default: 200Mi)
spec.memcachedMemoryRequest	string	Memcached deployment memory request (default: 100Mi)
spec.memcachedSlabPageSize	string	Memcached max item size (default: 1m, min: 1k, max: 1024m)
spec.migrationsRan	array	A list of CR data migrations that have been run
spec.oidcAuthIntrospectionURL	string	URL for OIDC authentication introspection Only used with the openid-connect authentication type. If not specified, the operator will attempt to fetch its value from the "introspection_endpoint" field in the Provider metadata at the OIDCProviderURL provided.
spec.oidcCaCertSecret	string	Secret containing the trusted CA certificate file(s) for the OIDC server. Only used with the openid-connect authentication type
spec.oidcClientSecret	string	Secret name containing the OIDC client id and secret Only used with the openid-connect authentication type
spec.oidcProviderURL	string	URL for the OIDC provider Only used with the openid-connect authentication type
spec.orchestratorCpuLimit	string	Orchestrator deployment CPU limit (default: 1000m)
spec.orchestratorCpuRequest	string	Orchestrator deployment CPU request (default: 200m)
spec.orchestratorImage	string	Image string used for the orchestrator deployment (default: / :)
spec.orchestratorImageName	string	Image name used for the orchestrator deployment (default: infra-mgmt-orchestrator)
spec.orchestratorImageNamespace	string	Image namespace used for the orchestrator and worker deployments (default: manageiq)
spec.orchestratorImageTag	string	Image tag used for the orchestrator and worker deployments (default: latest-18.y)

Property	Type	Description
spec.orchestratorInitialDelay	string	Number of seconds to wait before starting the orchestrator liveness check (default: 2400)
spec.orchestratorMemoryLimit	string	Orchestrator deployment memory limit (default: 2048Mi)
spec.orchestratorMemoryRequest	string	Orchestrator deployment memory request (default: 1024Mi)
spec.postgresqlCpuLimit	string	PostgreSQL deployment CPU limit (default: 10000m)
spec.postgresqlCpuRequest	string	PostgreSQL deployment CPU request (default: 500m)
spec.postgresqlImage	string	Image string used for the postgresql deployment (default: :)
spec.postgresqlImageName	string	Image used for the postgresql deployment (Default: docker.io/manageiq/postgresql)
spec.postgresqlImageTag	string	Image tag used for the postgresql deployment (Default: 13-el9)
spec.postgresqlMaxConnections	string	PostgreSQL maximum connection setting (default: 1000)
spec.postgresqlMemoryLimit	string	PostgreSQL deployment memory limit (default: 8192Mi)
spec.postgresqlMemoryRequest	string	PostgreSQL deployment memory request (default: 2048Mi)
spec.postgresqlSharedBuffers	string	PostgreSQL shared buffers setting (default: 1GB)
spec.serverGuid	string	Server GUID (default: auto-generated)
spec.storageClassName	string	StorageClass name that will be used by manageiq data stores
spec.tlsSecret	string	Secret containing the tls cert and key for the ingress, content generated if not provided (default: tls-secret)
spec.uiWorkerImage	string	Image string used for the UI worker deployments By default this is determined by the orchestrator pod
spec.opentofuRunnerImage	string	Image string used for the Opentofu runner deployments By default this is determined by the orchestrator pod
spec.webserverWorkerImage	string	Image string used for the webserver worker deployments By default this is determined by the orchestrator pod
spec.zookeeperCpuRequest	string	Zookeeper deployment CPU request (default: 250m)
spec.zookeeperCpulimit	string	Zookeeper deployment CPU limit (default: 250m)
spec.zookeeperImage	string	Image string used for the zookeeper deployment (default: :)
spec.zookeeperImageName	string	Image used for the zookeeper deployment (default: docker.io/bitnami/zookeeper)
spec.zookeeperImageTag	string	Image tag used for the zookeeper deployment (default: latest)
spec.zookeeperMemoryLimit	string	Zookeeper deployment memory limit (default: 256Mi)

Property	Type	Description
spec.zookeeper MemoryRequest	string	Zookeeper deployment memory request (default: 256Mi)
spec.zookeeper VolumeCapacity	string	Zookeeper volume size (default: 1Gi)

status

Description: IMInstallStatus defines the observed state of IMInstall

Type: object

Table 1. Description of IMInstall/v1alpha1 status

Property	Type	Description
status	object	IMInstallStatus defines the observed state of IMInstall

Accessing APIs

You can access and use IBM Cloud Pak® for AIOps APIs after you obtain an access token for authentication.

About this task

To use any IBM Cloud Pak for AIOps API, you must first authenticate your credentials by including an access token in your API calls. This token is required because a password-based authentication approach is not supported for API calls with IBM Cloud Pak for AIOps, for instance, you cannot call an LDAP server to directly authenticate passwords.

The token that is generated is scoped to your user profile and the permissions that you are assigned when the access token is generated. With your access token, you can use API to access everything that you would typically be able to access when you log in to the IBM Cloud Pak for AIOps console.

To get a Platform UI access token, first you must have either an Identity and Access Management (IAM) access token or an API key that is authenticated by your user credentials. Then, you can use the IAM access token or API key to obtain your Platform UI access token. This Platform UI access token must be included in the authorization header of your calls to IBM Cloud Pak for AIOps API.

Before you create your token, run the following commands as per your deployment:

1. Set the variable for project (namespace). Example is shown in the following command:

   ```
   NAMESPACE=cp4aiops
   ```

2. If you have an existing installation of Cloud Pak for AIOps on Linux, run the following command:

   ```
   ROUTE=$(oc get ingress zen-ingress -n $NAMESPACE -o
   jsonpath='{.spec.rules[0].host}')
   ```

3. If you have an existing installation of Cloud Pak for AIOps on OpenShift Container Platform, run the following command:

   ```
   ROUTE=$(oc get route cpd -n $NAMESPACE --no-headers | awk '{print $2}')
   ```

Note: To call the API end points, you need to use the **`$ROUTE`** variable in multiple API end points. You can use the **`echo $ROUTE`** command to view the value for the route variable.

To create your token, complete either of the following procedures based on your preferred method:

- Get access token by using your API key
- Get access token by using your username and password
- Get authentication by using Platform UI API Key

Important: You can directly access the Swagger UI for only the Topology API and Metric API.

- For more information about accessing Metric service Swagger UI, see Accessing Metric service Swagger UI.
- For more information about accessing topology service Swagger UI, see Accessing Topology service Swagger UI.

Get access token by using your API key

1. Generate your platform API key:
    1. Log in to the IBM Cloud Pak for AIOps console to authenticate your user credentials.

       For more information, such as for how to obtain the URL route for accessing the Cloud Pak for AIOps console, see Accessing the Cloud Pak for AIOps console.
    2. From the toolbar, click your avatar.
    3. Click **Profile and settings**.
    4. Click **API key** > **Generate new key**.
    5. Click **Generate**.
    6. Click **Copy** and save your key somewhere safe. You cannot recover this key if you lose it.

    If you lose your API key, repeat the preceding steps to generate a new API key. Your old API key becomes invalid, and any applications or scripts cannot authenticate to the platform until you provide your new API key. For more information, see Generating API keys for authentication.
2. Get your Platform UI access token by running the following cURL command:

```
curl -k -X POST \
  https://$ROUTE/icp4d-api/v1/authorize \
  -H 'Content-Type: application/json' \
  -d '{
    "username": "<username>",
    "api_key": "<API_key>"
  }'
```

    - **`ROUTE`** - The IBM Cloud Pak platform (**`cpd`**) route hostname.
    - **`<username>`** - Your username.
    - **`<API_key>`** - The API key that you generated in the previous step.

    The API response can resemble the following sample:

```
{
    "_messageCode_": "200",
    "message": "Success",
```

```
    "token": "<encoded-token-string>"
}
```

- `<encoded-token-string>` - Your Platform UI access token.

Get access token by using your username and password

1. Get your Identity and Access Management (IAM) access token with the following cURL command:

```
curl -k -X POST -H "Content-Type: application/x-www-form-
urlencoded;charset=UTF-8" \
-d "grant_type=password&username=<username>&password=<password>&scope=openid"
\
https://<cluster_address>/idprovider/v1/auth/identitytoken
```

Where:

- `<username>` - Your username.
- `<password>` - Your password.
- `<cluster_address>` - The IBM Cloud Pak for AIOps console route. For more information about how to get this Cloud Pak for AIOps console route, see Accessing the Cloud Pak for AIOps console.

The command returns an **access_token**, **refresh_token**, and **id_token**, as shown in the following example:

```
{"access_token":
"eb837eaf32459b711945a9d2259880119056e805ff0d2f36421cc171f94e58fef349f0005d21
7e36889b62271d9f00fc7cb5b1fe5a86546d9dee8e22bca39b8f90d6cb61dae7fc383447823a0
9e380feeefba5bea0c994408470a49db0df32ddc2b0cca9381519e60a63daae9b87ebfe9400b0
c4af818b7f7d6c32e21465909efc8aa02804808f23ff96ac342b3b1c35230ac8858dbcb797999
5d7044c7b9cb05945c91b63a938703641e0fded339fb4c22e2383743a94a30c41892804193744
e0c0f020909f9579555bf691b240fafb558f76877fe0cb88ecbb3266fedbc7c541129270f6778
4d11ed658998b536841e0fdfc50a9ad056d2cabf717cb13326e4f620a6ce172d8da4701b820c5
ffe23223e7fb5725b244a1dd45538a0c7ca09a643759aaa2d8585a28689cae968ab3328351e3c
38a8b199040067ca5837169ce62a88282d1c8551d762fbdff77727cb51dd62213cef58dfb88e3
04abbc48063246b7e9f39650a0ac86f6c72973b702b79faf34b68afb9412c9e0e56b104e12bf1
ea3764faeac258fe1c3e896da412607a71ad8b4224efcfa0eafbc15f5e7af5b8baa41163c2204
19c7249e9652ca7b5692b42cbe4c7d88c18d77440ec350582f51880e7354eed76ebd8ce760b27
a6ca5808c6fc51ef843ee5d98e5bbafaeb4096301853fa5fdc876275defd3ecabd0eb656c7cd2
441e523e0c5468a1f261fb44",
"token_type":"Bearer","expires_in":43199,"scope":"openid"
"refresh_token":"6q4griAg9yCiGINQvF0Dp7N9hqXhcXZrAsqWWYgl6XQ80Uexsq",
"id_token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdF9oYXNoIjoiYWRmZDc4MmEw
OTc1ZTNmMzc2ZTkxZTI3YjJkNTYxZmQ0OTNiNTQzMSIsInJlYWxtTmFtZSI6ImN1c3RvbVJlYWxtI
iwidW5pcXVlU2VjdXJpdHlOYW1lIjoiYWRtaW4iLCJpc3MiOiJodHRwczovL215Y2x1c3Rlci5pY3
A6OTQ0My9vaWRjL2VuZHBvaW50L09QIiwiYXVkIjoiMGQzYzA3MTc5OTYxYmEzMWEyODY5NDU0NDQ
wM2E0NDYiLCJleHAiOjE1NTQ5MTQ2NTIsImlhdCI6MTU1NDg4NTg1Miwic3ViIjoiYWRtaW4iLCJ0
ZWFtUm9sZU1hcHBpbmdzIjpbXX0.CnT0qWECpJR9R16W-
IOqrXjSJR8DelRsDUXcX6hy_I0DPQ7hU55Bhcq6UChEg3qiWWRbKwrFIxikXPjEjw2B9oziEd8U8A
EO-4LEaXOpc5Lk1shvyxBQFDDgyUwgyGb-
erRbO_Sl1K4xotuTLg4nhoydwTXs7lZn7GC4UW8j1qkhlbFe5iLgKidCZsjyPo-
2GNYEQn0ufHH3KCR4DkHi6GX2RUxisNecwDzNl9P5JSyjlS-
r5QUZJ0b0DytKuY5HxpswpIFaO9U8JlYAFoOZ18eO_CzERHRQ_Ii1ePmagGAk-
eLJjmCNqY1zynfpEUuKlWUR5rVGHGzSbGA8J4CLvg"}
```

From this example, the following content is the **access token** that is needed for obtaining a platform UI access token:

```
"access_token":
"eb837eaf32459b711945a9d2259880119056e805ff0d2f36421cc171f94e58fef349f0005d21
```

```
7e36889b62271d9f00fc7cb5b1fe5a86546d9dee8e22bca39b8f90d6cb61dae7fc383447823a0
9e380feeefba5bea0c994408470a49db0df32ddc2b0cca9381519e60a63daae9b87ebfe9400b0
c4af818b7f7d6c32e21465909efc8aa02804808f23ff96ac342b3b1c35230ac8858dbcb797999
5d7044c7b9cb05945c91b63a938703641e0fded339fb4c22e2383743a94a30c41892804193744
e0c0f020909f9579555bf691b240fafb558f76877fe0cb88ecbb3266fedbc7c541129270f6778
4d11ed658998b536841e0fdfc50a9ad056d2cabf717cb13326e4f620a6ce172d8da4701b820c5
ffe23223e7fb5725b244a1dd45538a0c7ca09a643759aaa2d8585a28689cae968ab3328351e3c
38a8b199040067ca5837169ce62a88282d1c8551d762fbdff77727cb51dd62213cef58dfb88e3
04abbc48063246b7e9f39650a0ac86f6c72973b702b79faf34b68afb9412c9e0e56b104e12bf1
ea3764faeac258fe1c3e896da412607a71ad8b4224efcfa0eafbc15f5e7af5b8baa41163c2204
19c7249e9652ca7b5692b42cbe4c7d88c18d77440ec350582f51880e7354eed76ebd8ce760b27
a6ca5808c6fc51ef843ee5d98e5bbafaeb4096301853fa5fdc876275defd3ecabd0eb656c7cd2
441e523e0c5468a1f261fb44"
```

2. Get your Platform UI access token by running the following cURL command:

```
curl -k X GET 'https://$ROUTE/v1/preauth/validateAuth' \
-H 'username: admin' \
-H 'iam-token: <iam-token>'
```

   - `ROUTE` - The IBM Cloud Pak platform (`cpd`) route hostname.
   - `<username>` - Your username.
   - `<iam-token>` - The IAM access token that you obtained from the response in the previous step.

   The API response can resemble the following sample:

```
{
    "_messageCode_": "200",
    "message": "Success",
    "token": "<encoded-token-string>"
}
```

   - `<encoded-token-string>` - Your Platform UI access token.

Get authentication by using Platform UI API Key

1. Generate your platform API (Zen) key:

   1. Log in to the IBM Cloud Pak for AIOps console to authenticate your user credentials.

      For more information, such as for how to obtain the URL route for accessing the Cloud Pak for AIOps console, see Accessing the Cloud Pak for AIOps console.

   2. From the toolbar, click your avatar.
   3. Click **Profile and settings**.
   4. Click **API key** > **Generate new key**.
   5. Click **Generate**.
   6. Click **Copy** and save your key somewhere safe. You cannot recover this key if you lose it.

   If you lose your API key, repeat the preceding steps to generate a new API key. Your old API key becomes invalid, and any applications or scripts cannot authenticate to the platform until you provide your new API key. For more information, see Generating API keys for authentication.

2. Generate a base64 format of your Platform UI API Key:

```
ZENAPIKEY=$(echo "<username>:<API_key>" | base64 -w 0)
```

- `<username>` - Your username.
- `<API_key>` - The API key that you generated in the previous step.

3. The output is an encoded value. See the following example.

```
YWRtaW46MUFKblRIYngxNFJUVnBoR0FTdW0wcDJEOFl6UXFyVllmU2tmeW5qYgo=
```

   This encoded (base64 format) Platform UI API Key is used as the value for `<ZenApiKey>`.

4. Construct your Authorization header with the Platform UI API Key.

```
--header "Authorization: ZenApiKey <ZenApiKey>"
```

   - `<ZenApiKey>` - The Platform UI API key that you get in the previous step.

5. Run the following cURL command:

```
curl "https://<Endpoint URL>" --header 'Content-Type: application/json' --
header "Authorization: ZenApiKey <ZenApiKey>" --header 'X-TenantID: cfd95b7e-
3bc7-4006-a4a8-a73a79c71255' --insecure
```

   - `<Endpoint URL>` - The Endpoint URL as specified on the API page of the service that you want to use, for example, you can find the Endpoint URL for metrics here.

Next steps

Access IBM Cloud Pak for AIOps API with your Platform UI authentication token or API keys. Include the Platform UI access token in the authorization header of your API calls.

For more information on the available API and related Swagger documentation, see APIs.

Generating API keys for authentication

With an API key, you can automatically authenticate to the platform or to a specific instance of a service from a script or application. Your API keys are associated with your credentials and are specific to you. With API keys, you can authenticate without entering your username and password.

You can generate:

- A platform API key

Your platform API key enables scripts and applications to access everything that you would typically be able to access when you log in to the console.

Before you begin

Before you generate an API key, consider the following factors:

What level of access is required?

As a security best practice, it is recommended that you give the least amount of access necessary to the script or application. For example, if a script needs to interact only with a specific service instance, generate an API key for that service instance.

Important: Be sure to store your API keys somewhere safe. If you lose an API key, you'll need to create a new key and update any scripts or applications where the key is used.

Platform API key

With a platform API key, you can access everything that you would typically be able to access when you log in to the console.

To generate a platform API key:

1. Log in to the console.
2. From the toolbar, click your avatar.
3. Click **Profile and settings**.
4. Click **API key** > **Generate new key**.
5. Click **Generate**.
6. Click **Copy** and save your key somewhere safe. You cannot recover this key if you lose it.

If you lose your API key, repeat the preceding steps to generate a new API key. Your old API key becomes invalid, and any applications or scripts cannot authenticate to the platform until you provide your new API key.

If you believe that your API key is compromised:

1. Log in to the console.
2. From the toolbar, click your avatar.
3. Click **Profile and settings**.
4. Click **API key** > **Revoke current key**.
5. Click **Revoke**.

Any applications or scripts that use the key won't be able to authenticate to the platform.

Platform UI APIs

Learn more about the Platform UI APIs that are available for you to use to manage users and the personalization of the console.

User management APIs

Create and manage users and access to the console with the user management API.

Custom cards API

Create custom cards to display information on the console home page with the Custom cards REST API.

Creating custom home page cards with API

Create custom cards to display information on the console home page with the Custom cards REST API.

For example, if you want to show disk usage as a percentage of available disk space, you can add a custom card to the home page that displays the data in a donut or bar chart.

You can use the Custom cards REST API to complete the following tasks:

- Generate an authorization token
- Get a list of custom cards
- Create or replace a custom card
- Edit a custom card
- Delete a custom card

You cannot use the Custom cards API to replace, edit, or delete existing cards that are added by default.

Sample custom cards

You can start with a template to create each of the custom cards that are shown in the following image. Review the template types for a description of the templates that are used.

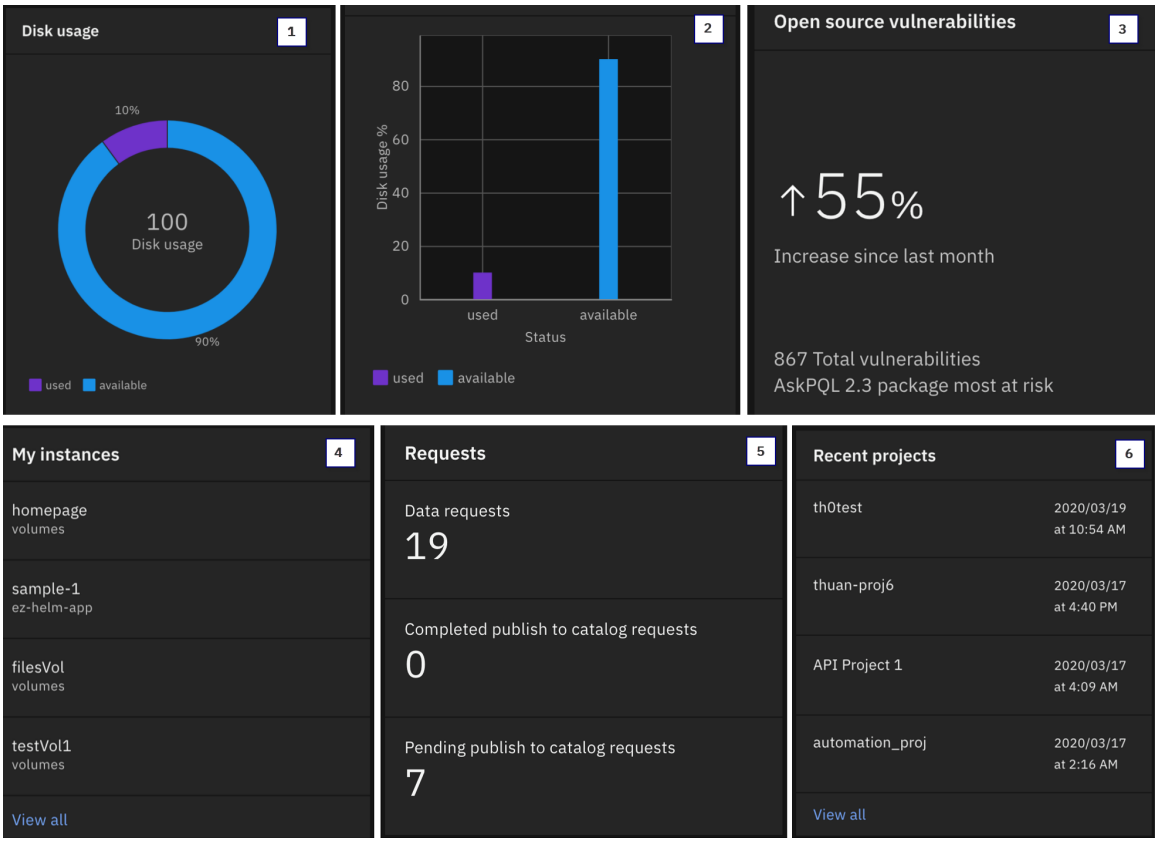


Figure 1. Custom cards types

Template types:

1. The `donut` template illustrates an array of values as percentages in a donut chart.
2. The `bar` template creates a two-dimensional bar graph.
3. The `big_number` template highlights significant data.
4. The `text_list` displays a list of text strings.
5. The `number_list` displays a list of numeric values and associated text.
6. The `list` displays a list of navigation links. The list can contain headers and associated clickable links.

For the JSON payload that describes these custom cards, see the sample command in Create or replace a custom card.

Generate an authorization token

To use the Custom cards REST API, you must first authenticate. Use the token that you get in the response as a token in subsequent API calls.

1. Get an Identity Management (IM) access token:

```
curl -k -X POST -H "Content-Type: application/x-www-form-
urlencoded;charset=UTF-8" \
-d "grant_type=password&username=<username>&password=<password>&scope=openid"
\
https://<cluster_address>/idprovider/v1/auth/identitytoken
```

   - `<username>` is your username.
   - `<password>` is your password.
   - `<cluster_address>` is the IBM Cloud Pak console route. You can get the IBM Cloud Pak console route by running the following command:

```
oc get route -n <your-foundational-services-namespace> cp-console -o
jsonpath='{.spec.host}'
```

   Following is a sample output:

```
cp-console.apps.mycluster.mydomain.com
```

   Based on the example output, your console URL would be `https://cp-console.apps.mycluster.mydomain.com`.

   The username and password are defined in the `config.yaml` file.

2. Use the IM access token to call the Platform UI token API:

```
curl -k X GET 'https://<master_node_hostname>/v1/preauth/validateAuth' \
-H 'username: admin' \
-H 'iam-token: <iam-token>'
```

   - `<master_node_hostname>` the hostname for the master node.
   - `<iam-token>` is the IM token that you obtained from the response in the previous step.

Get a list of custom cards

Retrieves the properties for the custom cards that you can update.

Curl

```
GET /zen-data/v1/custom_cards
```

Sample command

```
curl -kiv -X GET \
  https://my-deployment-url/zen-data/v1/custom_cards \
  -H 'Authorization: Bearer authorization-token' \
  -H 'Content-Type: application/json'
```

Sample response

In the following code block, the body of the response is replaced with `....`

```
"_messageCode_": "success",
    "message": "Successfully retrieved custom cards",
    "requestObj": {
```

```
      ...
}
```

Create or replace a custom card

Creates a custom card or replaces an existing custom card.

You must include the card definition as a JSON string in the HTTP request body. The definition specifies the structure and the content of the card.

Table 1. Card parameters

Parameter	Type	Description
`key`	`String`	The name of the card that you want to create or replace. The string must be unique. If a custom card with the specified key does not exist, the custom card is created, otherwise the existing custom card is replaced.
`data`	`object`	Provides data to the card. You must provide either a `data_url` object or a `data` object. The `data` object is an object that populates data in the card. The properties of the object must match the type of template defined. The expected structure for each `template_type` parameter is in following examples.
`data_url`	`endpoint`	Provides data to the card. You must provide either a `data_url` object or a `data` object. The `data_url` object is an endpoint that is called to populate data in the card. The response that is sent by the endpoint matches the type of template defined. The expected response structure for each `template_type` parameter is shown in the following examples.
`description`	`String`	Helps users understand the purpose of the card. The string must be 200 characters or less.
`drilldown_url` (Optional)	`String`	Defines the URL that the user navigates to when they click **View details**. If this parameter is not defined, the card does not display a **View details** link.
`order`	`integer`	Defines the position of the card. The value must be in the range 1 - 50.
`permissions`	`array`	Specifies the permissions that a user needs to see the card. The array must have at least one permission. If the array has no `permissions` parameter, the card is shown to everyone.
`refresh_rate` (Optional)	`Integer`	The interval in seconds at which the content is refreshed by the source.
`template_type`	`enumerated_type`	Defines the display type of the card. The `template\_type` parameter must be one of the following values: - `donut` - `bar` - `big_number` - `text_list` - `number_list` - `list`
`title`	`String`	The display name for the card. The string must be 40 characters or less.

Parameter	Type	Description
`window_open_target` (Optional)	`string`	Specifies the target attribute or the name of the window. The following values are supported: - Empty or not provided - the URL gets loaded on the current page by default. - `_blank` - URL is loaded into a new window or tab every time. - `name` - The name of the window where the URL is loaded. The name does not specify the title of the new window.

Use the following examples to understand the `data` object.

****bar_data****

```
"data": {
        "bar_data": {
              x_axis_label: 'Packages',
              y_axis_label: 'Projects',
                "data_array": [
                  {
                        "group": "Qty",
                        "value": 65000
                    },
                    {
                        "group": "More",
                        "value": 29123
                    },
                    {
                        "group": "Sold",
                        "value": 35213
                    },
                    {
                        "group": "Restocking",
                        "value": 51213
                    },
                    {
                        "group": "Misc",
                        "value": 16932
                    }
                ]
        }
}
```

****big_number_data****

```
"data": {
       "big_number_data": {
              prefix: 'ArrowUp32',
              metric: '1.6',
              suffix: '%',
              sub_text: 'Since last month',
              footer_1: '867 Total vulnerabilities',
              footer_2: 'AskPQL 2.3 package most at risk'
       }
}
```

****donut_data****

```
"data": {
              "donut_data": {
                "center_label": "Disk usage",
                "data_array": [
```

```
                {
                  "group": "2V2N 9KYPM version 1",
                  "value": 20000
                },
                {
                  "group": "L22I P66EP L22I P66EP L22I P66EP",
                  "value": 65000
                },
                {
                  "group": "JQAI 2M4L1",
                  "value": 75000
                },
                {
                  "group": "J9DZ F37AP",
                  "value": 1200
                },
                {
                  "group": "YEL48 Q6XK YEL48",
                  "value": 10000
                },
                {
                  "group": "Misc",
                  "value": 25000
                }
              ]
            }
          }
```

list_data

```
"data": {
       list_data: {
              headers: [
                     'project_name',
                     'last_updated_time'
              ],
              rows: [
              {
                     'project_name': 'Improving customer online registration',
                     'last_updated_time': '2 min ago',
                     'nav_url': ' /projects/49753ad5-dcf5-4f94-913f-10acf78a4e71
              },
              {
                     'project_name': '2020 Q1_Increasing Search performance
and..',
                     'last_updated_time': '30 min ago',
                     'nav_url': ' /projects/49753ad5-dcf5-4f94-913f-abc
              },
              {
                     'project_name': '2019 Customer shoes sales analysis',
                     'last_updated_time': '1 hour ago',
                     'nav_url': ' /projects/49753ad5-dcf5-4f94-913f-def
              },
              {
                     'project_name': 'Project X Final version',
                     'last_updated_time': 'Today 11:04 PM',
                     'nav_url': ' /projects/49753ad5-dcf5-4f94-913f-qrs
              },
              {
                     'project_name': 'Project X Final version 2',
                     'last_updated_time': 'Yesterday 12:45 AM',
                     'nav_url': ' /projects/49753ad5-dcf5-4f94-913f-tuv
              }
              ]
```

```
        }
}
```

number_list_data

```
"data": {
       "number_list_data": {
              rows: [
              {
                     drilldown_url: '/zen/#/openSource/packages',
                     label: 'Total approved packages',
                     value: '3'
              },
              {
                     label: 'Total licenses',
                     value: '2345'
              },
              {
                     label: 'Total user interactions',
                     value: '25K'
              }
              ]
       }
}
```

text_list_data

```
"data": {
       "text_list_data": {
              rows: [
              {
                     drilldown_url: '/zen/#/openSource/packages',
                     label: 'Data virtualization',
                     sub_text: 'dv'
              },
              {
                     label: 'Watson assistant',
                     sub_text: 'assistant'
              },
              {
                     label: 'Watson openscale',
                     sub_text: 'aios'
              }
              ]
       }
}
```

Use the following examples to understand the **data_url** object. The format of the expected response that is provided in the **data_url** from the endpoint for each card type is shown in the following examples.

donut

```
{
      "center_label": "Disk usage",
      "data_array": [
         {
            "group": "2V2N 9KYPM version 1",
            "value": 20000
         },
         {
            "group": "L22I P66EP L22I P66EP L22I P66EP",
            "value": 65000
         },
         {
```

```
            "group": "JQAI 2M4L1",
            "value": 75000
         },
         {
            "group": "J9DZ F37AP",
            "value": 1200
         },
         {
            "group": "YEL48 Q6XK YEL48",
            "value": 10000
         },
         {
            "group": "Misc",
            "value": 25000
         }
      ]
}

**bar**

{
      "x_axis_label": "Packages",
      "y_axis_label": "Projects",
      "data_array": [
         {
            "group": "Australia",
            "value": 250
         },
         {
            "group": "Brazil",
            "value": 120
         },
         {
           "group": "Canada",
            "value": 190
         },
         {
            "group": "Germany",
            "value": 170
         }
      ]
}

**big\_number**

{
       prefix: 'ArrowUp32',
       metric: '1.6',
       suffix: '%',
       sub_text: 'Since last month',
       footer_1: '867 Total vulnerabilities',
       footer_2: 'AskPQL 2.3 package most at risk'
}

**text\_list**

{
      rows: [
         {
            label: 'Data virtualization',
            sub_text: 'dv'
         },
         {
            label: 'Watson assistant',
```

```
            sub_text: 'assistant'
         },
         {
            label: 'Watson openscale',
            sub_text: 'aios'
         }
      ]
}

**number\_list**

{
      rows: [
         {
            label: 'Total approved packages',
            value: '3'
         },
         {
            label: 'Total licenses',
            value: '2345'
         },
         {
            label: 'Total user interactions',
            value: '25K'
         }
      ]
}

**list**

{
       headers: [
              'project_name',
              'last_updated_time'
       ],
       rows: [
       {
              'project_name': 'Improving customer online registration',
              'last_updated_time': '2 min ago',
              'nav_url': ' /projects/49753ad5-dcf5-4f94-913f-10acf78a4e71
       },
       {
              'project_name': '2020 Q1_Increasing Search performance',
              'last_updated_time': '30 min ago',
              'nav_url': '/zen/#/mydata/datasets'
       },
       {
              'project_name': '2019 Customer shoes sales analysis',
              'last_updated_time': '1 hour ago',
              'nav_url': '/zen/#/mydata/datasets'
       },
       {
              'project_name': 'Project X Final version',
               'last_updated_time': 'Today 11:04 PM',
               'nav_url': '/zen/#/mydata/datasets'
       },
       {
              'project_name': 'Project X Final version 2',
              'last_updated_time': 'Yesterday 12:45 AM',
              'nav_url': '/zen/#/mydata/datasets'
       }
       ]
}
```

Curl

```
PUT /zen-data/v1/custom_cards/{key}
```

Sample command

```
curl -k -X PUT 'https://my-deployment-url/zen-data/v1/custom_cards/unique-card-
key' \
--header 'Authorization: Bearer authorization-token' \
--header 'Content-Type: application/json' \
--data-raw '{
  "permissions": ["manage_catalog"],
  "order": 1,
  "title": "Disk usage",
  "template_type": "big_number",
  "data": {      "big_number_data": {
        "metric":  "55",
        "sub_text": "Increase since last month",
        "prefix": "ArrowUp32",
        "suffix": "%",
        "footer_1": "867 Total vulnerabilities",
        "footer_2": "AskPQL 2.3 package most at risk"
      }
    }
}
'
```

Sample response

```
{
    "_messageCode_": "success",
    "data": [
        {
            "id": "537163840404062200",
            "name": "os_vulnerabilities",
            "title": "Open source vulnerabilities",
            "description": "",
            "drilldown_url": "",
            "order": 12,
            "template_type": "big_number",
            "data_url": "",
            "empty_state": {},
            "source_url": "",
            "refresh_rate": 0,
            "data": {
                "big_number_data": {
                    "metric": "55",
                    "sub_text": "Increase since last month",
                    "prefix": "ArrowUp32",
                    "suffix": "%",
                    "footer_1": "867 Total vulnerabilities",
                    "footer_2": "AskPQL 2.3 package most at risk"
                }
            }
        }
    ],
    "message": "PUT custom card succeeded"
}
```

Edit a custom card

Edit the information that is displayed on an existing card.

You can edit the following content:

- `data`
- `data\_url`
- `description`
- `drilldown\_url`
- `order`
- `permissions`
- `template\_type`
- `title`

To edit a custom card, use the rules that are specified in Create or replace a custom card.

Curl

```
PATCH /zen-data/v1/custom_cards/{key}
```

Path parameters

Table 2. Path parameters - patch

Parameter	Type	Description
key	`String`	The name of the card that you want to update.

Sample command

This command updates the title of a custom card called **`disk_usage_donut`**.

```
curl -kiv -X PATCH \
  https://my-deployment-url/zen-data/v1/custom_cards/disk_usage_donut \

  -H 'Authorization: Bearer authorization-token' \
  -d '{

  "title": "Disk usage"
}'
```

Sample response

```
{
    "_messageCode_": "success",
    "data": [
        {
            "id": "1",
            "name": "homepage_card_disk_usage",
            "title": "Disk usage",
            "description": "",
            "drilldown_url": "/zen/#/v1/diskUsage",
            "order": 2,
            "template_type": "donut",
            "data_url": "/zen-data/v1/homepage/diskUsage",
            "empty_state": {},
            "source_url": "",
            "refresh_rate": 0,
            "data": null
        }
    ],
```

```
    "message": "Successfully edited custom card"
}
```

Delete a custom card

Deletes an existing card with the specified key.

Curl

```
DELETE /zen-data/v1/custom_cards/{key}
```

Path parameters

Table 3. Path parameters - delete

Parameter	Type	Description
key	`String`	The name of the custom card that you want to delete.

Sample command

This command deletes the `os_vulnerabilities` card.

```
curl -kiv -X DELETE \
  https://my-deployment-url/zen-data/v1/custom_cards/os_vulnerabilities \
  -H 'Authorization: Bearer authorization-token'
```

Sample response

```
{
    "_messageCode_": "success",
    "message": "Successfully deleted custom card"
}
```

IM APIs

Identity management (IM) APIs.

- OIDC Registration APIs
- Identity Provider V3 APIs

Preparing to run API commands

Before you run API commands, retrieve the authentication token and download the CA certificate for your cluster.

Procedure

1. Set the project to the namespace where you deployed foundational services.

   ```
   oc project <your-foundational-services-namespace>
   ```

2. Retrieve the authentication tokens. You can run curl commands.

```
curl -k -X POST -H "Content-Type: application/x-www-form-
urlencoded;charset=UTF-8"  -d "grant_type=password&username=`oc get secret
platform-auth-idp-credentials -o json | jq -r .data.admin_username| base64 -
d`&password=`oc get secret platform-auth-idp-credentials -o json | jq -r
.data.admin_password | base64 -d`&scope=openid" https://`oc get routes cp-
console -o jsonpath='{.spec.host}'`/idprovider/v1/auth/identitytoken | jq
```

For the unified console, use CPD routes instead of the **cp-console** route as shown in the following example.

```
curl -k -X POST -H "Content-Type: application/x-www-form-
urlencoded;charset=UTF-8"  -d "grant_type=password&username=`oc get secret
platform-auth-idp-credentials -o json | jq -r .data.admin_username| base64 -
d`&password=`oc get secret platform-auth-idp-credentials -o json | jq -r
.data.admin_password | base64 -d`&scope=openid" https://`oc get routes cpd -o
jsonpath='{.spec.host}'`/idprovider/v1/auth/identitytoken | jq
```

The command returns an **access_token**, **refresh_token**, and **id_token**, as shown in the following example:


```
{"access_token":"eb837eaf32459b711945a9d2259880119056e805ff0d2f36421cc171f94e
58fef349f0005d217e36889b62271d9f00fc7cb5b1fe5a86546d9dee8e22bca39b8f90d6cb61d
ae7fc383447823a09e380feeefba5bea0c994408470a49db0df32ddc2b0cca9381519e60a63da
ae9b87ebfe9400b0c4af818b7f7d6c32e21465909efc8aa02804808f23ff96ac342b3b1c35230
ac8858dbcb7979995d7044c7b9cb05945c91b63a938703641e0fded339fb4c22e2383743a94a3
0c41892804193744e0c0f020909f9579555bf691b240fafb558f76877fe0cb88ecbb3266fedbc
7c541129270f67784d11ed658998b536841e0fdfc50a9ad056d2cabf717cb13326e4f620a6ce1
72d8da4701b820c5ffe23223e7fb5725b244a1dd45538a0c7ca09a643759aaa2d8585a28689ca
e968ab3328351e3c38a8b199040067ca5837169ce62a88282d1c8551d762fbdff77727cb51dd6
2213cef58dfb88e304abbc48063246b7e9f39650a0ac86f6c72973b702b79faf34b68afb9412c
9e0e56b104e12bf1ea3764faeac258fe1c3e896da412607a71ad8b4224efcfa0eafbc15f5e7af
5b8baa41163c220419c7249e9652ca7b5692b42cbe4c7d88c18d77440ec350582f51880e7354e
ed76ebd8ce760b27a6ca5808c6fc51ef843ee5d98e5bbafaeb4096301853fa5fdc876275defd3
ecabd0eb656c7cd2441e523e0c5468a1f261fb44","token_type":"Bearer","expires_in":
43199,"scope":"openid","refresh_token":"6q4griAg9yCiGINQvF0Dp7N9hqXhcXZrAsqWW
Ygl6XQ80Uexsq","id_token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdF9oYXNoI
joiYWRmZDc4MmEwOTc1ZTNmMzc2ZTkxZTI3YjJkNTYxZmQ0OTNiNTQzMSIsInJlYWxtTmFtZSI6Im
N1c3RvbVJlYWxtIiwidW5pcXVlU2VjdXJpdHlOYW1lIjoiYWRtaW4iLCJpc3MiOiJodHRwczovL21
5Y2x1c3Rlci5pY3A6OTQ0My9vaWRjL2VuZHBvaW50L09QIiwiYXVkIjoiMGQzYzA3MTc5OTYxYmEz
MWEyODY5NDU0NDQwM2E0NDYiLCJleHAiOjE1NTQ5MTQ2NTIsImlhdCI6MTU1NDg4NTg1Miwic3ViI
joiYWRtaW4iLCJ0ZWFtUm9sZU1hcHBpbmdzIjpbXX0.CnT0qWECpJR9R16W-
IOqrXjSJR8DelRsDUXcX6hy_I0DPQ7hU55Bhcq6UChEg3qiWWRbKwrFIxikXPjEjw2B9oziEd8U8A
EO-4LEaXOpc5Lk1shvyxBQFDDgyUwgyGb-
erRbO_Sl1K4xotuTLg4nhoydwTXs7lZn7GC4UW8j1qkhlbFe5iLgKidCZsjyPo-
2GNYEQn0ufHH3KCR4DkHi6GX2RUxisNecwDzNl9P5JSyjlS-
r5QUZJ0b0DytKuY5HxpswpIFaO9U8JlYAFoOZ18eO_CzERHRQ_Ii1ePmagGAk-
eLJjmCNqY1zynfpEUuKlWUR5rVGHGzSbGA8J4CLvg"}
```


From the example, following is the **access token**:


```
"access_token":
"eb837eaf32459b711945a9d2259880119056e805ff0d2f36421cc171f94e58fef349f0005d21
7e36889b62271d9f00fc7cb5b1fe5a86546d9dee8e22bca39b8f90d6cb61dae7fc383447823a0
9e380feeefba5bea0c994408470a49db0df32ddc2b0cca9381519e60a63daae9b87ebfe9400b0
c4af818b7f7d6c32e21465909efc8aa02804808f23ff96ac342b3b1c35230ac8858dbcb797999
5d7044c7b9cb05945c91b63a938703641e0fded339fb4c22e2383743a94a30c41892804193744
e0c0f020909f9579555bf691b240fafb558f76877fe0cb88ecbb3266fedbc7c541129270f6778
4d11ed658998b536841e0fdfc50a9ad056d2cabf717cb13326e4f620a6ce172d8da4701b820c5
ffe23223e7fb5725b244a1dd45538a0c7ca09a643759aaa2d8585a28689cae968ab3328351e3c
38a8b199040067ca5837169ce62a88282d1c8551d762fbdff77727cb51dd62213cef58dfb88e3
04abbc48063246b7e9f39650a0ac86f6c72973b702b79faf34b68afb9412c9e0e56b104e12bf1
```

```
ea3764faeac258fe1c3e896da412607a71ad8b4224efcfa0eafbc15f5e7af5b8baa41163c2204
19c7249e9652ca7b5692b42cbe4c7d88c18d77440ec350582f51880e7354eed76ebd8ce760b27
a6ca5808c6fc51ef843ee5d98e5bbafaeb4096301853fa5fdc876275defd3ecabd0eb656c7cd2
441e523e0c5468a1f261fb44"
```

From the example, following is the **refresh token**:

```
"refresh_token":"6q4griAg9yCiGINQvF0Dp7N9hqXhcXZrAsqWWYgl6XQ80Uexsq"
```

From the example, following is the **ID token**:

```
"id_token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdF9oYXNoIjoiYWRmZDc4MmEw
OTc1ZTNmMzc2ZTkxZTI3YjJkNTYxZmQ0OTNiNTQzMSIsInJlYWxtTmFtZSI6ImN1c3RvbVJlYWxtI
iwidW5pcXVlU2VjdXJpdHlOYW1lIjoiYWRtaW4iLCJpc3MiOiJodHRwczovL215Y2x1c3Rlci5pY3
A6OTQ0My9vaWRjL2VuZHBvaW50L09QIiwiYXVkIjoiMGQzYzA3MTc5OTYxYmEzMWEyODY5NDU0NDQ
wM2E0NDYiLCJleHAiOjE1NTQ5MTQ2NTIsImlhdCI6MTU1NDg4NTg1Miwic3ViIjoiYWRtaW4iLCJ0
ZWFtUm9sZU1hcHBpbmdzIjpbXX0.CnT0qWECpJR9R16W-
IOqrXjSJR8DelRsDUXcX6hy_I0DPQ7hU55Bhcq6UChEg3qiWWRbKwrFIxikXPjEjw2B9oziEd8U8A
EO-4LEaXOpc5Lk1shvyxBQFDDgyUwgyGb-
erRbO_Sl1K4xotuTLg4nhoydwTXs7lZn7GC4UW8j1qkhlbFe5iLgKidCZsjyPo-
2GNYEQn0ufHH3KCR4DkHi6GX2RUxisNecwDzNl9P5JSyjlS-
r5QUZJ0b0DytKuY5HxpswpIFaO9U8JlYAFoOZ18eO_CzERHRQ_Ii1ePmagGAk-
eLJjmCNqY1zynfpEUuKlWUR5rVGHGzSbGA8J4CLvg"
```

3. Store the authentication token in a variable. You can access IBM Cloud Pak® for AIOps APIs by specifying an authentication token in the request header. Run the following command, where **<ID token>** is the displayed ID token:

   ```
   export ID_TOKEN=<ID token>
   ```

4. Store the access token in a variable. Include the full contents of the access token, including the Bearer value. For example, from the access token in the Curl command output in step 1, you must include the token value from **"eb837e** to **1fb44"**. You can access IBM Cloud Pak for AIOps user management APIs by specifying the access token in the request header. Run the following command, where **<Access token>** is the following displayed access token:

   ```
   export ACCESS_TOKEN=<Access token>
   ```

5. Obtain a copy of the CA certificate for your cluster. If you can access the boot node, the CA certificate file is **/<installation_directory>/cluster/cfc-certs/root-ca/ca.crt**.

OIDC Registration APIs

APIs to manage authentication.

Base path: **https://<cluster_address>/idprovider/v1/auth/**

Note: The **/idauth** route is removed and not supported for client registration.

Client Registration API

Following is the curl command to register the API:

```
curl -i -k -X POST --header "Authorization: Bearer $ACCESS_TOKEN" \ --header
"Content-Type: application/json" \ --data "@platform-oidc-registration.json" \
https://<cluster_address>/idprovider/v1/auth/registration
```

You can also use the access token in the body:

```
curl -i -k -X POST -d "access_token=$ACCESS_TOKEN" \
https://<cluster_address>/idprovider/v1/auth/registration
```

The contents of the platform-oidc-regisration.json file are in the following example:

```
{
    "token_endpoint_auth_method": "client_secret_basic",
    "client_id": <WLP_CLIENT_ID>,
    "client_secret": <WLP_CLIENT_SECRET>,
    "scope": "openid profile email",
    "grant_types": ["authorization_code", "client_credentials", "implicit",
           "refresh_token", "urn:ietf:params:oauth:grant-type:jwt-bearer"],
    "response_types": ["code", "token", "id_token token"],
    "application_type": "web",
    "subject_type": "public",
    "post_logout_redirect_uris": ["https://<ICP_PROXY_IP>:
<PORT_WHERE_SERVICE_RUNS>"],
    "preauthorized_scope": "openid profile email general",
    "introspect_tokens": true,
    "trusted_uri_prefixes": ["https://<ICP_ENDPOINT>:8443",
"https://<ICP_PROXY_IP>"],
    "redirect_uris": ["https://<ICP_PROXY_IP>:
<PORT_WHERE_SERVICE_RUNS>/auth/liberty/callback"],
}
```

The **WLP_CLIENT_ID** and **WLP_CLIENT_SECRET** must be generated by the user who is trying to register the client and the values must be unique.

Note: Calling **/Userinfo** does not work for the tokens with **grant_type** as **client_credentials**. Use **/instrospect** endpoints instead.

Delete the client ID

1. Export these variables:

   ```
   export CLIENT_ID=<client_id>
   ```

2. Run the curl command to delete the ID.

   ```
   curl -i -k -X DELETE --header "Authorization: Bearer $ACCESS_TOKEN"\
   https://icp-cluster-ip:8443/idprovider/v1/auth/registration/$CLIENT_ID
   ```

Call the authorization endpoint to display the login page

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/authorize

Command

GET

Command output format

application/json

The sample command resembles the following code:

```
GET https://<cluster_address>/idprovider/v1/auth/authorize?
client_id=$oauth_client_id&redirect_uri=https://$http_host/auth/liberty/callback&r
esponse_type=code&scope=openid+email+profile&state=$request_uri;
```

Get access token by using username and password

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/identitytoken

Command

POST

Command output format

application/json

Following is the curl command:

```
curl -k -X POST -H "Content-Type: application/x-www-form-urlencoded;charset=UTF-8"
\
-d "grant_type=password&client_id=<client_ID>&client_secret=
<client_secret>&username=<username>&password=<password>&scope=openid" \
https://<cluster_address>/idprovider/v1/auth/identitytoken --insecure
```

The sample command and response resembles the following code:

```
curl -k -X POST -H "Content-Type: application/x-www-form-urlencoded;charset=UTF-8"
\
```

```
-d "grant_type=password&client_id=<client_ID>&client_secret=
<client_secret>&username=admin&password=admin&scope=openid" \
https://9.37.239.32/idprovider/v1/auth/identitytoken --insecure

{
  "access_token":
"38400d87f39a7c328a4605265eb601bebd9426e2ef6f1b51a449da6a9cb08e03543857ac4ffbd7d2c
259867c89324563c5a89f026683aca9262858ae7ffb1e635242eabab3d579793e8f9da09070708dccf
2a8d660f3be06550f02af681d2fa64562fb9dc3df1b19839a5d3933311f89348634fa6908fa7d2d505
84ffd36f9dc298a3411d3f5abad5c7f45283428ecf0de249eac5534136c31317493f85363126bfe9a6
f582403c34a3dab96e3e7bba83c263f1a4ff8d8609fca888852e097e3bc382b822576a53e55e6753c5
7f79d5703cf6b6bed4b015702ce3ce1636fd834944231fa77eb90079bca398be511f22fd58792a3766
a100af10f274e6b9d75a2be2fe6ab18a3ce2ed0c8da7542e0b79f08e32a9ddced6a389572e6247230e
1b62adf5fb0ee6549c06f99b85afc7cccd7a51012dea5df40fc27a934be37e9465ddb46a4f43ec542f
aecb4e6dd062189392b802b8a0ad8c38a00a14f7b9625bcdab251b87cd478c0e5d3c79f8887797da76
7f209f5fb2b3d44c8b051f49c2ed680a14cd15388b545ca573540184bb27be28378dbe0ecbe2d71c0a
c3d7365fce5f1948ead1576f444f70c87d7ba89352b0d2b795a11ccc5ad06441c4143a3e78f80316c7
2110ba7062159f249719c664818befd6514b1526498729fe624852128495a5fa9c57ba8c9386a1040e
0bb8013e93a751722de6e85966994cefce4c43066",
  "token_type": "Bearer",
  "expires_in": 43199,
  "scope": "openid",
  "refresh_token": "ryJlHRTJu0ZWgpDm9Ci11YenaPUk2ehZ51p1gAmL2w5VAThuff",
  "id_token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdF9oYXNoIjoiNjZrYjBqMTY1NDBuN3ZhZXczem4i
LCJyZWFsbU5hbWUiOiJjdXN0b21SZWFsbSIsInVuaXF1ZVNlY3VyaXR5TmFtZSI6ImFkbWluIiwiaXNzIj
oiaHR0cHM6Ly9teWNsdXN0ZXIuaWNwOjk0NDMvb2lkYy9lbmRwb2ludC9PUCIsImF1ZCI6IjZhNTVlMWEz
ZmY1Mjc5NjY2YTBiNmI4NzcxYTViMzEwIiwiZXhwIjoxNTI5MzQzOTM4LCJpYXQiOjE1MjkzMTUxMzgsIn
N1YiI6ImFkbWluIiwidGVhbVJvbGVNYXBwaW5ncyI6W119.OHZTG7I5SjTk3uHIJsk7zzg5ueQM5fEU9nC
11jSvpRw-tm1T-OBqjKHPQ_g-uhmFuuym3hvQcEB-
wRQi4NMB_d580eeXHYYl_NiawunkHIl7AISQQetc7HS4U7ZXx3Mc2EmvqyVyo0zSYowGfT6D_X36O_E6Ri
z-
_rrGvc1nrzOdGa8IjJIi_GncSs5IFNUQxtRA9ZwdtIbQcRrSs9B3hPH8sJqUnaZnOjAkctJA8zQY0eV3IA
Z4lFc01_hT5DrOdtAiSAQBoakttxbY8iqEaNHAc07wUiN6J4rcgtJE2ZwOZth1D_39KyD5nbRbNO8HJh6h
YFcBplFGwp9FDZb27A"
}
```

Note: All the idprovider APIs are rate limited. If rate limit exceeds, it throws **429 error**. Standard rate limits: 100 requests per 10 seconds and burst of 20.

Get access token by using client_credentials

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/token

Command

POST

Command output format

application/json

Following is the curl command:

```
curl -k -X POST -H "Content-Type: application/x-www-form-urlencoded;charset=UTF-8"
\
-d "grant_type=client_credentials&client_id=<oidc_client_ID>&client_secret=
<oidc_client_secret>&scope=openid" \
https://<cluster_address>/idprovider/v1/auth/token --insecure
```

Note: You can also use `https://<cluster_address>/idprovider/v1/auth/identitytoken`

The sample response resembles the following code:

```
{
  "access_token":
"38400d87f39a7c328a4605265eb601bebd9426e2ef6f1b51a449da6a9cb08e03543857ac4ffbd7d2c
259867c89324563c5a89f026683aca9262858ae7ffb1e635242eabab3d579793e8f9da09070708dccf
2a8d660f3be06550f02af681d2fa64562fb9dc3df1b19839a5d3933311f89348634fa6908fa7d2d505
84ffd36f9dc298a3411d3f5abad5c7f45283428ecf0de249eac5534136c31317493f85363126bfe9a6
f582403c34a3dab96e3e7bba83c263f1a4ff8d8609fca888852e097e3bc382b822576a53e55e6753c5
7f79d5703cf6b6bed4b015702ce3ce1636fd834944231fa77eb90079bca398be511f22fd58792a3766
a100af10f274e6b9d75a2be2fe6ab18a3ce2ed0c8da7542e0b79f08e32a9ddced6a389572e6247230e
1b62adf5fb0ee6549c06f99b85afc7cccd7a51012dea5df40fc27a934be37e9465ddb46a4f43ec542f
aecb4e6dd062189392b802b8a0ad8c38a00a14f7b9625bcdab251b87cd478c0e5d3c79f8887797da76
7f209f5fb2b3d44c8b051f49c2ed680a14cd15388b545ca573540184bb27be28378dbe0ecbe2d71c0a
c3d7365fce5f1948ead1576f444f70c87d7ba89352b0d2b795a11ccc5ad06441c4143a3e78f80316c7
2110ba7062159f249719c664818befd6514b1526498729fe624852128495a5fa9c57ba8c9386a1040e
0bb8013e93a751722de6e85966994cefce4c43066",
  "token_type": "Bearer",
  "expires_in": 43199,
  "scope": "openid",
}
```

Get access token by using cpclient_credentials

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/token

Command

POST

Command output format

application/json

Following is the curl command:

```
curl -k -X POST -H "Content-Type: application/x-www-form-urlencoded;charset=UTF-8"
\
-d "grant_type=cpclient_credentials&client_id=<client_ID>&client_secret=
<client_secret>&scope=openid" \
https://<cluster_address>/idprovider/v1/auth/token --insecure
```

The sample response resembles the following code:


```
{
  "access_token":
"3b17b831c5f21e46c64718d26fe415ce2fc2f5482121ab8f213a28db8f3838c6a6cda5e3a9a8d38be
359cb3c8351655e7fac2af40a805fb42e701d0ad2a28ddf92421a282165d3d1ba7b2dc56c163e907c1
ba9662b2b5ea30eb29bf28b6d999bb73deff665c12320ee22aacbedd13ba86d53467887cfee9d7962f
6968906d2ab8f7b716a521cfbdc9b2820c424c1609f933a5240b0c637c6e4ef76007abbe6dd6e0f41e
5d4f2867ca9afe6efb3160d9573b4d18ef02eb044c86e3faf0f9095a7989e1bb756f2e862781e8600b
50854f648a3cde34c9359a26596c42096fc2d30d6742885d8c5d5eb4ac9a81ff130f255d304e4b6f63
041983ac5897a2941dcb9e69d1537c176c94854b70e761bfbefcf77aba5fff3f397fcddce3f317e49f
148740d659d807b51f52757c5c5a0fd23fcf8bbd397fefe6c63e850c641eb458791194c2c8eb0652ba
373bab41b95651778f8185924d7c2ec9c9584e8cbbac256884feb4028a92d5e1bb569775b5b5362f8a
5a58d154d98d5158fda494e067e96d91aeea3e88ea6768e7a9aa1da273bc9282c168e57c826ffcf8b3
01967260eadab33d98c656fd756fcc206f6e5b659f1d6f844cb52c4008d2aa5a94ea86c2448fdb2947
c7d2d507690e5ab7d08fb463c2471b52fbff5149bc21f1169ba0e52daa125cb859c9f574297814ccd3
a8eef8b5c92f5f890a3831ef3c42bfa0ae98d457d",
  "token_type": "Bearer",
  "expires_in": 43199,
  "scope": "openid",
  "id_token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6ImNPeXZidk5sQXhKUWk2dnZ5OU5jRGJHUkh3R
W5VVnJRbWphUVlCWjRYOTQifQ.eyJoYXNoIjoiYzllZGM5ODY2YmNlM2VlNTgwOGQ2Zjc1ODkzM2ZiZTgw
YzEyZGUyZCIsInJlYWxtTmFtZSI6ImN1c3RvbVJlYWxtIiwidW5pcXVlU2VjdXJpdHlOYW1lIjoiY3A0YX
V0b19jbGllbnQiLCJpc3MiOiJodHRwczovLzEyNy4wLjAuMTo0NDMvaWRhdXRoL29pZGMvZW5kcG9pbnQv
T1AiLCJhdWQiOiJjcDRhdXRvX2NsaWVudCIsImV4cCI6MTYyMDIwMjM4NywiaWF0IjoxNjIwMTU5MTg3LC
JzdWIiOiJjcDRhdXRvX2NsaWVudCIsInByZWZlcnJlZF91c2VybmFtZSI6ImNwNGF1dG9fY2xpZW50Iiwi
bmFtZSI6ImNwNGF1dG9fY2xpZW50IiwiZGlzcGxheV9uYW1lIjoiIiwiZ3JvdXBzIjpbInJlYWRlciIsIn
dyaXRlciIsIm1hbmFnZXIiXSwiZnVuY3Rpb25hbF91c2VyX2lkIjoiY3A0YXV0b190ZXN0ZXIifQ.ImJFI
9KM-
zt3yJziSolpzenagWGzh9Rdlta64KwOCeeszNLFNKWezAV_5_ShP40pkMBCZa7mwCLVFCpAbFlrlwANIG4
4zBfndgqYnBVmqDDIYrJUk-
yyLBOSiJ0CP02wn7Ltjtv70XP_sbcP9uxsu2V7TwyITsxSpucyd58A376jfG1EJTKmsEzlC4EfdE3qYy-
7Dn8UMF-
6QrvHUxEnSDEbxe0OJFz4NLnkJ1NspWbO5k5k4B1yPXqOAvynQVsmQ_095oFf2owFQDxQyZt7SBVg4mpe_
6n2eJr21tj2Uyabhz-N7ZwNUSYxegBnO0WO9xKJDeP8tjpy0dJ5obUHCQ"
}
```


Get information about a user

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/userInfo

Command

POST

Command output format

application/json

For information about **$ACCESS_TOKEN**, see Preparing to run API commands.

The sample curl command resembles the following code:

```
curl -k -X POST --header "Authorization: Bearer $ACCESS_TOKEN" \
https://<cluster_address>/idprovider/v1/auth/userInfo
```

You can also use the access token in the body:

```
curl -k -X POST -d "access_token=$ACCESS_TOKEN" \
https://<cluster_address>/idprovider/v1/auth/userInfo
```

The response resembles the following code:

```
{
  "sub":"admin",
  "iss":"https://mycluster.icp:9443/idprovider/v1/auth"
}
```

Call introspect endpoint

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/introspect

Command

GET

Command output format

application/json

1. Export these variables. To get the values, see Client Registration API.

```
export TOKEN=<your access token here>
export CLIENT_ID=<client_id here>
export CLIENT_SECRET=<client_secret here>
```

2. Get the Basic authorization header by using the following command:

```
BASIC_AUTH_HEADER=`echo -n "$CLIENT_ID:$CLIENT_SECRET" | base64 -w 0`
```

3. Run the curl command to call the endpoint.

```
curl -H "Authorization: Basic $BASIC_AUTH_HEADER" -d "token=$TOKEN"
https://<cluster_address>/idprovider/v1/auth/introspect
```

The response resembles the following code:

```
{
  "sub": "admin",
  "grant_type": "resource_owner",
  "realmName": "customRealm",
  "scope": "openid",
  "uniqueSecurityName": "admin",
  "iss": "https://127.0.0.1:9443/idprovider/v1/auth",
  "active": true,
  "exp": 1529358338,
  "token_type": "Bearer",
  "iat": 1529315138,
  "
```

Revoke access token or refresh token

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/revoke

Command

POST

Command output format

application/json

1. Export these variables. To get the values, see Client Registration API.

```
export TOKEN=<your access token here>
export CLIENT_ID=<client_id here>
export CLIENT_SECRET=<client_secret here>
```

2. Get the Basic authorization header by using the following command:

```
BASIC_AUTH_HEADER=`echo -n "$CLIENT_ID:$CLIENT_SECRET" | base64 -w 0`
```

3. Run the curl command to call the endpoint.

```
curl -k -X POST -H "Authorization: Basic $BASIC_AUTH_HEADER" -d
"token_type_hint=access_token&token=$TOKEN" \
https://<cluster_address>/idprovider/v1/auth/revoke
```

The response resembles the following code:

```
{}
```

Get a new access token by using the refresh token

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/token

Command

POST

Command output format

application/json

To get the refresh token, see Preparing to run API commands. To get the client ID and secret, see Client Registration API.

The sample curl command resembles the following code:

```
curl -H "Content-Type: application/x-www-form-urlencoded;charset=UTF-8" \
-d "grant_type=refresh_token&client_id=<client_ID>&client_secret=
<client_secret>&scope=openid&refresh_token=<refresh_token>" \
https://<cluster_address>/idprovider/v1/auth/token --insecure
```

The response resembles the following code:


```
{
  "access_token":
"77f3ea9695e50d147a3081990c331f8ce9baa0b6d02ac4e970c886eabccd7aa7e7f12e1897ceacbdf
6bdaf0881ed5a725f214209eb20b9415c2fcf4ad1afb90412a247aeab6ab0e026e08013b8f2b773b5b
db2d8d3c1247e9e7ebeaa8c9c9c66c1e85caf78105e35e934a28f21619bef2ff17cebe75792da86b4a
65c19973713559569e92ae6aa86ddb8ee48991c6ced9caf41ae6c3b88f67fcaacf8c2c6af82018b5f5
5a4e35c1b9026438b690a606de0314bdced35eab21642b4b6c33c5241db457f2564840b9d32c255d0b
fa9e4fda176416f7481c205ee98912790a11134597ce7245264669568fd69153a8e2f240df9edb4df3
b219e213c3cfb0366713802a9a525fe85c9ec2a8c54ba61b5d845054ff23eb466c990c15dcb025ef32
0f36bb21ec0d0a412bcdecafba57da6b239891e22c139a7d4057f84fd741215ed5567c3f4b824d9bbf
e92d56b77fe1712d35cea60e12f5207b727e3cc658db1b8b5002780049a5faefd8ccc2ccee9100472d
fff58978ee3e7303547dc4ea03025275e58ec4e3da8e6ae91939bfb092f1ce78fe2d91124c179f55bd
a4027957093090c4f47037771e9cacf227867063c909e9aee3bf87140426052821116c6484037822a4
1f05a0fa565276b5ff1a8a654d3d5d119f6a665469a7591e4ec197d6a90bd586b8b95e227b9869b865
4c23c10f78fc6a3fcbbe6d543638f379736193643",
  "token_type": "Bearer",
  "expires_in": 43199,
  "scope": "openid",
  "refresh_token": "5QM3H8fmGjxhPRyYlQ77s4Z5APOHVk5276ItT8q41e2xKNMxF6"
}
```


Get the OIDC configuration from the well-known configuration endpoint

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/.well-known/openid-configuration

Command

GET

Command output format

application/json

The sample curl command resembles the following code:

```
curl -k https://<cluster_address>/idprovider/v1/auth/.well-known/openid-
configuration --insecure
```

The response resembles the following code:

```
{"introspection_endpoint":"https://<cluster_address>/idprovider/v1/auth/introspect
","coverage_map_endpoint":"https://<cluster_address>/idprovider/v1/auth/coverage_m
ap","issuer":"https://<cluster_address>/idprovider/v1/auth","authorization_endpoin
t":"https://<cluster_address>/idprovider/v1/auth/authorize","token_endpoint":"http
s://<cluster_address>/idprovider/v1/auth/token","jwks_uri":"https://<cluster_addre
ss>/idprovider/v1/auth/jwk","response_types_supported":["token","id_token
token"],"subject_types_supported":
["public"],"id_token_signing_alg_values_supported":
["RS256"],"userinfo_endpoint":"https://<cluster_address>/idprovider/v1/auth/userIn
fo","registration_endpoint":"https://<cluster_address>/idprovider/v1/auth/registra
tion","scopes_supported":["openid","email","profile"],"claims_supported":
["sub"],"response_modes_supported":["query"],"grant_types_supported":
["client_credentials","password","refresh_token","authorization_code"],"token_endp
oint_auth_methods_supported":["client_secret_post"],"display_values_supported":
["page"],"claim_types_supported":
["distributed"],"claims_parameter_supported":true,"request_parameter_supported":tr
ue,"request_uri_parameter_supported":true,"require_request_uri_registration":true,
"check_session_iframe":"https://<cluster_address>/idprovider/v1/auth/check_session
_iframe","end_session_endpoint":"https://<cluster_address>/idprovider/v1/auth/end_
session","revocation_endpoint":"https://<cluster_address>/idprovider/v1/auth/revok
e","app_passwords_endpoint":"https://<cluster_address>/idprovider/v1/auth/app-
passwords","app_tokens_endpoint":"https://<cluster_address>/idprovider/v1/auth/app
-
tokens","personal_token_mgmt_endpoint":"https://<cluster_address>/idprovider/v1/au
th/personalTokenManagement","users_token_mgmt_endpoint":"https://<cluster_address>
/idprovider/v1/auth/usersTokenManagement","client_mgmt_endpoint":"https://<cluster
_address>/idprovider/v1/auth/clientManagement","code_challenge_methods_supported":
["plain","S256"]}
```

Get the Liberty and iam-token keys from the JSON Web Token (JWK) endpoint

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

/idprovider/v1/auth/jwk

Command

GET

Command output format

```
application/json
```

The sample curl command resembles the following code:

```
curl -k https://<cluster_address>/idprovider/v1/auth/jwk --insecure
```

The response resembles the following code:


```
{"keys":
[{"kty":"RSA","e":"AQAB","use":"sig","kid":"N23lhv0Waa2mXKHDvlF37tVByN8PofD1PPouOB
X-Bq8","alg":"RS256","n":"o1TdyuOlHNzVNGN8S0aBWiSd7E-89k741fQ-iJd4nD3ZFlz4TbmM-
lGhr2zsbV91M_IeoJzlJZcUKevl2us2JLKOQ8bG6T58qsioxwt3AL7KJ_aP3d3sUqNQk6zWzf08_BUhizs
lmIjsJkhH9Gl_Mj5vUaU8mL4k6P6SsSMk1wifHrCkt0N2fPM6SvBRoRTW0En14IczEyPXLRbPpy92YLMab
PEdP0Mmv4iK6_m1uXES3HD9cpfm7LuYgqZ2Ws3NrvfM4a9FqR8OriR6tD5t4hbtT5S3UGq-
eE6vTqyJtix0uPcTMRSGAx9xSWi6B-bXKRnuBATNez-FdvTMcCf_oQ"},
{"kty":"RSA","n":"AIdJNOnqGoCpfZcg1-
AMOOnVaQCfcZkAweku7D5uM6CVuXsdsxip_liHpTs7A01e8BM3qCxH_YbtTqbLqxR2TKmLSzGMG3QnMZzm
OunBuR_w2KuBQyz7IBDImaQlCDuEEv05wnQiryFj5B_wK6dHIRdbrlOFTP2ebjEf8gkwjxdyl32vJ-
Pqy0FksAfxHFTaccSuOrVycFLtx_MyzyexP_N76du_n6GyjwkqzeUbLDdJHET4Vfdp6R4O5Cdz9zMQI4sy
7r07rFLLJMrP9rcuRZQWVKZjM4X6Cw3ptnOwVlsvEesD4W0mBHZoqjZ3Dz5ET5IGHGWPuV8p39M6rDON2A
s","e":"AQAB","alg":"RS256","kid":"20170721-00:00:00","use":"sig"},
{"kty":"RSA","n":"AIaaIOCBGGJGXsKmW7JGLXqe7nHSWcnCMQ51gDvNhL5oAV0QxWRjpW0GDScpoA_K
CBn5zEkTqSDrxgmhWKa3-Uv_D6xaewZWRWh7Wnp3AIbc-LuL-aiKuwwWqKc1moopovT_F5-
IEfPZHN87COPEOyQ_MY-jgSuG5UBHaBhMP8dEb7C35IIFqbahHKFBPgZiTwu97OkAZYzJmzFw-
1PbAfSR3EHkxcduTeSWlROat1gr7CmJhNBYKUk4xpb_vt4iDWvkbiKbWR_x3nsWnzntq5TjGYjejyS7rYN
FR2W-gmkxGXbMiAzq2UkbFJwTnxEp7rMuKC4CiybCWyNO-
VmDKVU","e":"AQAB","alg":"RS256","kid":"20170515-00:00:00","use":"sig"},
{"kty":"RSA","n":"AIkkfUhukP6NMes2A4423vhLCXNbbqshDNOaN5y-GHUOxn-
rSsXJY9yBu9ykMwhcZTsWuCvtlFyo2IwQC9aBya1MJ6elebw1fYVbwG3w55ZevLl9LJCzSI8vtC8yitv6x
KF1dHRWf4VHq35PCdY8gm-uOm-7OUzm7qB1NOIb8c-nhfK-
PK0cfkAVJKdpEu8ALLwG6pBBk7ZAoe3PLpABY1my40iKEy1D0jqWpb2mamKZdLuOZ2QbhUgE5aI1eGtXYS
eIbFuFTTlhJUEkjA-iBD2mE7CFKhfkjqFVjy_jDMg-X8y9sR8jrX0sWp2Z3mYkCmc-
XgN1I_Ws5sIyTxoyOJU","e":"AQAB","alg":"RS256","kid":"20170401-
00:00:00","use":"sig"},{"kty":"RSA","n":"AIdJNOnqGoCpfZcg1-
AMOOnVaQCfcZkAweku7D5uM6CVuXsdsxip_liHpTs7A01e8BM3qCxH_YbtTqbLqxR2TKmLSzGMG3QnMZzm
OunBuR_w2KuBQyz7IBDImaQlCDuEEv05wnQiryFj5B_wK6dHIRdbrlOFTP2ebjEf8gkwjxdyl32vJ-
Pqy0FksAfxHFTaccSuOrVycFLtx_MyzyexP_N76du_n6GyjwkqzeUbLDdJHET4Vfdp6R4O5Cdz9zMQI4sy
7r07rFLLJMrP9rcuRZQWVKZjM4X6Cw3ptnOwVlsvEesD4W0mBHZoqjZ3Dz5ET5IGHGWPuV8p39M6rDON2A
s","e":"AQAB","alg":"RS256","kid":"20170301-00:00:00","use":"sig"},
{"e":"AQAB","kty":"RSA","n":"o1TdyuOlHNzVNGN8S0aBWiSd7E-89k741fQ-iJd4nD3ZFlz4TbmM-
lGhr2zsbV91M_IeoJzlJZcUKevl2us2JLKOQ8bG6T58qsioxwt3AL7KJ_aP3d3sUqNQk6zWzf08_BUhizs
lmIjsJkhH9Gl_Mj5vUaU8mL4k6P6SsSMk1wifHrCkt0N2fPM6SvBRoRTW0En14IczEyPXLRbPpy92YLMab
PEdP0Mmv4iK6_m1uXES3HD9cpfm7LuYgqZ2Ws3NrvfM4a9FqR8OriR6tD5t4hbtT5S3UGq-
eE6vTqyJtix0uPcTMRSGAx9xSWi6B-bXKRnuBATNez-
FdvTMcCf_oQ","use":"sig","kid":"ICk0vQmsxQXvN87q-C8-2s91ts6xiifg15T0iv1KJpo"}]}
```


Identity Provider APIs

Use Identity Provider (IdP) V3 APIs to register IdP connection and configuration.

If you are configuring SSO (single sign-on), you must use the IdP V3 APIs.

See the following notes:

- You can get the hostname from the route available in your foundational services namespace. To get the host details, see Getting the host details.

- The **`Administrator`** IM role is allowed to access the IdP V3 APIs.

IdP V3 APIs

You can register the following providers and connections by using the IdP V3 APIs:

- OIDC provider For more information, see Registering OIDC clients by using IdP V3.
- SAML provider For more information, see Registering SAML clients by using IdP V3. To configure SAML, use IBM Cloud Pak for AIOps UI. For more information, see Configuring single sign-on using IBM Cloud Pak for AIOps UI.
- LDAP For more information, see Configuring LDAP connection by using IdP V3.

OIDC clients

Currently, only the following platforms are verified to register OIDC by using IdP V3 APIs.

Note: Service providers might change the links that are listed here. If any links is broken, check the service provider website for the new location.

- IBM Security Verify
  - **Note:** In the IBM Security Verify application, whenever you map attributes, enable the **Send all known user attributes in the ID token** option. And, do not use spaces in IBM Security Verify groups.
- Google Cloud Platform
- Microsoft
- Okta

Before you begin, register yourself in your service provider application. During registration, use the following URLs:

- For application URL, use the cp-console URL.
- For redirect URL, use **`https://<cp-console-url>/ibm/api/social-login/redirect/<name of the oidc>`**.

After you register, you get a unique client ID, client secret, and **`discovery_url`** endpoint.

Note: The curl commands that are used in the following sections contain different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3.

- Registering the OIDC clients
- Deleting the registration of the client
- Getting the list of registered OIDC clients
- Updating the OIDC clients

Registering OIDC clients

API details - register OIDC clients - POST

API version
1.0.0
API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

idprovider/v3/auth/idsource

Command

POST

Command output format

application/json

Note: By using the POST call, you can get the unique identifier (UID) as a successful response. The UID is required for the further operations or tasks. You can also get the list of UIDs of registered IdPs by using the Getting the list of registered OIDC clients by query API.

Sample curl command to register a client

```
curl --insecure POST 'https://cp-console.apps.vij-
soc.cp.fyre.ibm.com/idprovider/v3/auth/idsource/'
--header "Authorization: Bearer $ACCESS_TOKEN"
--header 'Content-Type: application/json'
--data-raw
'{
    "name": "oidc_isv",
    "description": "This is a oidc config for isv",
    "protocol": "oidc",
    "type": "IBMVerify",
    "idp_config": {
        "discovery_url": "https://bedrock-
iam.verify.ibm.com/oidc/endpoint/default/.well-known/openid-configuration",
        "client_id": "4ad41ddc-5f10-4d84-a24f-fb776607895e",
        "client_secret": "rqJMyEgee",
        "token_attribute_mappings": {
            "groups": "groupIds",
            "given_name": "given_name",
            "family_name": "family_name",
            "sub": "uid",
            "email": "email"
        }
    }
}'
```

Sample response - register a client

```
{
    "status": "success",
    "message": "Identity provider {oidc_isv} is successfully registered with
unique identifier NpiV6qBhsSDEjKRvJDQLl"
}
```

Sample error message - register a client

```
400 "error":  "E11000 duplicate key error collection: platform-
db.cloudpak_ibmid_v3 index: name_1 dup key: { : <oidc name>" }"
500 "error": "Other errors"
401 "error": "Insufficient user permission for role :Viewer"
202 OK Response
```

Deleting the registration of a client

API details - delete registration - DELETE

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

idprovider/v3/auth/idsource/:UID

Command

DELETE

Command output format

application/json

Sample curl command to delete the client registration

```
curl --insecure DELETE 'https://cp-console.apps.vij-
soc.cp.fyre.ibm.com/idprovider/v3/auth/idsource/NpiV6qBhsSDEjKRvJDQLl' --header
"Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - delete client registration

```
{
    “status”: “success”,
    “message”: “{NpiV6qBhsSDEjKRvJDQLl} is deleted”
}
```

Sample error message - delete client registration

```
400 "error": "Cannot find {oidc uid}"
401 "error": "Insufficient user permission for role :Viewer"
200: Ok Response
```

Getting the list of registered OIDC clients

You can get the list of registered OIDC clients by using either of the following methods:

- Unique identifier (UID)
- Query

Getting the list of registered OIDC clients by UID

API details - get OIDC clients by UID - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

GET

Command output format

application/json

Sample curl command to get the list of registered clients by UID

```
curl --insecure --request GET 'https://cp-
console.cp.fyre.ibm.com/idprovider/v3/auth/idsource/yOLi5oHo3t-es4aJFLkC8' --
header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - list registered clients by UID

```
{
  "name": "oidc_1",
  "description": "This is a oidc config",
  "protocol": "oidc",
  "type": "IBMVerify",
```

```
  "idp_config": {
    "discovery_url": "https://bedrock-
iam.verify.ibm.com/oidc/endpoint/default/.well-known/openid-configuration",
    "client_id": "My_Client_ID",
    "client_secret": "My_Client_Secret",
    "token_attribute_mappings": {
      "groups": "groupIds"
    }
  },
  "uid": "NpiV6qBhsSDEjKRvJDQLl"
}
```

Note: For security reasons, the values of the **client_secret** is not displayed in the output when you retrieve the list of registered clients with UID.

Sample error message - list registered clients by UID

```
400 "error": "Cannot find {oidc uid}"
401 "error": "Insufficient user permission for role :Viewer"
200: Ok Response
```

Getting the list of registered OIDC clients by query

API details - get OIDC clients by query - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

GET

Command output format

application/json

Sample curl command to get the list of registered clients by query

```
curl --insecure --request GET 'https://cp-
console.cp.fyre.ibm.com/idprovider/v3/auth/idsource?protocol=oidc' --header
```

```
"Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - list registered clients by query

```
{
    "idp": [
        {
            "name": "oidc_google",
            "description": "This is a oidc config for google",
            "protocol": "oidc",
            "type": "google",
            "idp_config": {
                "discovery_url": "https://accounts.google.com/.well-known/openid-
configuration",
                "client_id": "My_Client_ID",
                "client_secret": "My_Client_Secret",
                "token_attribute_mappings": {
                    "groups": "groupIds",
                    "given_name": "firstName",
                    "family_name": "lastName",
                    "sub": "uid",
                    "email": "email",
                    "uniqueSecurityName": "uniqueSecurityName"
                }
            },
            "uid": "Uo4Y31mV7unaj5NtyfpIY"
        },
        {
            "name": "oidc_isv",
            "description": "This is a oidc config",
            "protocol": "oidc",
            "type": "IBMVerify",
            "idp_config": {
                "discovery_url": "https://bedrock-
iam.verify.ibm.com/oidc/endpoint/default/.well-known/openid-configuration",
                "client_id": "My_Client_ID",
                "client_secret": "My_Client_Secret",
                "token_attribute_mappings": {
                    "groups": "groupIds",
                    "given_name": "given_name",
                    "family_name": "family_name",
                    "sub": "uid",
                    "email": "email",
                    "uniqueSecurityName": "uniqueSecurityName",
                    "preferred_username": "preferred_username",
                    "displayName": "displayName"
                }
            },
            "uid": "NpiV6qBhsSDEjKRvJDQLl"
        }
    ]
}
```

Note: For security reasons, the values of the **client_secret** is not displayed in the output when you retrieve the list of registered clients with query.

Sample error message - list registered clients by query

```
400 "error": "Cannot find {oidc uid}"
401 "error": "Insufficient user permission for role :Viewer"
200: Ok Response
```

Updating the OIDC clients

API details - update OIDC clients - PUT

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster Port

Path

idprovider/v3/auth/idsource/:UID

Command

PUT

Command output format

application/json

Sample curl command to update the client

```
curl --insecure PUT 'https://cp-
console.cp.fyre.ibm.com/idprovider/v3/auth/idsource/NpiV6qBhsSDEjKRvJDQLl'
--header "Authorization: Bearer $ACCESS_TOKEN"
--header 'Content-Type: application/json'
--data-raw
'{
    "name": "oidc_isv",
    "description": "This is a oidc config for isv",
    "protocol": "oidc",
    "type": "IBMVerify",
    "idp_config": {
        "discovery_url": "https://bedrock-
iam.verify.ibm.com/oidc/endpoint/default/.well-known/openid-configuration",
        "client_id": "4ad41ddc-5f10-4d84-a24f-fb776607895e",
        "client_secret": "rqJMyEgee",
        "token_attribute_mappings": {
            "groups": "groupIds",
            "given_name": "given_name",
            "family_name": "family_name",
            "sub": "uid",
            "email": "email"
        }
    }
}'
```

Sample response - update client

```
{
    "status": "success",
    "message": "{NpiV6qBhsSDEjKRvJDQLl} is Updated."
}
```

Sample error message - update client

```
400 "error": "Document not found"
401 "error": "Insufficient user permission for role :Viewer"
200: Ok Response
```

SAML clients

You can register the SAML IdP by the following methods:

- SAML with SCIM dependency registration
- SAML registration without any dependency
- SAML with LDAP dependency registration

Note: You can configure only one SAML registration at an instance. The UID generated on the POST request is **`defaultSP`**. For **`idp_metadata`**, you need to provide the base64-encoded value of the generated IdP metadata **`xml`** file.

Get the base64-encoded value by using the following command:

```
cat idp-metadata-xml | base64
```

Note: The maximum accepted size of the metadata file is **`2 MB`**. If the size of the metadata file is more than 2 MB, an error message is displayed during the SAML registration.

SAML with SCIM dependency registration

You can register IdP V3 with the following platforms:

- IdP V3 registration with IBM Security Verify
- IdP V3 registration with Okta

Note: The curl commands that are used in the following sections contain different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3.

Note: By using the POST call, you can get the unique identifier (UID) as a successful response. The UID is required for the further operations or tasks. You can also get the list of UIDs of registered IdPs by using the Getting the list of registered OIDC clients by query API.

IdP V3 registration with IBM Security Verify

The following example shows IdP V3 registration with IBM Security Verify that supports SCIM-enabled IdP with SAML. Hence, the **`token_attribute_mappings`** and **`scim_attribute_mappings`** can be different for the respective IdPs.

API details - registration with IBM Security Verify - POST

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource

Command

POST

Command output format

application/json

Sample curl command - registration with IBM Security Verify

```
curl --insecure POST 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource'
--header "Authorization: Bearer $ACCESS_TOKEN"
--header 'Content-Type: application/json'
--data-raw
'{
  "name": "idp_saml_isv",
  "description": "This is a saml isv config",
  "protocol": "saml",
  "type": "isv",
  "idp_config": {
    "token_attribute_mappings": {
      "sub": "userID",
      "given_name": "given_name",
      "family_name": "family_name",
      "groups": "groupIds",
      "email": "email"
    },
    "idp_metadata": "<base64-encoded-idp-xml>"
  },
  "scim_config": {
    "scim_base_path": "https://bedrock-iam.verify.ibm.com/v2.0/",
    "grant_type": "client_credentials",
    "token_url": "https://bedrock-iam.verify.ibm.com/v1.0/endpoint/default/token",
    "client_id": "Your_isv_client_id",
    "client_secret": "Your_isv_client_secret",
    "scim_attribute_mappings": {
      "user": {
        "principalName": "userName",
        "givenName": "name.givenName",
        "middleName": "name.middleName",
        "familyName": "name.familyName",
```

```
        "formatted": "name.formatted"
      },
      "group": {
        "principalName": "displayName",
        "created": "meta.created",
        "lastModified": "meta.lastModified"
      }
    }
  },
  "jit": "false"
}'
```

Sample response - registration with IBM Security Verify

```
{
    "status": "success",
    "message": "Identity provider {idp_saml_isv} is successfully registered with
unique identifier defaultSP"
}
```

Sample error message - registration with IBM Security Verify

```
401 "error": "Insufficient user permission for role : rolename"
400 "error": "schema error:schema validation error followed by error"
400 "error": "duplicate : Idp with protocol=saml is already created"
500 "error": "Other errors"
200 OK Response
```

IdP V3 registration with Okta

The following example shows IdP V3 registration with Okta.

API details - registration with Okta - POST

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/

Command

POST

Command output format

```
application/json
```

Table 1. Command parameters

Mapping	Description
`token_attribute_mappings`	A custom mapping that you can provide based on the SAML attribute mapping that is configured at Okta. You must modify the values, not the key, based on the Okta SAML attribute configuration. For more information about attributes, see step 9 in Configure CloudPak as SAML service provider (SP) at OKTA and Okta doc for SAML attributes.
`scim_attribute_mappings`	Provide the mapping attributes as shown in the curl command for OKTA registration as SAML IdP. The defined mapping is used to filter query, and to create attributes and responses to be sent.
`redirect_url`	Only applicable for the IdP **"type": "okta"**. Must be an array. The `redirect_url` is used to create the OIDC client registration for the OKTA instance. For a non-IBM OIN app, the `redirect_uri` can be one of the following URIs based on the environment (a standard org, an EMEA org, or an Okta Preview org): https://system-admin.okta.com/admin/app/cpc/${appName}/oauth/callback https://system-admin.okta-emea.com/admin/app/cpc/${appName}/oauth/callback https://system-admin.oktapreview.com/admin/app/cpc/${appName}/oauth/callback **Note:** Replace the `${appName}` with the variable name for your application instance. If you don't know the variable name for application instance, you can find it as part of the App embed link from the browser by opening the embed link, or you can also check the **Name** attribute for the application at the `/Apps API` endpoint. For example, if the URL embedded link is https://dev-68798908-admin.okta.com/admin/app/dev-68798908_iamsaml_3/instance/0oa6vnyhkm86B4xha5d7/#tab-general, the **${appName}** is **dev-68798908_iamsaml_3** and so the redirect_uri is https://system-admin.okta.com/admin/app/cpc/dev-68798908_iamsaml_3/oauth/callback. For more information, see Okta documentation.

Sample curl command - registration with Okta

```
curl --location --request POST
'https://cpconsoleroute/idprovider/v3/auth/idsource' \
--header 'Authorization: Bearer $accessToken' \
--header 'Content-Type: application/json' \
--data-raw '{
            "name": "OKTASAML",
            "description": "OKTASAML",
            "protocol": "saml",
            "type": "okta",
            "idp_config": {
                "token_attribute_mappings": {
                    "sub": "uid",
                    "given_name": "firstName",
                    "family_name": "lastName",
                    "groups": "groups",
                    "email": "emailAddress",
                    "first_name": "firstName",
                    "last_name": "lastName"
                },
                "idp_metadata": "Base64-encoded-metadata"
```

```
            },
            "jit": false,
            "scim_config": {
                "redirect_url": ["okta_redirect_url"],
                "scim_attribute_mappings": {
                    "user": {
                        "principalName": "userName",
                        "name": {
                            "givenName": "givenName",
                            "familyName": "familyName"
                        },
                        "displayName": "displayName",
                        "emails": [
                            {
                                "value": "emails",
                                "type": "home"
                            }
                        ],
                        "id": "id",
                        "userName": "userName"
                    },
                    "group": {
                        "principalName": "displayName",
                        "id": "displayName"
                    }

                }
            }
        }'
```

Sample response - registration with Okta

```
{
    "status": "success",
    "message": "Identity provider {oktasaml} is successfully registered with
unique identifier : defaultSP, oktaclient created with clientId:
yx3a5pwtdrmn0sf8ockyf97dg55lswo7 , client secret: okta-oidcclient-secret"
}
```

Sample error message - registration with Okta

```
401 "error": "Insufficient user permission for role : rolename"
400 "error": "schema error:schema validation error followed by error"
400 "error": "duplicate : Idp with protocol=saml is already created"
500 "error": "Other errors"
200 OK Response
```

SAML registration without any dependency

API details - registration without any dependency - POST

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/

Command

POST

Command output format

application/json

See the following notes:

- You can find the SAML attribute and claim names from the IdP administration. Specify the SAML attributes and claim names as the values in the **token_attribute_mappings** attributes.
- You can also provide an optional attribute named **uniqueSecurityName** in **token_attribute_mappings** payload to map the IdP attributes. By default, the **uniqueSecurityName** attribute is mapped to the **Subject** in the response of **userinfo** or **introspect** endpoint.

Sample curl command for SAML registration with w3id SAML provider

```
curl -k -X POST 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
'{
  "name": "w3id-sample-saml",
  "description": "w3id-sample-saml-test",
  "protocol": "saml",
  "type": "default",
  "idp_config": {
    "token_attribute_mappings": {
      "sub": "uid",
      "given_name": "firstName",
      "family_name": "lastName",
      "groups": "blueGroups",
      "email": "emailAddress"
    },
    "idp_metadata": "<base64-encoded-idp-xml>"
  },
  "jit": true
}'
```

Sample response - registration with w3id SAML provider

```
{
    "status": "success",
    "message": "Identity provider {w3id-sample-saml} is successfully registered
with unique identifier defaultSP"
}
```

Sample error message - registration with w3id SAML provider

```
401 "error": "Insufficient user permission for role : rolename"
400 "error": "schema error:schema validation error followed by error"
400 "error": "duplicate : Idp with protocol=saml is already created"
500 "error": "Other errors"
200 OK Response
```

SAML with LDAP dependency registration

The following example is provided for IBM W3 SAML provider and with dependency of an IBM Tivoli Directory Server LDAP. The **token_attribute_mappings** might change based on the SAML provider.

API details - SAML with LDAP dependency registration - POST

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/

Command

POST

Command output format

application/json

Note: When LDAP is configured, the response for **userinfo** or **introspect** endpoints are directly collected from liberty. Therefore, the attributes that are provided in the **token_attribute_mappings** payload are not available in the response of the **userinfo** or **introspect** endpoints.

Sample curl command - SAML with LDAP dependency registration

```
curl -k -X POST 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
'{
  "name": "saml-test",
  "description": "saml bluepages with ldap",
  "protocol": "saml",
  "type": "default",
  "idp_config": {
    "token_attribute_mappings": {
      "sub": "uid",
      "given_name": "firstName",
      "family_name": "lastName",
      "groups": "blueGroups",
      "email": "email"
    },
    "idp_metadata": "<base64-encoded-idp-xml>"
  },
  "ldap_config": {
    "ldap_id": "<existing_ldap_id1>,<existing_ldap_id2>,<existing_ldap_id3>"
  },
  "jit": false
}'
```

Sample response - SAML with LDAP dependency registration

```
{
    "status": "success",
    "message": "Identity provider {saml-test} is successfully registered with
unique identifier defaultSP"
}
```

Sample error message - SAML with LDAP dependency registration

```
401 "error": "Insufficient user permission for role : rolename"
400 "error": "schema error:schema validation error followed by error"
400 "error": "duplicate : Idp with protocol=saml is already created"
500 "error": "Other errors"
200 OK Response
```

Getting SAML registration by UID

You can get the IdP registration by the following ways:

- By name
- By list
- By query

Getting IdP V3 registration by name

The following example shows how to get the details of IdP V3 registration by name:

API details - registration by UID - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/defaultSP

Command

GET

Command output format

application/json

Note: The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3.

Sample curl command - registration by UID

```
curl -k -X GET 'https://cp-
console.apps.mycluster.mydomain.com/idprovider/v3/auth/idsource/defaultSP' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - registration by UID

```
{
    "name": "w3id-sample-saml",
    "description": "w3id-sample-saml-test",
    "protocol": "saml",
    "type": "default",
    "idp_config": {
        "token_attribute_mappings": {
            "sub": "uid",
            "given_name": "firstName",
            "family_name": "lastName",
            "groups": "blueGroups",
            "email": "emailAddress"
        },
        "idp_metadata": "<base64-encoded-idp-xml>"
    },
    "jit": true,
    "uid": "defaultSP"
}
```

Sample error message - registration by UID

```
401 "error": "Insufficient user permission for role : rolename"
404 "error": "Cannot find {defaultSP}
500 "error": "Other errors"
200 OK Response
```

Getting the list of IdP V3 registration

The following example shows how to get the list of IdP V3 registration:

Note: This API returns all registered IdPs of SAML or OIDC clients through IdP V3 API.

API details - list of IdP V3 registration - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource

Command

GET

Command output format

application/json

Note: The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3.

Sample curl command - IdP V3 registration

```
curl -k -X GET 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - IdP V3 registration

```
{
    "idp": [
        {
            "name": "w3id-sample-saml",
```

```
            "description": "w3id-sample-saml-test",
            "protocol": "saml",
            "type": "default",
            "idp_config": {
                "token_attribute_mappings": {
                    "sub": "uid",
                    "given_name": "firstName",
                    "family_name": "lastName",
                    "groups": "blueGroups",
                    "email": "emailAddress"
                },
                "idp_metadata": "<base64-encoded-idp-xml>"
            },
            "jit": true,
            "uid": "defaultSP"
        }
    ]
}
```

Sample error message - IdP V3 registration

```
401 "error": "Insufficient user permission for role : rolename"
500 "error": "Other errors"
200 OK Response
```

Getting SAML registration by query

The following example shows how to get the IdP V3 registration by query:

API details - SAML registration by query - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

GET

Command output format

application/json

Note: The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3.

Sample curl command - SAML registration by query

```
curl -k -X GET 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource?
protocol=saml'
\
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - SAML registration by query

```
{
    "idp": [
        {
            "name": "w3id-sample-saml",
            "description": "w3id-sample-saml-test",
            "protocol": "saml",
            "type": "default",
            "idp_config": {
                "token_attribute_mappings": {
                    "sub": "uid",
                    "given_name": "firstName",
                    "family_name": "lastName",
                    "groups": "blueGroups",
                    "email": "emailAddress"
                },
                "idp_metadata": "<base64-encoded-idp-xml>"
            },
            "jit": true,
            "uid": "defaultSP"
        }
    ]
}
```

Sample error message - SAML registration by query

```
401 "error": "Insufficient user permission for role : rolename"
500 "error": "Other errors"
200 OK Response
```

Updating SAML registration

To update the SAML registration, you require the IdP UID. To get the IdP UID, see Get the list of IdP V3 registration. Alternatively, you can use the UID as **`defaultSP`** because only one SAML registration at an instance is supported for now.

The following example shows how to update the IdP registration:

API details - Update SAML registration - PUT

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

PUT

Command output format

application/json

Note: The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3.

Sample curl command - Update SAML registration

```
curl -k -X PUT 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/defaultSP
' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
' {
  "name": "w3id-sample-saml",
  "description": "this is plain saml testing",
  "protocol": "saml",
  "type": "default",
  "idp_config": {
    "token_attribute_mappings": {
      "sub": "uid",
      "given_name": "firstName",
      "family_name": "lastName",
      "groups": "blueGroups",
      "email": "emailAddress"
    },
    "idp_metadata": "<base64-encoded-idp-xml>"
  },
  "jit": true
}'
```

Sample response - Update SAML registration

```
{
    "status": "success",
    "message": "{defaultSP} is Updated."
}
```

Sample error message - Update SAML registration

```
401 "error": "Insufficient user permission for role : rolename"
400 "error": "schema validation error:followed by error"
500 "error": "Other errors"
200 OK Response
```

Deleting SAML registration

To delete the SAML registration, you require the IdP UID. To get the IdP UID, see Get the list of IdP V3 registration. Alternatively, you can use the UID as **defaultSP** because only one SAML registration at an instance is supported for now.

The following example shows how to delete the IdP registration:

API details - Delete SAML registration - DELETE

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

DELETE

Command output format

application/json

Note: The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3.

Sample curl command - Delete SAML registration

```
curl -k -X DELETE 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/defaultSP
' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - Delete SAML registration

```
{
    "status": "success",
    "message": "{defaultSP} is deleted"
}
```

Sample error message - Delete SAML registration

```
400 "error": "Insufficient user permission for role : rolename"
404 "error": "Document not found"
500 "error": "Other errors"
202 OK Response
```

SAML metadata download by using IdP V3

You can export the SAML metadata by using IdP V3 API.

Details of downloading SAML metadata by using IdP V3 API.

Downloading the SAML metadata by using IdP V3 API

The following example shows how to export SAML metadata by using the **`samlmetadata`** API:

API details - download SAML metadata - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

/idprovider/v3/auth/idsource/

Command

GET

Command output format

application/xml

Sample curl command - download SAML metadata

```
curl -k -v -X GET
'https://<cluster_address>/idprovider/v3/auth/idsource/<SERVICE_PROVIDER_ID>' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Example command - download SAML metadata

```
curl -k -v -X GET 'https://cp-
console.apps.cp.fyre.ibm.com/idprovider/v3/auth/idsource/defaultSP' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Note: If you choose to download SAML metadata API, the response code shows `200 OK` status with SAML metadata sample.

Sample response - download SAML metadata

```
<?xml version="1.0" encoding="UTF-8"?>
<md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata"
entityID="https://cp-console.apps.tamil-bedrock-
dev.cp.fyre.ibm.com/ibm/saml20/defaultSP">
    <md:SPSSODescriptor AuthnRequestsSigned="true" WantAssertionsSigned="true"
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
        <md:KeyDescriptor use="signing">
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
                <ds:X509Data>

<ds:X509Certificate>MIIDWjCCAkKgAwIBAgIRALy+U3ooWL4WotnPuKUixTswDQYJKoZIhvcNAQELBQ
AwHDEaMBgGA1UE
AxMRY3MtY2EtY2VydGlmaWNhdGUwHhcNMjIwNDE4MTIyODI4WhcNMjMwNDE4MTIyODI4WjAdMRsw
GQYDVQQDExJtYW5hZ2VtZW50LWluZ3Jlc3MwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIB
AQC6h0rYALFRA104/DGWLlfiuftuoxT+Ab2FzsnmkO1q8Iu/SUhLf5FFnb05BrcemrL+0OJcZEyR
ALdxliWSWPG/I/9Uj+BcVlVdQupND/RQTQHS5ECYEoLJJvHkFHVj7+/WxyN3eYmqlz9OZHKp17f1
t79HDOYPZSKVzhLhZ9MsLM/G3xIF8feRpJRoQSYUUKHB6gDXZ4Kbfui7kd/LMrQK820psA8z/Bjy
dYQ2SXXUo4BfgpUxjN+VM1dGUy0khkuUSdNKlW9pLmZdT1FH7FjWH9zBWOOHz8szr+Pr4ZavHmFL
SUP4nDMpFeH/v6+cBm48LrezWAAUzZuSTYxRQeuhAgMBAAGjgZUwgZIwDgYDVR0PAQH/BAQDAgWg
MBMGA1UdJQQMMAoGCCsGAQUFBwMBMAwGA1UdEwEB/wQCMAAwHwYDVR0jBBgwFoAUzW7sLnme6JsG
DZM/SC2ZRUwS/OswPAYDVR0RBDUwM4IxY3AtY29uc29sZS5hcHBzLnRhbWlsLWJlZHJvY2stZGV2
LmNwLmZ5cmUuaWJtLmNvbTANBgkqhkiG9w0BAQsFAAOCAQEASHgDST++rAi79e0P5TXrUF2QQ7dc
o2pgiv/xNKvwm/URzaSGZ1CYq69hXOd+SFZ0sverFFEQZhBXxKVIDGFMT2Jll189sTtQsQnodbpj
gwQxB7c4KbqI4EKU33jj9lT5CmmzHWtO5cPa7br4nOcPKp7bl2q79/aDbP7GYLMp8sGMacopq5J7
vAkL3O3De3sv2p7wg5nHB0JBA/k7Ecd/34gKpftfwZxNsBAxL+dfwOdHjK1bSlHoyo5F7gjwvLVx
dCGdw0AfSNuw0efyGEK2+bA9bc+pzGQf6vKaaSQpdL/YC6b5QLpQkpculwl+lLESFJkdEzmM2VRB
XfgckB39kg==</ds:X509Certificate>
                </ds:X509Data>
            </ds:KeyInfo>
        </md:KeyDescriptor>
        <md:KeyDescriptor use="encryption">
            <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
                <ds:X509Data>

<ds:X509Certificate>MIIDWjCCAkKgAwIBAgIRALy+U3ooWL4WotnPuKUixTswDQYJKoZIhvcNAQELBQ
AwHDEaMBgGA1UE
AxMRY3MtY2EtY2VydGlmaWNhdGUwHhcNMjIwNDE4MTIyODI4WhcNMjMwNDE4MTIyODI4WjAdMRsw
GQYDVQQDExJtYW5hZ2VtZW50LWluZ3Jlc3MwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIB
AQC6h0rYALFRA104/DGWLlfiuftuoxT+Ab2FzsnmkO1q8Iu/SUhLf5FFnb05BrcemrL+0OJcZEyR
ALdxliWSWPG/I/9Uj+BcVlVdQupND/RQTQHS5ECYEoLJJvHkFHVj7+/WxyN3eYmqlz9OZHKp17f1
t79HDOYPZSKVzhLhZ9MsLM/G3xIF8feRpJRoQSYUUKHB6gDXZ4Kbfui7kd/LMrQK820psA8z/Bjy
dYQ2SXXUo4BfgpUxjN+VM1dGUy0khkuUSdNKlW9pLmZdT1FH7FjWH9zBWOOHz8szr+Pr4ZavHmFL
SUP4nDMpFeH/v6+cBm48LrezWAAUzZuSTYxRQeuhAgMBAAGjgZUwgZIwDgYDVR0PAQH/BAQDAgWg
MBMGA1UdJQQMMAoGCCsGAQUFBwMBMAwGA1UdEwEB/wQCMAAwHwYDVR0jBBgwFoAUzW7sLnme6JsG
DZM/SC2ZRUwS/OswPAYDVR0RBDUwM4IxY3AtY29uc29sZS5hcHBzLnRhbWlsLWJlZHJvY2stZGV2
LmNwLmZ5cmUuaWJtLmNvbTANBgkqhkiG9w0BAQsFAAOCAQEASHgDST++rAi79e0P5TXrUF2QQ7dc
```

```
o2pgiv/xNKvwm/URzaSGZ1CYq69hXOd+SFZ0sverFFEQZhBXxKVIDGFMT2Jll189sTtQsQnodbpj
gwQxB7c4KbqI4EKU33jj9lT5CmmzHWtO5cPa7br4nOcPKp7bl2q79/aDbP7GYLMp8sGMacopq5J7
vAkL3O3De3sv2p7wg5nHB0JBA/k7Ecd/34gKpftfwZxNsBAxL+dfwOdHjK1bSlHoyo5F7gjwvLVx
dCGdw0AfSNuw0efyGEK2+bA9bc+pzGQf6vKaaSQpdL/YC6b5QLpQkpculwl+lLESFJkdEzmM2VRB
XfgckB39kg==</ds:X509Certificate>
                </ds:X509Data>
            </ds:KeyInfo>
        </md:KeyDescriptor>
        <md:SingleLogoutService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST" Location="https://cp-
console.apps.tamil-bedrock-dev.cp.fyre.ibm.com/ibm/saml20/defaultSP/slo"/>
        <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST" Location="https://cp-
console.apps.tamil-bedrock-dev.cp.fyre.ibm.com/ibm/saml20/defaultSP/acs" index="0"
isDefault="true"/>
    </md:SPSSODescriptor>
</md:EntityDescriptor>
```

Configuring LDAP connection by using IdP V3 API

IMPORTANT: Creating and deleting an LDAP connection simultaneously can cause misconfiguration at the backend. Therefore, avoid creating and deleting the LDAP connection simultaneously.

Before you begin: Ensure that you are informed about the other LDAP configuration parameters that are used in the Identity Provider V3 APIs. For more information, see LDAP authentication, LDAP filters and Default LDAP filters by LDAP type. The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3 and Schema elements for LDAP connection.

Validating LDAP configuration while creating the LDAP connection using IdP V3 API

The following example shows how to validate the LDAP connection while creating an LDAP connection by using IdP V3 API:

API details - validate LDAP connection while creating - POST

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/ldap/validateConnection

Command

POST

Command output format

application/json

See the following notes:

- If you are setting up multiple LDAP connections, you must individually connect to each directory.
- In the curl command, you must use a base64-encoded password in the **"LDAP_BINDPASSWORD"** parameter. To encode the password, use the following command:

  ```
  echo -n <password> | base64
  ```

  Following is an example output:

  ```
  UGFzc3cwcmQ=
  ```

Sample curl command for validating LDAP configuration while creating an LDAP connection

```
curl -k -X POST 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/ldap/vali
dateConnection' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
'{
    "name": "openLDAP",
    "idp_config": {
        "ldap_url": "ldap://9.35.210.15:389",
        "ldap_basedn": "dc=ibm,dc=com",
        "ldap_binddn": "cn=admin,dc=ibm,dc=com",
        "ldap_bindpassword": "HNJat5opcqR="
    }
}'
```

Sample response - validate LDAP connection while creating

```
{"status":"success","result":"Successfully validated LDAP configuration."}%
```

Validating LDAP configuration while updating the LDAP connection by using IdP V3 API

The following example shows how to validate the LDAP connection while you update an LDAP connection by using IdP V3 API:

API details - validate LDAP connection while updating - POST

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

`Cluster port`

Path

`idprovider/v3/auth/idsource/ldap/validateConnection`

Command

`POST`

Command output format

`application/json`

See the following notes:

- The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3 and Schema elements for LDAP connection.
- In the curl command, you must use a base64-encoded password in the **"LDAP_BINDPASSWORD"** parameter. To encode the password, use the following command:

  ```
  echo -n <password> | base64
  ```

  Example output:

  ```
  UGFzc3cwcmQ=
  ```

Sample curl command for validating LDAP configuration while you update an LDAP connection

```
curl -k -X POST 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/ldap/vali
dateConnection' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
'{
    "name": "openLDAP",
    "idp_config": {
        "ldap_url": "ldap://9.35.210.15:389",
        "ldap_basedn": "dc=ibm,dc=com",
        "ldap_binddn": "cn=admin,dc=ibm,dc=com"
    }
}'
```

Sample response - validate LDAP connection while updating

```
{"status":"success","result":"Successfully validated LDAP configuration."}%
```

Registering an LDAP connection by using IdP V3 API

See the following notes:

- The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3 and Schema elements for LDAP connection.
- In the curl command, you must use a base64-encoded password in the **"LDAP_BINDPASSWORD"** parameter. To encode the password, use the following command:

```
echo -n <password> | base64
```

Example output:

```
UGFzc3cwcmQ=
```

The following example shows how to register an LDAP connection by using IdP V3 API:

API details - register an LDAP connection - POST

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/

Command

POST

Command output format

application/json

The following sample curl command is for creating a custom LDAP connection. Replace the **"type"** parameter value with your LDAP type. For more information about the supported LDAP types, see Configuring LDAP connection.

You must also replace the parameter values in the **"idp_config"** section based on your LDAP type. For more information about the parameter values for each LDAP type, see Default LDAP filters by LDAP type.

Sample curl command for creating a custom LDAP connection

```
curl -k -X POST 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
'{
  "name": "openldap",
  "description": "",
  "protocol": "ldap",
  "type": "Custom",
  "idp_config": {
        "ldap_id": "openldap",
        "ldap_realm": "REALM",
        "ldap_url": "ldap://9.30.253.13:389",
        "ldap_basedn": "dc=ibm,dc=com",
```

```
        "ldap_binddn": "cn=admin,dc=ibm,dc=com",
        "ldap_bindpassword": "UGFzc3cwcmQ=",
        "ldap_type": "Custom",
        "ldap_ignorecase": "false",
        "ldap_userfilter": "(&(uid=%v)(objectclass=person))",
        "ldap_useridmap": "*:uid",
        "ldap_groupfilter": "(&(cn=%v)(objectclass=groupOfUniqueNames))",
        "ldap_groupidmap": "*:cn",
        "ldap_groupmemberidmap": "groupOfUniqueNames:uniqueMember",
        "ldap_nestedsearch": "false",
        "ldap_pagingsearch": "false"
        }
}'
```

Getting information about all LDAP connections

The following example shows how to get the list of all the registered LDAP directories:

API details - information about all LDAP connections - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource?protocol=ldap

Command

GET

Command output format

application/json

Note: The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3 and Schema elements for LDAP connection.

Sample curl command to get the list of registered LDAP connections

```
curl -k -X GET 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/?
protocol=ldap' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - information about all LDAP connections

```
{"id":"8967e54c-7f6f-4755-a944-
cf4d2a4b8a64","LDAP_ID":"openldap","LDAP_REALM":"REALM","LDAP_HOST":"corp.abc.com"
,"LDAP_PORT":"389","LDAP_IGNORECASE":"false","LDAP_BASEDN":"dc=ibm,dc=com","LDAP_B
INDDN":"cn=admin,dc=ibm,dc=com","LDAP_TYPE":"Custom","LDAP_USERFILTER":"(&(uid=%v)
(objectclass=person))","LDAP_GROUPFILTER":"(&(cn=%v)
(objectclass=groupOfUniqueNames))","LDAP_USERIDMAP":"*:uid","LDAP_GROUPIDMAP":"*:c
n","LDAP_GROUPMEMBERIDMAP":"groupOfUniqueNames:uniquemember","LDAP_URL":"ldap://co
rp.abc.com:389","LDAP_PROTOCOL":"ldap"}
```


Getting information about LDAP connection by using UID

The following example shows how to get the registered LDAP directory by UID:

API details - LDAP information by UID - GET

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

GET

Command output format

application/json

Note: The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3 and Schema elements for LDAP connection.

Sample curl command to get the list of registered LDAP connections by UID

```
curl -k -X GET 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/:UID' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - LDAP information by UID


```
{"id":"8967e54c-7f6f-4755-a944-
cf4d2a4b8a64","LDAP_ID":"openldap","LDAP_REALM":"REALM","LDAP_HOST":"corp.abc.com"
,"LDAP_PORT":"389","LDAP_IGNORECASE":"false","LDAP_BASEDN":"dc=ibm,dc=com","LDAP_B
```

```
INDDN":"cn=admin,dc=ibm,dc=com","LDAP_TYPE":"Custom","LDAP_USERFILTER":"(&(uid=%v)
(objectclass=person))","LDAP_GROUPFILTER":"(&(cn=%v)
(objectclass=groupOfUniqueNames))","LDAP_USERIDMAP":"*:uid","LDAP_GROUPIDMAP":"*:c
n","LDAP_GROUPMEMBERIDMAP":"groupOfUniqueNames:uniquemember","LDAP_URL":"ldap://co
rp.abc.com:389","LDAP_PROTOCOL":"ldap"}
```

Updating an LDAP connection by using IdP V3 API

See the following notes:

- Use **`UID`** to update the LDAP connection. To get the UID, see Getting information about LDAP connection by using UID.
- The curl command contains different schema elements. To understand the use of these schema elements, see Different schema elements for IdP V3 and Schema elements for LDAP connection.
- In the curl command, you must use a base64-encoded password in the **`"LDAP_BINDPASSWORD"`** parameter. To encode the password, use the following command:

  ```
  echo -n <password> | base64
  ```

  The following is an example output:

  ```
  UGFzc3cwcmQ=
  ```

The following example shows how to update an LDAP connection by using IdP V3 API:

API details - update an LDAP connection - PUT

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

PUT

Command output format

application/json

Sample curl command to update an LDAP connection

```
curl -k -X PUT 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/8967e54c-
7f6f-4755-a944-cf4d2a4b8a64' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
'{
  "name": "openldap",
  "description": "",
  "protocol": "ldap",
  "type": "Custom",
  "idp_config": {
        "ldap_id": "openldap",
        "ldap_realm": "REALM",
        "ldap_url": "ldap://9.35.210.15:389",
        "ldap_host": "9.35.210.15",
        "ldap_port": "389",
        "ldap_protocol": "ldap",
        "ldap_basedn": "dc=ibm,dc=com",
        "ldap_binddn": "cn=admin,dc=ibm,dc=com",
        "ldap_bindpassword": "UGFzc3cwcmQ=",
        "ldap_type": "Custom",
        "ldap_ignorecase": "false",
        "ldap_userfilter": "(&(uid=%v)(objectclass=person))",
        "ldap_useridmap": "*:uid",
        "ldap_groupfilter": "(&(cn=%v)(objectclass=groupOfUniqueNames))",
        "ldap_groupidmap": "*:cn",
        "ldap_groupmemberidmap": "groupOfUniqueNames:uniqueMember",
        "ldap_nestedsearch": "false",
        "ldap_pagingsearch": "false"
        }
}'
```

Sample response - update an LDAP connection

```
{"status":"success","message":"{8967e54c-7f6f-4755-a944-cf4d2a4b8a64} is
Updated."}
```

Deleting an LDAP connection by using IdP V3 API

The following example shows how to delete an LDAP connection by using IdP V3 API:

API details - delete an LDAP connection - DELETE

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/:UID

Command

DELETE

Command output format

application/json

Note: To get the UID, see Getting information about LDAP connection by using UID.

Sample curl command to delete an LDAP connection

```
curl -k -X DELETE 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/8967e54c-
7f6f-4755-a944-cf4d2a4b8a64' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
```

Sample response - delete an LDAP connection

```
{"status":"success","message":"{8967e54c-7f6f-4755-a944-cf4d2a4b8a64} is deleted"}
```

Disabling OpenShift authentication by using IdP V3 API

The following example shows how to disable Red Hat OpenShift authentication by using IdP V3 API:

API details - disable OpenShift authentication - PUT

API version

1.0.0

API URI components

Scheme

HTTPS

Host IP

Cluster address

Port number

Cluster port

Path

idprovider/v3/auth/idsource/osauth

Command

PUT

Command output format

```
application/json
```

Sample curl command to disable Red Hat OpenShift authentication

```
curl -k -X PUT 'https://cp-
console.apps.mycluster.mydomain.fyre.ibm.com/idprovider/v3/auth/idsource/osauth' \
--header 'Content-Type: application/json' \
--header "Authorization: Bearer $ACCESS_TOKEN"
--data-raw
'{
    "uid": "osauth",
    "description": "OpenShift Authentication enablement",
    "enabled": "false",
    "idp_config": {
        "ROKS_URL": "https://oauth-openshift.apps.cp3custom.cp.fyre.ibm.com",
        "ROKS_USER_PREFIX": ""
    },
    "name": "OpenShift",
    "protocol": "oauth",
    "type": "OpenShiftAuthentication"
}'
```

Sample response - disable OpenShift authentication

```
{
    "status": "success",
    "message": "{osauth} is Updated."
}
```

Getting the host details

1. Install OC client and connect to the Red Hat® OpenShift® Container Platform cluster. By default, Red Hat® OpenShift® Container Platform is enabled automatically once you install foundational services.
2. Get the hostname from the route available in the namespace where the foundational services is installed.

   ```
   oc get route -n <your-foundational services-namespace> cp-console -o
   jsonpath='{.spec.host}' && echo
   ```

   Sample output:

   ```
   cp-console.apps.mycluster.mydomain.com
   ```

Different schema elements for IdP V3 APIs

See the following mandatory fields and their description:

- `name`: Name of the IdP that you want to register.
- `description`: Short description of the registration.
- `protocol`: Type of protocol. Currently, OIDC, SAML, and LDAP are the only supported protocols.
- `type`: Type of configured LDAP, OIDC, or SAML IdP. For example, Google, Microsoft, IBM ISV, and Okta for OIDC; `isv` and `default` for SAML.

- **`idp_config`**: Contains the following subfields:
  - OIDC has the following fields:
    - **`discovery_url`**: A well-known endpoint by OIDC provider.
    - **`client_id`**: A client ID that is provided by the registered application in the IdP.
    - **`client_secret`**: A client secret that is provided by the registered application in the IdP.
  - SAML has the following fields:
    - **`idp_metadata`**: A base64-encoded value of IdP metadata XML file. The maximum accepted size of the metadata file is **`2 MB`**. If the size of the metadata file is more than 2 MB, an error message is displayed during the SAML registration.

    - **`token_attribute_mappings`**: Used to map the token attributes with the IdP user attributes You need to replace only the **`values_to_be_mapped`** based on your IDP configuration at the backend.

      - Sample mapping:

        ```
        "token_attribute_mappings": {
                "groups": "values_to_be_mapped",
                "given_name": "values_to_be_mapped",
                "family_name": "values_to_be_mapped",
                "sub": "values_to_be_mapped",
                "email": "values_to_be_mapped"
                }
        ```

        The mapped values are case-sensitive.

Schema elements for SAML with SCIM dependency

See the following schema elements:

- **`scim_config`**: Used to provide SCIM configuration and SCIM attribute mapping information.

  - **`grant_type`** - Identifies the token generation flow. The **`client_credentials`** value is used to issue the tokens to access the SCIM APIs.

  - **`token_url`** - Token endpoint of IdP. It is used to request the token.

  - **`client_id`** - Used for the client credentials flow.

  - **`client_secret`** - Used for the client credentials flow.

Example schema

```
"scim_config": {
  "scim_base_path": "https://bedrock-iam.verify.ibm.com/v2.0/",
  "grant_type": "client_credentials",
  "token_url": "https://bedrock-iam.verify.ibm.com/v1.0/endpoint/default/token",
  "client_id": "Your_isv_client_id",
  "client_secret": "Your_isv_client_secret",
  "scim_attribute_mappings": {
    "user": {
      "principalName": "userName",
      "givenName": "name.givenName",
      "middleName": "name.middleName",
      "familyName": "name.familyName",
      "formatted": "name.formatted"
    },
    "group": {
      "principalName": "displayName",
```

```
      "created": "meta.created",
      "lastModified": "meta.lastModified"
    }
  }
}
```

Schema elements for SAML with LDAP dependency

ldap_config is used to provide LDAP dependency information. See the following example:

```
"ldap_config":
{
  "ldap_id": "existing-ldap-connection-name"
}
```

Schema elements for LDAP connection

To know more about LDAP parameters that are used in Identity Provider APIs, see LDAP authentication, LDAP filters, and Default LDAP filters by LDAP type.

- **LDAP_PAGINGSEARCH**: If the **LDAP_PAGINGSEARCH** parameter is set as **"false"**, it indicates that the pagination option is disabled. Set this parameter to **"true"** to enable the pagination option.
- **LDAP_PAGING_SIZE**: It is a numeric field that defines the page size of the LDAP type. You can set the number as per your requirement. It is an optional field. If you do not provide the value to the **LDAP_PAGING_SIZE** parameter, the default page size of the LDAP server is considered as a request in the API query.

  For example, when the LDAP type is Microsoft Active Directory and you do not provide the **LDAP_PAGING_SIZE** value in the API, then 1000 (default page size) value is considered as a request in the API. Also, the **LDAP_PAGING_SIZE** value must not exceed the default server-side page limit or the maximum configured value for the respective LDAP type.

Issue resolution API

The issue resolution API allows for the interaction with alerts, incidents and related artifacts.

Note: The API reference (Swagger) document for the issue resolution API is not interactive and is published for your understanding of the different APIs and schemas available. For more information about how to access these APIs, review the following commands and procedure.

Endpoint URLs

Issue resolution events API

```
$ROUTE/aiops/api/issue-resolution/v1/events/
```

Issue resolution alerts API

```
$ROUTE/aiops/api/issue-resolution/v1/alerts/
```

Issue resolution incidents API

```
$ROUTE/aiops/api/issue-resolution/v1/incidents
```

For more information about utilizing Issue resolution incidents API to enrich an incident by using IBM Tivoli Netcool/Impact policy, see Enrich an incident using policies.

Authentication

The API uses Platform UI (Zen) authentication, so you either need a JWT token or ZenApiKey to be authenticated. The token or key is used to determine whether a user or service ID has access when you use the API.

Accessing the API with cURL

1. Log in to your cluster and authenticate your admin user credentials for running API commands:
    1. Adjust the project (namespace) for your system:

        ```
        NAMESPACE=cp4aiops
        ```

    2. Get the admin username and set it as an environment variable by running the following command

        ```
        USER=$(oc -n $NAMESPACE get secret platform-auth-idp-credentials -o jsonpath='{.data.admin_username}' | base64 -d)
        ```

    3. Get the password for your admin username and set it as an environment variable by running the following command:

        ```
        PASS=$(oc -n $NAMESPACE get secret platform-auth-idp-credentials -o jsonpath='{.data.admin_password}' | base64 -d)
        ```

    Note: Two methods of authentication are available:

    - JWT Bearer Token

      This token is temporary and lasts about 4 hours.

    - Platform UI (ZenApiKey) API key

      This key remains until deleted.

    For more information about API authentication, see Accessing APIs

2. Set your authentication.

    - If you are using the Platform UI (ZenApiKey) API key option, encode your admin username and API key with Base64 encoding:

      ```
      ZENAPIKEY=$(echo "<username>:<API_key>" | base64 -w 0); echo $ZENAPIKEY
      ```

    - If you are using the JWT Bearer Token option, set the token as an environment variable. The following command uses variables that are set in the previous step:

      ```
      TOKEN=$(curl -k -X POST https://$ROUTE/icp4d-api/v1/authorize -H "Content-Type: application/json" -d "{\"username\": \"$USER\",\"password\": \"$PASS\"}" 2>/dev/null|awk -F'"' '{print $(NF-1)}')
      ```

3. Run a command to access the API. For example, the following command uses cURL to access the issue resolution alerts API to load alert data:

    - By using the route and the Platform UI (ZenApiKey) API key option:

      ```
      curl -k -v -X POST "https://${ROUTE}/aiops/api/issue-resolution/v1/alerts/" --header "Content-Type: application/json" --header "Authorization: ZenApiKey ${ZENAPIKEY}" --header 'X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255' --data @<FILEPATH/FILE>.json
      ```

      Where **<FILEPATH/FILE>** is the full path to the JSON file that includes your sample alert data to load.

    - By using the route and the JWT Bearer Token option:

      ```
      curl -k -v -X POST "https://${ROUTE}/aiops/api/issue-resolution/v1/alerts/" --header "Content-Type: application/json" --header "Authorization: Bearer ${TOKEN}" --header 'X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255' --data @<FILEPATH/FILE>.json
      ```

      Where **<FILEPATH/FILE>** is the full path to the JSON file that includes your sample alert data to load.

Enabling access to the Probable cause customization APIs

The Probable cause API ranks the alerts in an incident. This API uses a classifier that uses Natural Language Processing (NLP) to determine the meaning of an alert and calculates the topology path between alerts to create a ranking for each applicable alert.

Probable cause can be computed by the following two approaches:

- Word based approach is very useful when the environment does not have topological data. This approach works in partnership with the topological probable causes when the topological data is available. Word based approach allows the users to define keywords that are given an importance score. If a word is in the summary field, the alert is given that score (the sum of all words).
- With the severity based approach, the higher severity alert is assigned a larger score. A higher severity alert does not always mean that an alert is a root cause. Higher severity alert generally gives more context to the operator about the impact. These alerts within a large group is important so that an operator can get context quickly.

You can use the Probable cause API to customize the analytics for probable cause, and add edge configurations when new custom edge types are added to IBM Cloud Pak for AIOps.

Note: If the topological data includes a user-defined edge, then the probable cause API does not understand the relationships in the topology. You can see improved results when an edge type is added to the system.

The Probable cause customization APIs are exposed by default, but to access them you must generate the token first. The API uses the IBM Cloud Pak Platform UI authentication, so you need a JWT (JSON Web Token) to be authenticated. The token is used to determine whether a user or service ID has access when they use the API.

Probable Cause Customization APIs

1. Log in to the cluster.
2. Use the admin-user-details secret to set PASS.

   ```
   PASS=$(oc get secret admin-user-details -o
   jsonpath='{.data.initial_admin_password}' -n cp4aiops | base64 -d)
   ```

3. Create the TOKEN.

   ```
   TOKEN=$(curl -s -k -X POST https://$ROUTE/icp4d-api/v1/authorize -H 'Content-
   Type: application/json' -d '{"username": "admin","password": <PASS>}' | jq
   .token | sed 's/"//g')
   ```

   Where **PASS** is the password obtained in the preceding step.

Example of accessing the APIs

You can now access the APIs, for example:

`/customisation/edge_type`

Use a GET request **`/customisation/edge_type`** to get the list of edges in the system.

```
curl -k -X GET --header 'Accept: application/json' -H "Authorization: Bearer
${TOKEN}" https://$ROUTE/aiops/api/issue-
resolution/mime/v1/customisation/edge_type -H "accept: application/json" -H "X-
TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255"
```

The label and edge names are the same as those added in the topological resource group. Some edges have different weight due to directorial. Overall, an **`indegree`** edge has a higher weight than an **`outdegree`** edge.

Note: If the system encounters an **`unknown`** edge type, then it uses the average of weight in the edge label group.

The following example shows a response(200) for the GET request:

```
{
"edge_type_list": [
   {
      "edge": "string",
      "label": "AGGREGATION",
      "outdegree": 0,
      "indegree": 0
   }
]
}
```

Use a POST request **`/customisation/edge_type`** to overwrite the complete list of edge types.

```
curl -k -X POST -H "Content-Type: application/json" -H "Authorization: Bearer
${TOKEN}" https://$ROUTE/aiops/api/issue-
resolution/mime/v1/customisation/edge_type -H "X-TenantID: cfd95b7e-3bc7-4006-
a4a8-a73a79c71255" -d '{"edge_type_list":[{ "edge": "newEdgeName", "label":
"association", "outdegree": 20,"indegree": 5}]}'
```

The following example shows a response(200) for the POST request:

```
{"Result":"Success"}
```

After the path score is calculated, the summary is classified in one of the following 8 categories.

Table 1. Label and Weight relationship for classification of summary

Label	Weight
errorrate	1.4
exception	1.2
information	1
latency	2
saturation	2.2
stateChange	1.8
traffic	2.4
unknown	-1

In order to modify the label weight, use the API **`/customisation/label_weight`**.

`/customisation/label_weight`

Use a POST request **`/customisation/label_weight`** to add or update label weight. The request body is an array of label weight objects.

```
curl -k -X POST -H "Content-Type: application/json" -H "Authorization: Bearer
${TOKEN}" https://$ROUTE/aiops/api/issue-
resolution/mime/v1/customisation/label_weight -H "X-TenantID: cfd95b7e-3bc7-4006-
a4a8-a73a79c71255" -d '{"label_weight_list":
[{"label":"association","weight":1.2}]}'
```

The following example shows a response(200) for the POST request:

```
{"Result":"Success"}
```

Note: Before removing a label from the list, remove all of the training examples from the classifier. The classifier looks at the word in the summary and attempts to put the alert in the in the appropriate category.

`/customisation/words`

Use a GET request **`/customisation/words`** to get the list of words in alert summary that is used by probable cause for the specified tenant ID. If the GET request is successfull(200 response), the response includes an array of **`words`** for the alert.

```
curl -k -X GET --header 'Accept: application/json' -H "Authorization: Bearer
${TOKEN}" https://$ROUTE/aiops/api/issue-resolution/mime/v1/customisation/words -H
"accept: application/json" -H "X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255"
```

The following example shows a response(200) for the GET request:

```
{
   "words":[
      {
         "word":"error",
         "caseSenstive":false,
         "weight":100
      },
      {
         "word":"ECC",
         "caseSenstive":true,
         "weight":10
```

```
      }
   ]
}
```

If both words "error" and "ECC" are found in same alert summary then the weight(score) is added. For example, if **error** has a weight of 100 and **ECC** has a weight of 10, then the total word score is 110. The higher the score, the more likely that this alert is a root cause.

Use a POST request **/customisation/words** to update(replace) the words in the system. The request body is the JSON array of words for probable cause.

```
curl -k -X POST --header 'Accept: application/json' -H "Authorization: Bearer
${TOKEN}" -H "Content-Type: application/json" https://$ROUTE/aiops/api/issue-
resolution/mime/v1/customisation/words -H "accept: application/json" -H "X-
TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255" -d '{"words":
[{"word":"error","caseSenstive": false, "weight":100},
{"word":"ECC","caseSenstive": true, "weight":10}]}'
```

The following example shows a response(200) for the POST request:

```
{"Result":"Success"}
```

/customisation/severity

Use a GET request **customisation/severity** to get the list of severities that is used by probable cause for the specified tenant ID. Based on the severity of the alert, the weight is added to the word score. Refer to the following table for more details:

Table 2. Severity and Weight relationship

Severity	Weight
1	10
2	20
3	30
4	40
5	50
6	60

If the word score is 110(Error and ECC from previous example) and the severity of the alert is critical(which is 6), then 60 is added to the score, resulting in a word score of 170.

```
curl -k -X GET --header 'Accept: application/json' -H "Authorization: Bearer
${TOKEN}" https://$ROUTE/aiops/api/issue-resolution/mime/v1/customisation/severity
-H "accept: application/json" -H "X-TenantID: cfd95b7e-3bc7-4006-a4a8-
a73a79c71255"
```

The following example shows a response(200) for the GET request:

```
{
   "severities":[
      {
         "severity":2,
         "weight":20
      }
   ]
}
```

Use a POST request **/customisation/severity** to update(replace) the probable cause to use the array of severities. The request body is JSON array of severities which will overwrite the current array of severities.

The following example shows a response(200) for the POST request:

```
{"Result":"Success"}
```

Use a DELETE request **/customisation/severity** to delete all the severities for the specified tenant ID.

The following example shows a response(200) for the DELETE request:

```
{"Result":"Success"}
```

/customisation/scoring/config

Use a GET request **customisation/scoring/config** to get the scoring configuration parameters for the specified tenant ID. If the API request is successfull(200 response), the response includes a list of the scoring configuration attributes that are turned on or off.

```
curl -k -X GET --header 'Accept: application/json' -H "Authorization: Bearer
${TOKEN}" https://$ROUTE/aiops/api/issue-
resolution/mime/v1/customisation/scoring/config -H "accept: application/json" -H
"X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255"
```

The following example shows a response(200) for the GET request:

```
{
"pathCaculationEnabled": true,
"wordEnabled": true,
"severityEnabled": true,
"causeScoreEnabled": true,
"itnmproccessing": true,
"itnmCausesWeight": 110,
"firstBoost" : true,
"firstBoostWeight" : 100,
}
```

Use a POST request **customisation/scoring/config** to alter the scoring configuration parameters for the specified tenant ID.

The following example shows a response(200) for the POST request:

```
{"Result":"Success"}
```

For more information about probable cause ranking and probable cause determination, see Probable cause ranking

Next steps

For more information about probable cause, see Probable cause in ChatOps.

For more information about the available APIs and related Swagger documentation, see the additional topics under APIs. For example, for more on access tokens, see Accessing APIs.

IBM Cloud Pak for AIOps Change Risk REST API

This documentation is for the IBM Cloud Pak for AIOps Change Risk REST API

Version: 1.0.0

Endpoint Root URL

Change Risk REST API root URL is as follows:

```
$ROUTE/aiops/api/ai-manager/change-risk/v1
```

/change_requests/risk_assessment

POST

Description

Get the risk assessment by providing a change request ticket information.

Parameters

- include_explanation : true or false for turning on/off explanation. Explanation is included by default.

Responses

Code	Description
200	Response
400	Bad request
404	Not found
500	Internal server error
504	Internal processing timeout
default	unexpected error

Example

The sample curl command resembles the following code:

```
CLUSTER_URL=https://cpd-cp4waiops.apps.mycluster.mydomain.myco.com/aiops/api/ai-
manager/change-risk/v1
curl -X POST $CLUSTER_URL/change_requests/risk_assessment?include_explanation=true
\
-H "Authorization: Bearer $ACCESS_TOKEN" \
-H 'Content-Type: application/json' \
--data '{
   "number":"CHG0775062",
   "sys_domain":"crn:v1:bluemix:public::mil01::::",
   "business_service":"Network",
   "short_description":"is DATACENTER 2020-05-17 08:29:00 UTC",
   "description":"description text field",
   "justification":"none",
   "backout_plan":".",
   "impact":"No",
   "reason":"Replace server due to hardware issue.",
   "state":"Closed",
   "type":"softlayer",
   "sys_created_by":"SoftLayer Jira.SA",
   "assigned_to":"name@email.com",
   "close_code":"successful",
   "close_notes":"Change was completed successfully"
}'
```

The preceding data field must validate against a change record schema. The response will look similar to the following:


```
{
   "id":"77f3f724727811eb9e81c8e0eb1474cb",
   "prediction_text":"Prior ticket data insights classify this as a high risk
change.",
   "prediction_confidence":0.979616,
   "explanation":{
      "words":{
         "test":0.1,
         "run":0.073,
         "pod":0.059,
         "node":0.056,
         "error":0.041,
         "release":0.036,
         "remove":0.027,
         "backout":0.025,
         "fail":0.023,
         "plan":0.017,
         "utc":0.017,
         "system":0.013,
         "previous":0.012
      },
      "relevant_change_tickets":[
         {
            "number":"CHG0675192",
            "sys_domain":"crn:v1:bluemix:public::mil01::::",
            "business_service":"Network",
            "short_description":"is DATACENTER 2020-05-17 08:29:00 UTC",
            "description":"description text field",
            "justification":"none",
            "backout_plan":".",
            "impact":"No",
            "reason":"Replace server due to hardware issue.",
            "state":"Closed",
            "type":"softlayer",
            "sys_created_by":"SoftLayer Jira.SA",
            "assigned_to":"name@email.com",
            "close_code":"successful",
            "close_notes":"Change was completed successfully"
            "induced_incidents":[
               {
                  "description":"description text",
                  "number":"INC2468135",
                  "severity":"1 - High"
               }
            ]
         }
      ],
      "ticket_relevance_level":{
         "CHG0675192":10.0
      },
      "ticket_similarity_level":{
         "CHG0675192":1.0
      }
   }
}
```


/feedback/types/risk_assessment

POST

Description

Provide feedback for the corresponding change risk assessment

Responses

Code	Description
200	Response
400	Bad request
404	Not found
500	Internal server error
default	unexpected error

/feedback/types/risk_assessment_explanation

POST

Description

Provide feedback for the corresponding change risk assessment

Responses

Code	Description
200	Risk assessment response
400	Bad request
404	Not found
500	Internal server error
default	unexpected error

/healthcheck

GET

Summary

Gets the health of the server.

Responses

Code	Description
200	OK
500	Internal server error

Models

Change

Name	Type	Description	Required
number	string		Yes

Name	Type	Description	Required
sys_domain	string		No
business_service	string		No
short_description	string		No
description	string		No
justification	string		No
backout_plan	string		No
impact	string		No
reason	string		No
state	string		No
type	string		No
sys_created_by	string		No
assigned_to	string		Yes
close_code	string		No
close_notes	string		No

Incident

Name	Type	Description	Required
number	string		Yes
description	string		Yes
severity	string		No

Feedback

Name	Type	Description	Required
user_response	string		Yes

FeedbackRiskAssessment

Name	Type	Description	Required
assessment_id	string		Yes
feedback	object		Yes

FeedbackRiskAssessmentExplanation

Name	Type	Description	Required
assessment_id	string		Yes
relevant_ticket_number	string		Yes
feedback	object		Yes

RiskAssessment

Name	Type	Description	Required
number	string		No
id	string		Yes
prediction_text	string		Yes
prediction_confidence	float		Yes

Name	Type	Description	Required
change_risk_score	float		Yes
explanation	object		Yes

Explanation

Name	Type	Description	Required
words	object		Yes
relevant_tickets	[object & Change]		No
ticket_relevance_level	object		No
ticket_similarity_level	object		No

RelevantTickets

Name	Type	Description	Required
RelevantTickets	array		

ErrorContainer

Name	Type	Description	Required
errors	[object]		No
trace	string	*Example:* `"9daee671-916a-4678-850b-10b911f0236d"`	No

Errors

Name	Type	Description	Required
Errors	array		

Error

Name	Type	Description	Required
http_code	integer	*Example:* `500`	No
code	string	*Example:* `"missing_field"`	No
message	string	*Example:* `"The id field is required."`	No
more_info	string		No

Metric REST API

The Metric API allows for the posting of metric data for anomaly detection and forecasting.

Note: The API reference (Swagger) document for the metric API is not interactive and is published for your understanding of the different APIs and schemas available. For more information about how to access these APIs, review the following commands and procedure.

Endpoint URLs

Metric Ingestion API

`$ROUTE/aiops/api/app/metric-api/v1/metrics`

Metric Retrieval API

`$ROUTE/aiops/api/app/metric-api/v1/metrics`

Metric Forecasting API

`$ROUTE/aiops/api/app/metric-api/v1/forecast`

Authentication

The API uses Platform UI (Zen) authentication, so you either need a JWT token or ZenApiKey to be authenticated. The token or key is used to determine whether a user or service ID has access when you use the API.

Error handling

Metric Ingestion API error handling

Table. Metric ingestion API error codes

HTTP Error Code	Description	Recovery
`200`	Success	Success.
`400`	Bad Request	Invalid value in JSON.
`403`	Forbidden	The supplied authentication is not authorized to access '{namespace}'.
`500`	Internal Server Error	An error occurred processing your request.

Metric Retrieval API error handling

Table. Metric retrieval API error codes

HTTP Error Code	Description	Recovery
`200`	Success	Success.
`400`	Bad Request	Invalid parameters.
`403`	Forbidden	The supplied authentication is not authorized to access '{namespace}'.
`500`	Internal Server Error	An error occurred processing your request.

Metric Forecasting API error handling

Table. Metric forecasting API error codes

HTTP Error Code	Description	Recovery
`200`	Success	Success.

HTTP Error Code	Description	Recovery
204	No Content	Success, no data is found for provided attributes.
400	Bad Request	Invalid value in JSON.
401	Unauthorized	You are not authorized to make this request.
403	Forbidden	The supplied authentication is not authorized to access '{namespace}'.
500	Internal Server Error	An error occurred processing your request.

Ingest data

To inject data for metric analysis, you need to first log in to your cluster and authenticate for using APIs:

1. Log in to your cluster and authenticate your admin user credentials for running API commands:

    1. Adjust the project (namespace) for your system:

        ```
        NAMESPACE=cp4aiops
        ```

    2. Get the admin username and set it as an environment variable by running the following command

        ```
        USER=$(oc -n $NAMESPACE get secret platform-auth-idp-credentials -o jsonpath='{.data.admin_username}' | base64 -d)
        ```

    3. Get the password for your admin username and set it as an environment variable by running the following command:

        ```
        PASS=$(oc -n $NAMESPACE get secret platform-auth-idp-credentials -o jsonpath='{.data.admin_password}' | base64 -d)
        ```

    Note: Two methods of authentication are available:

    - JWT Bearer Token

      This token is temporary and lasts about 4 hours.

    - Platform UI (ZenApiKey) API key

      This key remains until deleted.

    For more information about API authentication, see Accessing APIs

2. Set your authentication.

    - If you are using the Platform UI (ZenApiKey) API key option, encode your admin username and API key with Base64 encoding:

      ```
      ZENAPIKEY=$(echo "<username>:<API_key>" | base64 -w 0); echo $ZENAPIKEY
      ```

    - If you are using the JWT Bearer Token option, set the token as an environment variable. The following command uses variables that are set in the previous step:

      ```
      TOKEN=$(curl -k -X POST https://$ROUTE/icp4d-api/v1/authorize -H 'Content-Type: application/json' -d "{\"username\":
      ```

```
\"$USER\",\"password\": \"$PASS\"}" 2>/dev/null|awk -F'"' '{print $(NF-1)}')
```

3. Run a command to load the training data file. As an example, the following command uses cURL to access the API.

   - By using the route and the Platform UI (ZenApiKey) API key option:

     ```
     curl -k -v -X POST "https://${ROUTE}/aiops/api/app/metric-api/v1/metrics" --header 'Content-Type: application/json' --header "Authorization: ZenApiKey ${ZENAPIKEY}" --header 'X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255' --data @<FILEPATH/FILE>.json
     ```

     Where **<FILEPATH/FILE>** is the full path to the JSON file that includes your training data to load.

   - By using the route and the JWT Bearer Token option:

     ```
     curl -k -v -X POST "https://${ROUTE}/aiops/api/app/metric-api/v1/metrics" --header 'Content-Type: application/json' --header "Authorization: Bearer ${TOKEN}" --header 'X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255' --data @<FILEPATH/FILE>.json
     ```

4. To ensure that data is injected, run a command to return data with the API. As an example, the following command uses cURL to access the API.

   - By using the route and the Platform UI (ZenApiKey) API key option:

     ```
     curl -v "https://${ROUTE}/aiops/api/app/metric-api/v1/metrics?resource=MyApp&group=Latency&metric=calls.sum" --header "Authorization: ZenApiKey ${ZENAPIKEY}" --header 'X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255' --insecure
     ```

   - By using the route and the JWT Bearer Token option:

     ```
     curl -v "https://${ROUTE}/aiops/api/app/metric-api/v1/metrics?resource=MyApp&group=Latency&metric=calls.sum" --header "Authorization: Bearer ${TOKEN}" --header 'X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255' --insecure
     ```

JSON schema for metrics data

JSON Payload:

```
{
    "groups": [
        {
            "timestamp": <timestamp>,
            "resourceID": "<resource ID>",
            "metrics": {
                "<kpi-1>": <value-1>,
                "<kpi-2>": <value-2>,
                ...
                "<kpi-n>": <value-n>
            },
            "attributes": {
                "group": "<metric namespace>",
                "node": "<top-level resource>"
            }
        }
```

```
    ]
}
```

- **`timestamp`** - The epoch time in milliseconds (13 digits) or ISO-8601.
- **`resourceID`** - The resource that is being measured.
- **`metrics`** - A map of metric name to metric value.
- **`attributes`** - Anything that helps provide context to the user.
- **`group`** - Field that you can use to namespace a metric. For example, "customer1". Therefore, group helps distinguish metrics names.
- **`node`** - The top-level resource.

Example JSON Payload:

```
{
    "groups": [
        {
            "timestamp": "1642516320000",
            "resourceID": "MyApp",
            "metrics": {
              "calls.sum": 476.0,
              "latency.mean": 1.6722689
            },
            "attributes": {
              "group": "Latency",
              "node": "MyApp"
            }
        }
    ]
}
```

Note: The **`attributes.group`**, **`resourceID`**, and the **`metrics`** (for example, **`metrics.access-denied`**) fields, along with the **`tenantID`** (part of the header in the HTTP request) make up the unique identifier for a KPI. Therefore, as part of the metric anomaly processing and detection, any data points with different values for these fields are considered as different metrics. For example, consider the following two data points:

```
{
  "groups": [
    {
      "timestamp": 1634846130000,
      "resourceID": "10.10.10.2",
      "metrics": {
        "access-denied": 14977
      },
      "attributes": {
        "group": "Org1",
        "node": "firewall-1"
      }
    },
    {
      "timestamp": 1634846130000,
      "resourceID": "10.10.10.2",
      "metrics": {
        "access-denied": 379
      },
      "attributes": {
        "group": "Org2",
        "node": "firewall-1"
      }
    }
  ]
}
```

These two data points have the same **timestamp** value (**1634846130000**) and **resourceID** value (**10.10.10.2**), but have two different group attribute names (**Org1** and **Org2**). This difference causes the data points to be treated as different metrics.

The JSON metric field for correlating a metric anomaly with an element in a resource group depends on the data source for the corresponding topology data. For example, for topology data that is ingested through an Instana connection, the following matching rule is used for correlation: **${name}:::${uniqueId}**. As another example, for topology data that is ingested through a file observer job, the **resourceID** field is used as the identifier for topology correlation.

Unpegging, accumulator and counter attributes: For metrics that count upwards only from when the system starts, the analytics wants to work on the delta values rather than the raw values.

- To identify which metrics are pegs, accumulators, or counters in metric anomaly detection, set an attribute per metric group, for example: `"accumulators":"TotalConnections,SuccessfulLoginCount"`
- Alternatively, you can set the attribute to **ALL** to apply to all metrics, for example: `"accumulators":"ALL"`

Aggregation attributes: Average is the default aggregation type, except for accumulators, where sum is the default.

- To identify which metrics are to be aggregated with sum in metric anomaly detection, set an attribute per metric group, for example: `"sumAggregator":"newConnections,Logins"`
- Alternatively, you can set the attribute to **ALL** to apply to all metrics, for example: `"sumAggregator":"ALL"`

Introduction

Runbook Automation is responsible for the management of runbooks and runbook instances. Runbooks are structured operational procedures which can be created, changed, and executed. They might include several types of automation and run manual, semi-automated, or fully-automated. Runbook instances are the execution records of Runbooks. They contain the status and details of one specific execution and can be changed to step through an execution and finish it. Additionally other administrative functions are presented in the API, including configuration management. To send API calls, create an API key in the application and find detailed API documentation inside the Swagger document.

Endpoint URLs

Runbook API

It is used to manage runbooks.

`$ROUTE/aiops/api/story-manager/rba/v1/runbooks`

Runbook instance API

It is used to manage runbook executions.

`$ROUTE/aiops/api/story-manager/rba/v1/runbookinstances`

Configuration API

It is used to manage Runbook Automation configuration.

`$ROUTE/aiops/api/story-manager/rba/v1/configuration`

Tag API

It is used to retrieve runbook tags.

`$ROUTE/aiops/api/story-manager/rba/v1/tags`

Usage Statistics API

It is used to retrieve Runbook Automation usage statistics.

`$ROUTE/aiops/api/story-manager/rba/v1/statistics/runbookinstances`

Status API

It is used to retrieve the status of the Runbook Automation deployment.

`$ROUTE/aiops/api/story-manager/rba/v1/status`

User API

It is used to retrieve a list of all users who have a given role.

`$ROUTE/aiops/api/story-manager/rba/v1/users`

Authentication

The API uses Zen authentication which requires a JWT token to be authenticated. The token is used to determine whether a user or service ID has access when they use the API.

Error handling

The Runbook Automation API returns one of the following HTTP responses for every API call.

HTTP Error Code	Description	Recovery
`200`	Success	The call was successful. See the output for the requested data.
`204`	No Content	Action completed successfully, no data is returned by the server.
`400`	Bad Request	The input was invalid. Read the error message and adjust your input before trying again.
`401`	Unauthorized	The request was not authorized. Make sure your API key is valid and the user and password are entered correctly.
`403`	Forbidden	The supplied authentication is not authorized to access the given API.
`500`	Internal Server Error	An internal error happened. A correlationId is provided. Include this ID in any request towards IBM support.

Application and topology APIs

IBM Cloud Pak® for AIOps offers a set of tools and APIs that you can use to predict, communicate, and resolve events before they become serious problems for your organization.

To learn more about how topologies are displayed in the UI, review the topology viewer documentation. For more information, see Topology viewer.

API authentication

For more information on how to authenticate to use APIs, see Accessing APIs.

Access to URL routes

For more information on how to access the URL routes for the topology Swagger documentation, see Enabling access to URL routes.

Access the topology service Swagger through port forwarding

- Accessing topology Swagger

Swagger documentation

Review the following Swagger documentation to use Topology APIs for configure File Observer and Rest Observer jobs.

- Topology File Observer API
- Topology Rest Observer API

To learn more about how to use these jobs, such as for including custom data in topologies, see:

- Configuring REST Observer jobs
- Configuring File Observer jobs

Troubleshooting: Swagger UI not responding during high-volume query

When querying for resources using **`GET /resources`**, the Swagger UI may become unresponsive if there are in excess of a thousand results.

- **Cause:** This is a Swagger limitation.
- **Workaround:** None. Avoid high-volume Swagger queries.

API reference documentation

To learn more about using the topology APIs, including terminology and properties, see Topology API reference.

Accessing Topology service Swagger UI

You can access the Topology service Swagger using port forwarding.

Before you begin

You need to be an administrator with **oc client** access in order to **get** secrets, services, and to enable port-forwarding in the ASM namespace.

About this task

A simple way to access the Topology service Swagger is to use the **oc port-forward** command. You can then access Swagger on your local browser through **localhost**. **Note:** Port forwarding will be enabled temporarily only. When there is no traffic, it will automatically be closed.

- To log on, use **oc login**
- To obtain the service name, use **oc get service**
- To enable port forwarding, use **oc port-forward service/aiops-topology-topology 8080**

Enable port forwarding

See the following steps as an example.

1. Log on using the **oc login** command.
2. Find the secret storing the credentials for Swagger using the **oc get secret | grep topology-asm-credentials** command.

   System output:

   ```
   aiops-topology-asm-credentials                                    Opaque
   2      3h54m
   ```

3. Find the topology service secret using the **oc get service | grep topology-topology** command.

   System output:

   ```
   aiops-topology-topology                                  ClusterIP   1.2.3.4
   <none>        8080/TCP,8081/TCP
   3h49m
   ```

4. Enable port forwarding using the **oc port-forward service/aiops-topology-topology 8080** command.

   System output:

   ```
   Forwarding from 127.0.0.1:8080 -> 8080
   Forwarding from [::1]:8080 -> 8080
   ```

   Note: If the port is not accessed for a while, the connection is closed:

   ```
   Forwarding from [::1]:8080 -> 8080
   E0622 07:57:52.101053   25681 portforward.go:233] lost connection to pod
   ```

5. Now access the topology service swagger api in a browser on the same host as the one the oc commands were run on, at the following URL **https://localhost:8080/1.0/topology/swagger**
6. Authorise using the credentials from the secret obtained in step 3.

Enabling access to URL routes for topology observer APIs

To access the URL routes for the topology Swagger documentation, you need to update the IBM ASM Operator configuration.

About this task

The URL to access an observer API consists of **{HOST}/{PATH}** and the Swagger UI URL consists of **{HOST}/{PATH}/swagger**.

Procedure

Complete the following configuration to expose all observers for both **full** and **listen** jobs.

1. Run the following command to edit the ASM operator configuration:

   ```
   oc edit asm aiops-topology
   ```

2. Search for the string **helmValuesASM:** within the **spec** section of the operator custom resource. Add **global.enableAllRoutes: true** to the list of values to enable the routes. The following example, shows this setting:

   ```
   spec:
     helmValuesASM:
       global.enableAllRoutes: true
   ```

3. Save your changes and wait for the ASM operator to reload.

4. If you have an existing installation of Cloud Pak for AIOps on Linux, run the following command:

   ```
   oc get ingress | grep topology
   ```

   Output:

   ```
   aiops-topology-topology-cp4aiops.apps.aiops.xxxxxx.xxxxxx.com /1.0/topology/
   ```

5. If you have an existing installation of Cloud Pak for AIOps on OpenShift Container Platform, run the following command. You can use the **oc get route | grep** command to view the route details for an observer API to check whether the topology routes are enabled:

   ```
   oc get route | grep topology
   ```

Examples

Example: View AppDynamics observer route details

The following example uses the AppDynamics observer:

```
oc get route | grep appdynamics
```

Output:

```
NAME                                                HOST/PORT
PATH                                SERVICES                                                PORT
TERMINATION          WILDCARD
aiops-topology-appdynamics-observer       aiops-topology-appdynamics-observer-
aiops.apps.myibm.com       /1.0/appdynamics-observer       aiops-topology-
appdynamics-observer       9148    reencrypt/Redirect   None
```

Example: cURL command to the AppDynamic Observer (or job) endpoint

```
curl -k -X GET --header 'Accept: application/json' -H "Authorization: Bearer
${TOKEN}" https://aiops-topology-appdynamics-observer-
aiops.apps.myibm.com/1.0/appdynamics-observer/jobs --header 'X-TenantID: cfd95b7e-
3bc7-4006-a4a8-a73a79c71255'
{"_items":[],"_limit":100}
```

Note: To obtain your access token, see Accessing APIs.

Example Swagger UI URL: `https://aiops-topology-appdynamics-observer-aiops.apps.myibm.com/1.0/appdynamics-observer/swagger`

Next steps

For more information on the available API and related Swagger documentation, see the additional topics under APIs.

Topology API reference

The following topics explain the common terminology, properties, types, and more for the available topology APIs.

Terminology

In addition to the following terms, ensure that you are familiar with the important terms that are defined within the Glossary.

resource: A resource is a node in an interconnected topology, sometimes also referred to as a vertex, or simply a node. It can be anything in a user-specific topology that is designated as such, for example a hardware or virtual device, a location, a user, or an application.

edge: An edge is a relationship between resources, also simply referred to as the 'link' between resources. Edges have a label, which allocates them to a family of edges with specific behavior. It also governs how they are displayed in the UI, and an edgeType, which defines the relationship in real terms.

tenant: A tenant is represented by a globally unique identifier, its tenant ID. The default tenant ID is: cfd95b7e-3bc7-4006-a4a8-a73a79c71255

provider: A provider is usually a single data source within the scope of a tenant. The provider is a namespace for the resource's uniqueId. That uniqueId property is unique only within the scope of a provider.

status: Status is a property of one or more resources, and a single resource can have different types of status.

Each status can be in one of three states:

- open

- clear
- closed

The severity can be one of the following ratings:

- clear
- information
- indeterminate
- warning
- minor
- major
- critical

The status of a resource is derived from events, or if the resource is retrieved through the status service. It can also be supplied when resources are posted to the topology service.

Reference documentation

Properties

The Topology Service has two categories of properties, generic and user. Generic properties have fixed data types, while user-defined properties do not.

Edge labels

The topology service defines a family of labels for the edges it supports.

Edge types

All edges created in the Topology Service should define an edge **type**. The following section lists the edge types that are associated with each of the public-facing edge labels. If none of the default edge types suffice, you can create custom edge types.

Entity types

Use the Topology Service to group together resources of the same type using the **`entityTypes`** property, and identify them in the UI by their icon. Usually a resource has only a single entity type but as the property is a *Set*, it is possible for a resource to have more than one entity type. A number of pre-defined entity types are supplied. In addition, you can create more entity types as required.

Status (and state)

Resources have specific statuses, and in turn each status has a state of open, clear or closed.

Timestamps

Both vertex and edge graph elements can have multiple timestamps, as are documented here.

Topology examples

You can review examples that describe best-practice guidelines to provide you with practical information, such as implementation examples and code samples, that you can use to get started.

Properties

The topology API has two categories of properties, generic and user. Generic properties have fixed data types, while user-defined properties do not.

- Some properties have special semantics, as described in **Generic properties**.
- The pre-defined properties are listed in the **Pre-defined properties** table.
- You can add any property as required, as described in **User properties**.

Generic properties

Generic properties are few and constrained to a fixed data type. They can also be subdivided into those which are read/write and those which are read-only.

- **`uniqueId`**

  The uniqueId is the string that is used to match resources from the same provider. It could be, for example, a UUID by which the provider can look up its own local data store for information about that device.

  If you send the same resource with the same uniqueId and the same provider more than once, it will be treated as the same resource.

  The provider is a project (namespace) for the resource's uniqueId. That uniqueId property is unique only within the scope of a provider.

- **`matchTokens`**

  These tokens are used to store strings, which are significant to that resource, and might match it to events.

- **`name`**

  This string does not have to be unique, and it should be short and memorable.

- **`tags`**

  Tags can be used to search for resources in the topology view, and to group resources. For more information on configuring tag rules and tag-based templates, see the following sections:

  - Defining rules
  - Group templates

- **`entityTypes`**

  These are defined as a set, though with usually only a single member, of one or more types this resource represents.

  For more information on entityTypes, see the Entity types topic.

Predefined properties

Table. Predefined properties

Name	Type	Cardinality	Alias	Read-only	Indexed
age	integer	single		no	y
aliasIds	ID	set	_aliasIds	yes	n
beginTime	long	single	_startedAt	yes	y
changeTime	long	single	_modifiedAt	yes	n

Name	Type	Cardinality	Alias	Read-only	Indexed
createTime	long	single	_createdAt	yes	y
deleteTime	long	single		yes	n
description	string	single		no	y
edgeTenantId	ID	single	_edgeTenantId	yes	n
edgeType	string	single	_edgeType	yes	n
endTime	long	single	_deletedAt	yes	y
entityTypes	string	set		no	y
eventId	string	single		yes	n
eventManager	string	single		yes	n
expireTime	long	single	_expiredAt	yes	n
geolocation	GeoLocation	single		no	n
hasState	string	single		yes	n
icon	string	single		no	n
id	long	single		yes	n
keyIndexName	string	single		yes	n
label	string	single		yes	n
matchTokens	string	set		no	y
name	string	single		no	y
observedTime	long	single	_observedAt	yes	n
operation	string	single		yes	n
partOfExternal	Boolean	single		yes	n
prevBeginTime	long	single		yes	y
providerId	ID	single		yes	n
providerName	string	single		yes	n
reconciliationTokens	string	set		yes	n
referenceId	ID	single		yes	y
referenceNo	long	single		yes	n
serializedHashMap	HashMap	single		yes	n
severity	string	single		no	n
speed	long	single		no	y
statusType	string	single		yes	n
tags	string	set		no	y
tenantIds	ID	set	_tenantIds	yes	y
uniqueId	string	single		no	y
uuid	ID	single	_id	yes	y
version	string	single		no	y
vertexType	string	single		yes	y

User properties

User-defined properties are free-form, and are not constrained by any data type. You can add new user properties as needed.

You can define any custom properties, such as, **`ipAddress`**.

Note: All user-defined properties such as **`ipAddress`**s are not in the generic set, and are stored as a serialized **`blob`** data type instead. The implication of this serialized data type storage convention is that these properties cannot be filtered, as they are incompatible with the **`_filter`** query parameter used in the REST API.

JSON record depicting both types of properties

The following picture depicts how the properties in a JSON record from an Observer can be split into both generic and user properties.

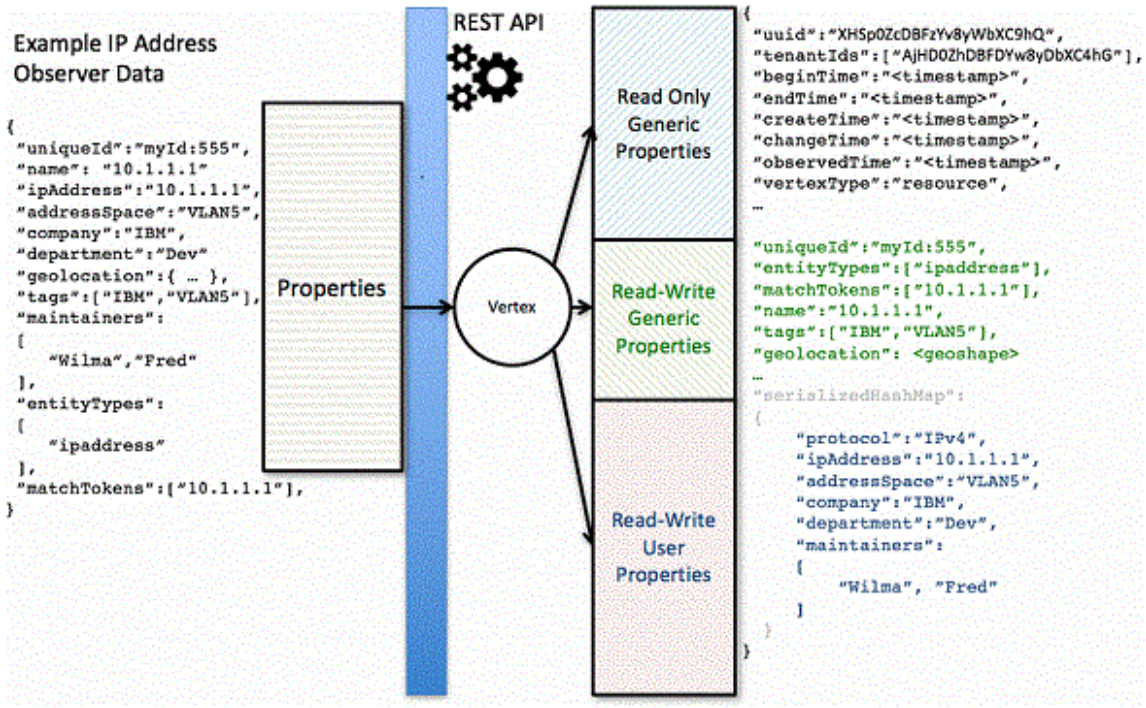


Figure. JSON record properties

Edge labels

The topology service defines a family of labels for the edges it supports.

Note: Most interactions with edges in the topology are with edge *types* rather than edge *labels*. These types can be conceptualized as instances of the edge label classes, and are documented separately. For more information, see Edge types.

- **`aggregation`**

  A relationship where the target vertex can exist independently of the source, but is needed for the source to function.

  A 'parent-child' relationship where the parent (source) is aggregating the children (target).

  The type of aggregation is determined by the value of the edge type.

Use this edge label to represent a resource in the UI that is composed of various elements, for example a book **`contains`** words.

- The direction is always from parent to child.
- Children can have multiple parents.

For more information on the types that associated with this label, see Edge types.

- **`association`**

  A 'weak' relationship where both source and target vertex can exist independently and each does not require the other to function.

  The specific type of association is determined by the edge type.

  Use this edge label to represent a general relationship between vertices in the UI, for example a person **`has`** a house.

  The label itself has no direction, but a direction could be implied by the edge **type** used.

  For more information on the types that associated with this label, see Edge types.

- **`composition`**

  A 'part-whole' relationship where the child (source) is a part of the parent (target).

  Use when you need to create a short-cut from a node to its root, bypassing fine-grained relationships, for example a component is **partOf** a device.

  - The direction is always child to parent.
  - Children cannot have multiple parents.
  - Composition relationships are hidden from the element-to-element view. They are used to construct the element-to-host view.
  - **partOf** is the only edge type you can use for the 'composition' label, and is reserved for its exclusive use.

- **`dataFlow`**

  A dataFlow label represents a data flow between a pair of vertices.

  The specific type of data flow is qualified by properties on the edge type.

  Use this label when you need to represent any form of data flow in the UI, for example a person emailing another person.

  The label itself has no direction, but a direction can be implied from the edge type used.

  For more information on the types that associated with this label, see Edge types.

- **`dependency`**

  A 'strong' relationship where the source depends on the target and cannot operate independently of it, and is not integral to it.

  The specific type of dependency is determined by the edgeType.

  Use this label when you need to represent the dependency of one resource on another in the UI, for example an application dependsOn a database.

  The direction is always from the dependent resource to the independent resource.

For more information on the types that associated with this label, see Edge types.

Edge types

All edges created in the topology should have an edge **type** defined. Review the following lists of the types that are associated with each edge label. If none of the default types suffice, you can create your own custom types.

Edge **types** can be thought of as being instances of the edge **label** classes. Most interactions with edges in the topology are with *types* rather than *labels*.

Edge types for labels

Types for the aggregation label

Edge type	Description	Example
contains	The source resource is considered to contain the target resource	Slot **`contains`** a card
federates	The source resource is a federation of the target resources	Database **`federates`** nodes
members	The source resource has the target resources as its members	Subnet **`members`** are IP addresses

Types for the association label

Edge type	Description	Example
aliasOf	Depicts that one resource is an alias of another; potentially from different providers	FQDN1 is an **`alias`** of FQDN2
assignedTo	When one resource is assigned to another	The alarm is **`assingedTo`** an operator
attachedTo	When one resource is attached to another	The plug is **`attachedTo`** to the cable
classifies	When one resource is used to classify another	The government **`classifies`** the document
configures	When one resource configures or provides configuration for another resource	The file **`configures`** the application
deployedTo	Describes when one resource is deployed to another resource	The application was **`deployedTo`** the server
exposes	When one resource exposes another	The application **`exposes`** the interface
has	Generalized relationship when one resource possesses, owns, or has another resource	Host **`has`** component
implements	When one resource implements another	The class **`implements`** the interface
locatedAt	When one resource is physically located in or at another resource's location	Host is **`locatedAt`** data center

Edge type	Description	Example
manages	When one resource manages another	The boss **manages** the employee
monitors	When one resource monitors another	The application **monitors** the host
movedTo	When one resource moves to a new and different resource	The service has **movedTo** the host
origin	Indicates the origin of a Vertex	Device's **origin** is a vendor
owns	Indicates ownership of one resource by another resource	The user **owns** the server
rates	Can be used when one resource rates another	The manager **rates** the employee
resolvesTo	When one resource resolves to another	The hostname **resolvesTo** an address
realizes	When one resource realizes another	The hypervisor **realizes** the virtual machine
segregates	When one resource segregates another	The firewall **segregates** the network
uses	When one resource takes, employs or deploys another resource as a means of achieving something	Application **uses** this disk

Types for the dataFlow label

Edge type	Description	Example
accessedVia	One resource is accessed through another, typically remote.	Server is **accessedVia** a FQDN
bindsTo	A network layering relationship such that the source runs on top of the target.	Logical interface **bindsTo** a physical port
communicatesWith	A relationship whereby one resource communicates with another	The sensor **communicatesWith** an application
connectedTo	A relationship whereby one resource is connected to another	The laptop is **connectedTo** the switch
downlinkTo	One resource is down-linked to another	The controller has a **downlinkTo** the sensor
reachableVia	A resource is reachable through another resource	The network is **reachableVia** the gateway
receives	A resource receives data from another	The Mail server **receives** an email
routes	A relationship whereby one resource routes data for another	The device **routes** the data
routesVia	A relationship whereby data from one resource routes through another	The traffic **routesVia** the device
loadBalances	One resource which balances load for others	The load balancer **loadBalances** to servers
resolved	When one resource resolved something for another	DNS server **resolved** IP address
resolves	Represents that one resource uses another to resolve it	FQDN **resolves** to the address
sends	A resource sends some data to another	The application **sends** an SMS message

Edge type	Description	Example
traverses	Describes when one resource traverses another	The message `traverses` the network
uplinkTo	One resource is up-linked to another	The sensor has an `uplinkTo` the controller

Types for the dependency label

Edge type	Description	Example
dependsOn	A generic dependency between resources	One application `dependsOn` another
runsOn	A resource runs on (and therefore depends on) another resource	The service `runsOn` the host

Custom edge types

If no default edge types suitably represents a relationship, you can define a custom edge type through the `https://localhost:8080/1.0/topology/types/edge` POST API.

Important: A custom edge type needs to be created **before** the observation job passes any edges with the type to the topology service.

- `edgeType`

  Required. The edgeType name, which must be unique and **cannot** match the edgeType name of a default edgeType, unless the edgeLabel also matches the corresponding default edge type's edgeLabel parameter.

  Restriction: A scenario where both edgeType and edgeLabel match the fields of a default edge type is equivalent to manually creating a default edge type, which is not necessary as default edge types are created implicitly by the topology when needed.

- `edgeLabel`

  Required. The edgeLabel of the custom edgeType, which must have one of the following labels:

  - dataFlow
  - dependency
  - association
  - aggregation

- `description`

  Optional, but recommended. This should be a meaningful description of the type of relationship this edge type represents.

Example

```
curl -k -X POST --header 'Content-Type: application/json' --header 'Accept:
application/json' --header 'X-TenantID: cfd95b7e-3bc7-4006-a4a8-a73a79c71255' -d
'{
  "edgeType": "connectedTo",
  "edgeLabel": "dataFlow",
  "description": "Default relationship between two devices that exchange data"
}' 'https://localhost:8080/1.0/topology/types/edge'
```

Entity types

The Topology Service allows you to group resources of the same type that uses the **`entityTypes`** property, and identify them in the UI by their icon.

Usually a resource has only a single entity type, but, as the property is a *Set*, it is possible for a resource to have more than one entity type. Pre-defined associations of icons to specific entityTypes are listed in the following sections. You can create additional entity types as required.

- An entityType is defined by defining a resource with that type in the entityTypes list field.
- You can associate icons with named types in the Topology Viewer.
- You can add custom icons.

Predefined entity types

An entity type is used to map the resource vertex to an icon.

Notes:

- An administrator user can create more custom resource or entity types as required. Any observer may add new types.
- The topology service API only returns the resource or entity types in use.

To assign an entityType

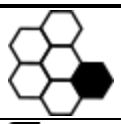
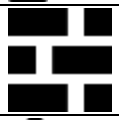
1. Select a resource with the provided entityType in the Topology Viewer.
2. Select the topology tools (the **cog** icon), then navigate to the resource types.
3. Find the type to be edited, and select the icon to assign it. You can find an icon either by name from the dropdown list, or by using the **View all icons** option (the **eye** icon).

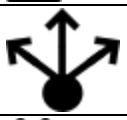
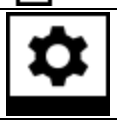
Entity type icons

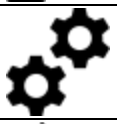
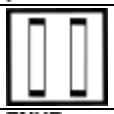
The following table lists these entity types with links to their icons, if defined:

Table. Entity type icons

Entity type	Icon
aircond	
alarm	
application	
backplane	
bridge	
building	
bypass	
card	

Entity type	Icon
cell	
channel	
chassis	
circuit	
cluster	
command	
component	
computer	
container	
cpu	
credential	
daemonset	
database	
datacenter	
deployment	
device	
directory	
disk	
domain	
emailaddress	
environment	
event	
fan	
file	
firewall	
flavor	
forum	
fqdn	
gauge	
group	

Entity type	Icon
host	
hsrp	
hub	
image	
input	
ipaddress	
ipam	
loadbalancer	
location	
memory	
message	
namespace	
network	
networkaddress	
networkgear	
networkinterface	
operatingsystem	
organization	
outlet	
output	
patch	
path	
pdu	
peripherals	
person	
pod	
printer	
process	
product	

Entity type	Icon
project	
psu	
resource	
router	
routetable	
routetarget	
rsm	
rule	
sector	
securitygroup	
sensor	
server	
service	
serviceaccesspoint	
slot	
snmpsystem	
stack	
statefulset	
status	
storage	
subnet	
subscription	
switch	
swarm	
task	
tcpudpport	
template	
tenant	
tier	
tool	

Entity type	Icon
ups	
variable	
version	
vlan	
vm	
volume	
vpn	
vrf	
workload	

Status (and state)

Resources have specific statuses, and in turn each status has a state of open, clear, or closed.

Status

A single status can affect multiple resources, and a single resource can have multiple different statuses. For example, a single *link down* event can generate the status for both the interface resource and the host resource, or, a single host might have *CPU* or *disk usage* status in addition to any *link down* status.

Resource status can be viewed in the Topology Viewer. An 'open' event is one with a severity other than 'clear'.

Important: When modeling resources, you must consider Status assignment from events.

Tip: The Topology Service stores the event **Severity**, and nodes in the UI are colored based on severity, which always has one of the following definitions:

- clear
- indeterminate
- information
- warning
- minor
- major
- critical

Look at the severity icons in the topology viewer reference topic Severity icons table.

Status assignment

The status of a single resource can be supplied, alongside other resource properties, when creating or updating a resource. Alternatively, an event can generate the status of one or more resources.

Remember: A status **always** has one of the following three states:

- **`Open`**

  Always has a severity other than 'clear' to indicate an active issue that can require your attention.

- **`Clear`**

  Working as expected. Also has a severity of 'clear'.

- **`Closed`**

  No longer active or relevant, a deleted event. Also has a severity of 'clear'.

Status assignment from events

The status service receives events and tries to find matching resources in the topology service. A resource with no match tokens defined will not have events matched to it, but if found, the status of those resources is set from the event data. The default fields of the alert used to look up resource matchTokens are *resource.sourceId* and *resource.name*. The assigned status depends on the following event data:

- **`matchTokens`**

  This property must be used to list any data that can identify (match) the resource.

  Each field may be globally unique, or may be unique within the scope of a composition. In other words, a resource modeled as a composition relationship, such as **`partOf`**, can be distinguished from other children within the composition using these fields.

  For example, either a globally unique and fully qualified domain name or IP address, or a locally unique interface name (that is, local within the scope of the host), can be used to identify the resource.

- **`partOf`** composition relationship

  The status service uses composition relationships to match fields that are unique only within the scope of a parent.

  For example, an IP address can be used to find a main node, and an interface name can be used to identify an interface within that main node.

- **Note:** If two different resources have the same **`matchTokens`** value, an event cannot be assigned to them unless they have been merged to become members of the same composite, **or** they have a parent-child **`partOf`** relationship.

Access scope

The scoping functions allows for preferential matching of IBM Cloud Pak for AIOps alerts. If similar resources and alerts in your environment exist in multiple different scopes, you can tag them with scope-specific information to allow a unique match to be made.

Such a scenario can occur (for example) under the following conditions:

- The same applications are running in test, staging and production environments.
- The same deployments are running in multiple namespaces in a single Kubernetes cluster.
- The same components are running in cloned Openstack projects.
- The same private IPs exist in multiple routing domains.

If an alert matches duplicate resources, a (case-insensitive) match of the alert scope against the resource scope is attempted.

Resource scope: Resources have an *accessScopeTokens* list property containing the following information:

- The provider of the resource, as defined by the observer job
- Optional additional scope supplied in the observer job's *Access Scope* CSV string parameter
- Scope explicitly added by some observers, such as the Kubernetes observer's *data_center* and *namespace* job parameters

Alert scope: The alert fields to use for scope are configurable.

- You can use the **AIOPS_SCOPING_FIELDS** variable defaults (*accessScope*, scopeId, *location*, *cluster*) to deuplicate resources when there are multiple matches on matchTokens.
- The values of these fields are compared with the resource property *accessScopeTokens* , with preferential matching based on case-insensitive comparison.

Troubleshooting alert lookup failures:

1. Check for a unique resource match at `GET /resources?_filter=matchTokens:{alertFieldToLookup}`
2. If the alert is older than the resource, it might not have existed when the alert processing looked it up.

Timestamps

Both vertex and edge graph elements can have multiple timestamps.

The following formats apply across all Topology APIs when specifying timestamps:

- Millisecond Epoch

  Example:

  ```
  date "+%s%3N"
  1655991641000
  ```

- RFC-3339 / ISO-8601

  Format:

  ```
  YYYY-MM-DDThh:mm:ss.sssZ
  ```

  Example:

  ```
  date --universal "+%FT%T.%3NZ"
  2022-06-23T13:40:41.000Z
  ```

- **`beginTime`** - The beginTime timestamp records the beginning of a period of time for which the resource was valid, with endTime marking the end of that period.

  Tip: A resource can have multiple begin times in its historic record, and gaps can happen in that record if the resource was offline for periods.

  - All resources and historic resources that are representations of the same thing have a distinct beginTime
  - Resource beginTime together with endTime is used in historic graph traversals, that is, when the _at parameter is supplied. The period during which a resource is valid is defined as:

```
atTime >= beginTime && atTime < endTime
```

    - A vertex or edge, which has the beginTime equal to the endTime, can be used to store audit information, such as the provider which deleted a resource. However, because it takes up zero time it does not form part of the history and is ignored by the preceding equation.

- **`prevBeginTime`** - If history exists for a resource, then this property is set to the beginTime of the most recent historical resource.
- **`changeTime`** - The changeTime timestamp records when the properties of the element last changed. The value can be less than the observedTime, which is updated on a POST or PUT even if no property values have changed.
- **`createTime`** - The createTime timestamp records when the element was first created in the topology service.
    - Historical resources do not store createTime, as it is shared with the anchor.
    - createTime is needed when looking for something older than 30 days – that is, when there is no beginTime this old because the historical resources have timed out.
- **`endTime`** - The endTime timestamp records when the element was deleted.
    - All resources and historic resources that are representations of the same thing have a distinct endTime.
    - Resource endTime is used in historic graph traversals, that is, when the _at parameter is supplied.
    - For current resources, endTime is LONG _MAX. This is sometimes hidden through the REST API.
- **`observedTime`** - The observedTime timestamp records when the element was last observed, that is, when data was last input to the topology service for the element.

Topology examples

The following examples describe best-practice guidelines to provide you with practical information, such as implementation examples and code samples, that you can use to get started.

Restriction: The examples provided here must be amended and adjusted to suit your own implementation.

Virtual machine example

One of the most important goals of IBM Cloud Pak® for AIOps is to support the assurance and provisioning of modern IT, network, and storage environments. These environments all make extensive use of increasingly nested virtualization technologies that need to be modeled. The following example introduces such an IT Virtualization scenario, and describes an OpenStack response that provides a solution.

Physical device example

The following example of an IBM Tivoli Network Manager environment accessed through IBM Cloud Pak for AIOps provides insights into any environment that uses physical network devices.

Virtual machine example

One of the most important goals of IBM Cloud Pak® for AIOps is to support the assurance and provisioning of modern IT, network, and storage environments. These environments all make extensive use of increasingly nested virtualization technologies that need to be modeled. The following recipe introduces such an IT Virtualization scenario, and describes an OpenStack response that provides a solution.

IT Virtualization

The IBM Cloud Pak for AIOps model of a nested virtualization scenario can extend the traditional types of models provided by other solutions.

This model can represent a multi-domain view of the world that links IT, network, storage, applications, and services. In addition, it can incorporate concepts such as OpenStack's Heat Orchestration and Slack collaboration relative to traditional IT resources.

Some of the benefits of this approach are:

- To provide extra context

  Increasingly, teams are more multi-disciplined and no longer operate in informational or functional silos. For example, network teams may include IT Virtualization specialists.

  Such teams need access to more context when needed, to be able to answer some of their business-critical questions, such as:

  - What storage volume is a VM attached to?
  - Which orchestration step realized a network port?
  - Who collaborated with whom for a particular incident?
  - Which applications and services are supported by a network subnet?
  - Which VM instances were shut down as part of a scale-in activity 1 hour ago?
  - What is the impact of removing a Hypervisor from the environment?
  - Which fixed IP addresses have a floating IP address that has been bound to in the last week?
- To provide a data-rich base

  Added value services can be bolted onto a base system, provided the information exists, and the system has an architecture that allows for rapid extension.

  For example, when building analytics on the topology data, the availability of information such as seasonality can provide extra insights.

  The following diagram depicts the nested layers of virtualization, including networking, between these layers and technologies such as Docker and LXC or LXD.

  Note: The services that are exposed can be applications or appear to be traditionally physical services such as network routers, switches, firewalls, and load balancers (a key goal of NFV).

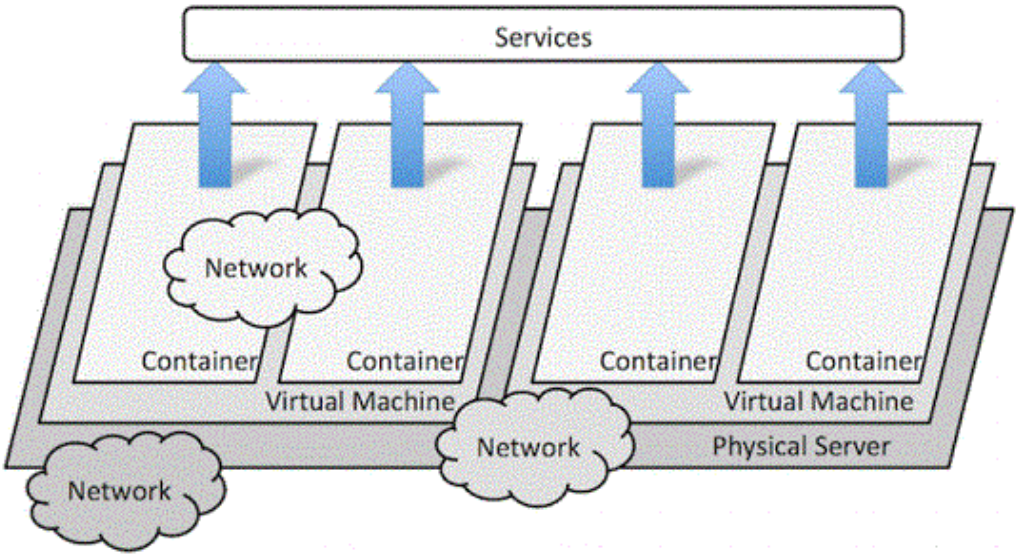


Figure. Physical services

OpenStack

OpenStack is a free and open source platform for cloud computing, typically deployed as an IaaS (Infrastructure-as-a-Service) capability. The software platform consists of interrelated components that control diverse, multi-vendor hardware pools of processing, storage, and network resources throughout and between data centers.

OpenStack provides a number of projects, and related services and APIs, that are summarized here, as they speak directly to the need to have a multi-domain view of the environment. For more information, see the OpenStack project navigator

OpenStack **core services** include the following services:

- Nova

  Compute manages the lifecycle of compute instances in an OpenStack environment.

- Neutron

  Networking enables network connectivity as a service for other OpenStack services.

- Swift

  Object Storage stores and retrieves arbitrarily unstructured data through a REST API.

- Cinder

  Block Storage provides persistent storage to running instances.

- Keystone

  Identity provides authentication and authorization services to OpenStack services and a service catalog.

- Glance

  Image Service stores and retrieves virtual machine disk images for use by Nova.

OpenStack **optional services** include the following services:

- Horizon

  dashboarding
- Ceilometer

  telemetry
- Heat

  orchestration
- Sahara

  Platform Symphony MapReduce
- Designate

  DNS
- Barbican

  Key Management

IBM Cloud Pak for AIOps provides an Observer that makes extensive use of OpenStack's core and Heat APIs to build an end-to-end (or multi-domain) model.

IT Virtualization OpenStack scenario

The following example of an OpenStack environment accessed through IBM Cloud Pak for AIOps provides insights into any environment consisting of, for example, a combination of physical services and storage combined with virtualization from VMware (or similar).

IT Virtualization OpenStack scenario

The following example of an OpenStack environment provides insights into any environment consisting of, for example, a combination of physical services and storage combined with virtualization from VMware, or similar virtualization providing infrastructure.

IT virtualization

The following figure depicts a Hypervisor running a VM (LB Server) that is assigned to a tenant, has classification data and a number of network interfaces. Each network interface has a MAC address, one or more IP addresses (v4 or v6) and each IP address is considered to be part of an IP subnet. It is also related to orchestration data from Heat that helps identify how the instance was created.

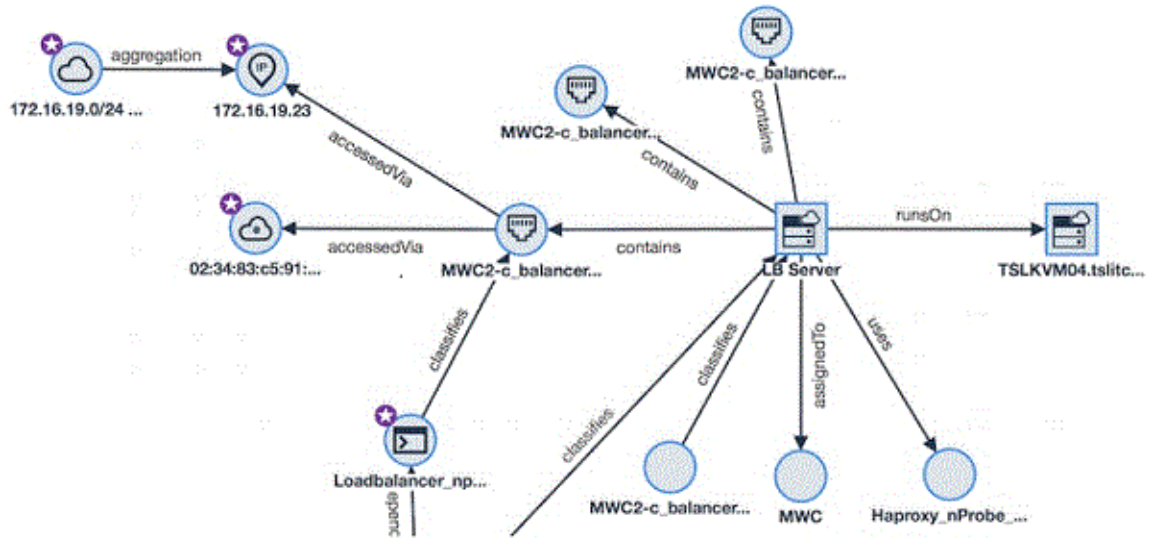


Figure. Example multi-domain topology

Property use guidance

Although the topology service is flexible, you should use the following guidelines when setting property values to ensure that elements are appropriately represented:

- Set **`entityType`** to one of the pre-defined entity types with an associated icon where possible.
- Set **`name`** and **`description`** to values that are easy to read or remember wherever possible.
- Use the default generic properties to represent generally applicable characteristics of an element and to provide a basic degree of labeling and filtering. For a list of generic properties, see the following topic: Properties

Model patterns - Part one

Stepping through each of the sections of the example of a multi-domain topology helps to identify reusable patterns.

In the following figure, the Hypervisor 'TSLKVM04' is running a VM 'LB Server'.

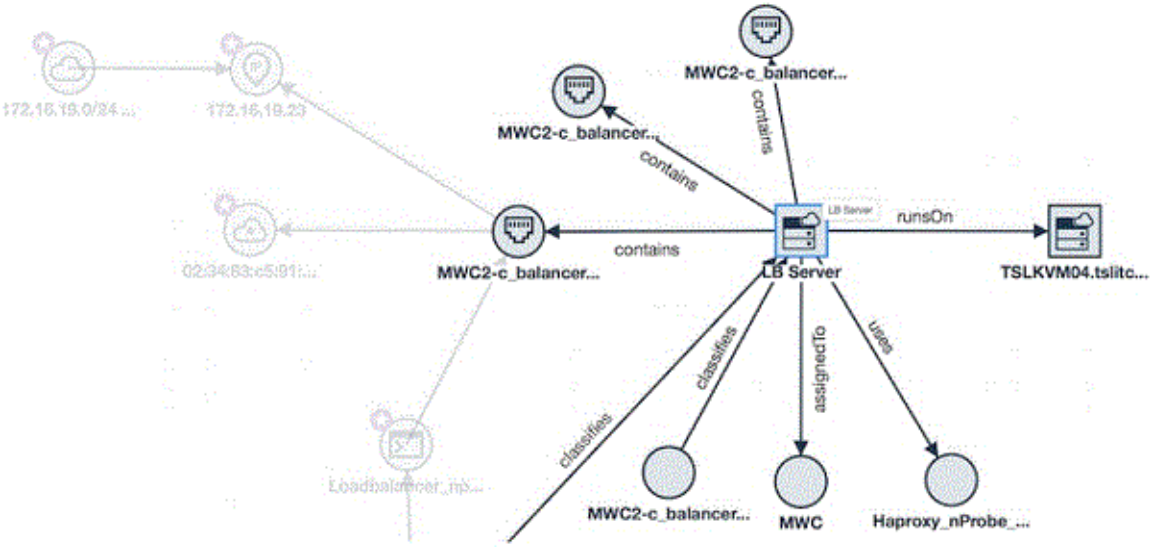


Figure. Figure of Hypervisor

The LB Server in this image is:

- Assigned to the 'MWC' tenant
- Contains three network interfaces
- Uses the Haproxy _nProbe image (OpenStack image)
- Is classified as an MWC2-c _balancer (OpenStack flavor)

This means that the following pattern can be identified:

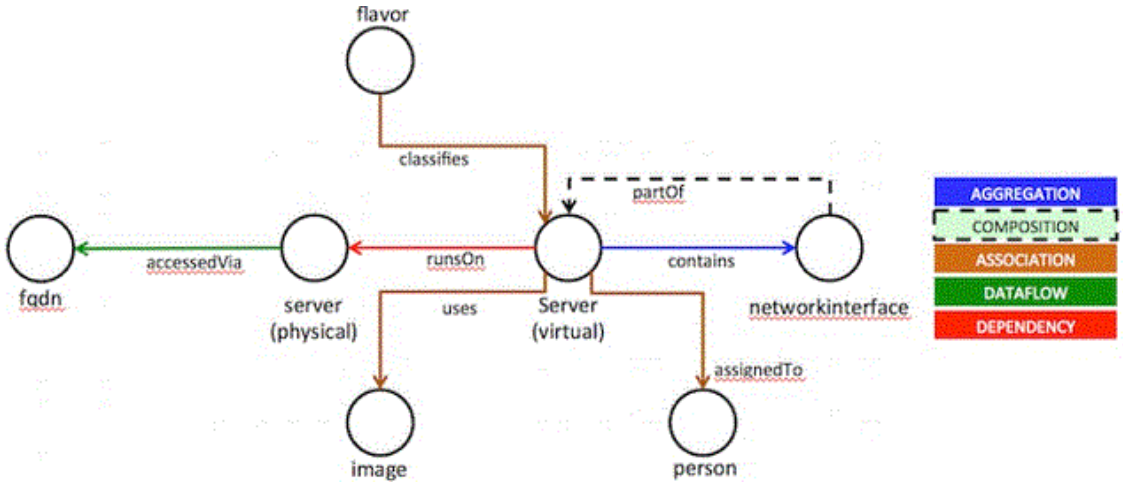


Figure. Pattern

Usage tips:

- The associations shown for flavor, image and person are optional.
- Network interfaces and other components of the device must be associated with it through a partOf relationship. The exception is if an IP or MAC address is known independently of any device, for example flow data would expose those but the device would be unknown.
- The FQDN (hostname, short or full DNS name) is associated directly with the server in this case as nothing else is known about the Hypervisor.

  Note: This is **not** shown in the topology fragment.
- The **`partOf`** composition is not shown by the topology GUI, but must be created where the relationship between a component and a device is known. This is in addition to relationships such as **`contains`**, which the IBM Cloud Pak for AIOps console will show.

Hypervisor example JSON

```
{
    "_executionTime": 3,
    "createTime": 1501741671066,
    "name": "TSLKVM04",
    "uniqueId": "CYooventry_DC1:MWC/ComputeHost/TSLKVM04",
    "observedTime": 1501776262368,
    "_startedAt": "2017-08-03T06:27:51.066Z",
    "entityTypes": [
        "server"
    ],
    "beginTime": 1501741671066,
    "_id": "KNN6TCGhKyM4MCI6jooGWg",
    "_observedAt": "2017-08-03T16:04:22.368Z",
    "_modifiedAt": "2017-08-03T06:27:51.066Z",
    "_createdAt": "2017-08-03T06:27:51.066Z",
```

```
    "changeTime": 1501741671066,
    "matchTokens": [
        "Coventry_DC1:MWC/ComputeHost/TSLKVM04",
        "TSLKVM04"
    ]
}
```

Note: Many of the properties starting with _ are internal (such as timestamps). Also:

- The `uniqueId` is a composite of a number of fields: data center, tenant, className and name of the Hypervisor because the ID is a highly ambiguous integer.
- The `name` is the name of the instance from OpenStack.
- The `entityType` is set to 'server' to ensure correct classification and icon use.

Virtual machine example JSON

```
{
    "instanceName": "LB server",
    "tenantId": "2f79c691570c4a598be386325ea01da8",
    "launchedAt": 1501741751194,
    "_executionTime": 4,
    "userId": "48c7cd25ad0842be8e9b84390de0e587",
    "imageName": "None Available",
    "availabilityZone": "nova",
    "createTime": 1501741733817,
    "flavorName": "m1.nano",
    "name": "LB server",
    "uniqueId": "1e35c68a-86b0-445f-9741-e581121a0577",
    "serverStatus": "active",
    "observedTime": 1501741752717,
    "_startedAt": "2017-08-03T06:29:12.717Z",
    "entityTypes": [
        "server",
        "vm"
    ],
    "beginTime": 1501741752717,
    "flavorId": "42",
    "vmState": "active",
    "_id": "3nDmTkKfrvNhZinSDCYHDw",
    "_observedAt": "2017-08-03T06:29:12.717Z",
    "createdAt": 1501741733000,
    "_modifiedAt": "2017-08-03T06:29:12.717Z",
    "_createdAt": "2017-08-03T06:28:53.817Z",
    "changeTime": 1501741752717,
    "matchTokens": [
        "1e35c68a-86b0-445f-9741-e581121a0577",
        "LB server"
    ]
}
```

Note: Many of the properties starting with _ are internal (such as timestamps). Also:

- The `uniqueId` in this case is the UUID of the instance from OpenStack.
- The `name` is the name of the instance from OpenStack.
- The `entityType` is set to 'server' and 'vm' to ensure correct classification and icon use.
- The `isVirtual` Boolean is set to true.

Network interface example JSON

```
{
    "_executionTime": 3,
    "isAdminStateUp": true,
    "createTime": 1501741717674,
    "name": "aab47c85-3110-401a-8bad-960b7c4bcd7b",
    "uniqueId": "aab47c85-3110-401a-8bad-960b7c4bcd7b",
    "observedTime": 1501741718855,
    "_startedAt": "2017-08-03T06:28:38.855Z",
    "entityTypes": [
        "networkinterface"
    ],
    "beginTime": 1501741718855,
    "isPortSecurityEnabled": true,
    "_id": "kkKpDH6yLn7cmNVXOddImg",
    "_observedAt": "2017-08-03T06:28:38.855Z",
    "_modifiedAt": "2017-08-03T06:28:38.855Z",
    "_createdAt": "2017-08-03T06:28:37.674Z",
    "changeTime": 1501741718855,
    "matchTokens": [
        "aab47c85-3110-401a-8bad-960b7c4bcd7b"
    ]
}
```


Note: As with the LB Server data, many of the properties are internal. Also:

- The `uniqueId` in this case is the UUID of the instance from OpenStack.
- The `name` is the name of the instance from OpenStack.
- The `entityType` is set to `networkinterface` to ensure correct classification and icon use.
- The type of the interface (such as `ifType` from IBM Tivoli Network Manager) is unknown.

Model patterns - Part two

Note: Each section of the example of the multi-domain topology identifies the reusable patterns.

In the following figure, the LB Server VM contains a network interface.

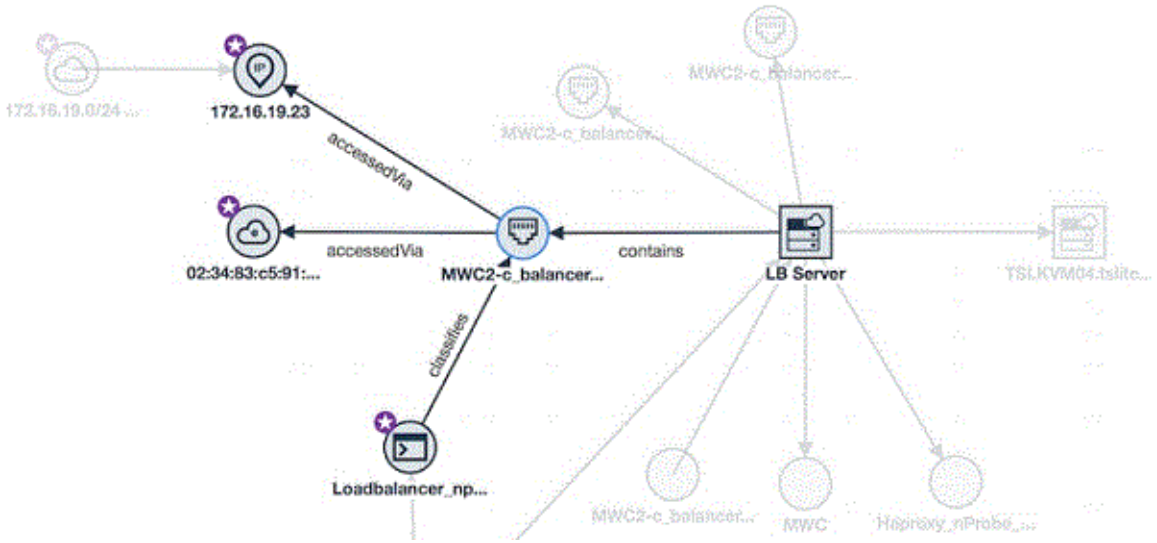


Figure. LB Server VM

The network interface in this image is:

- Accessed through an IP address and a MAC address

- Classified by a Heat orchestration resource

This means that the following pattern can be identified:

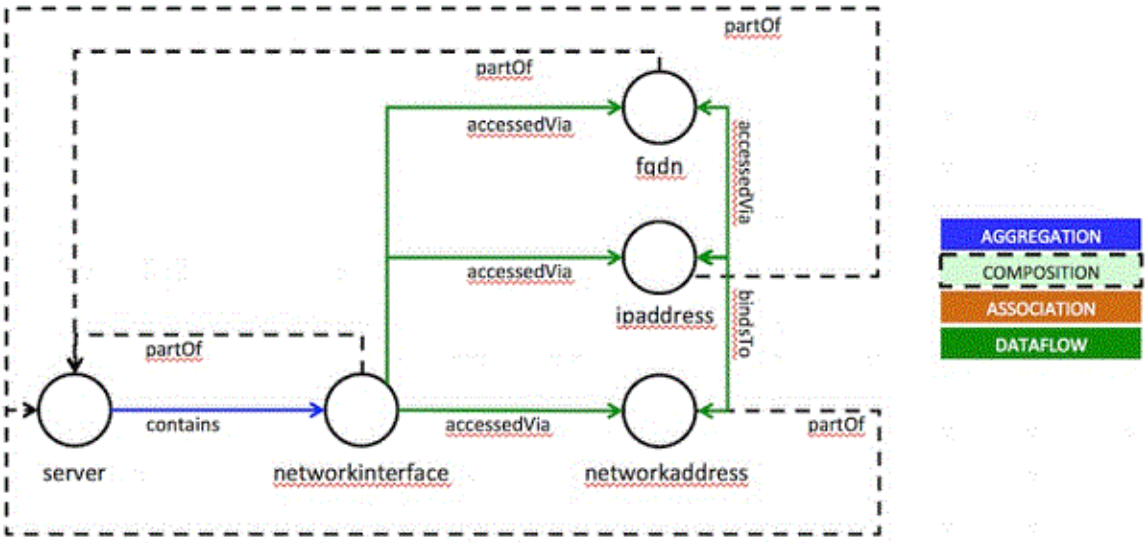


Figure. Pattern

Usage Tips:

- The Heat orchestration element and relationship is optional. Such things can be added if known to provide extra context.
- The network interface must be contained by and **`partOf`** the device. **`Contains`** is more fine-grained and may reference intermediate cards (for example) within the containment hierarchy of the device, such as

  `json device--contains-->card--->contains--->port`

- If an IP address, FQDN or MAC address is known to be related to a specific device, then they must be associated with the device through a **`partOf`** relationship *in addition* to the **`accessedVia`** relationship.
- If an IP address, FQDN, or MAC address is known independently of a device, then no **`partOf`** relationship from them is necessary.
- If an IP address is known to resolve to an FQDN, then relationships between them should be created to depict that one resolves to another and one accesses another (**`accessedVia`** shown in the example).

IP address example JSON

```
{
    "_executionTime": 3,
    "createTime": 1501741717656,
    "name": "172.24.4.5",
    "uniqueId": "172.24.4.5",
    "ipNumber": 2887255045,
    "addressSpace": "Coventry_DC1:MWC",
    "observedTime": 1501741731565,
    "_startedAt": "2017-08-03T06:28:51.565Z",
    "entityTypes": [
        "ipaddress"
    ],
    "beginTime": 1501741731565,
    "version": "IPv4",
```

```
    "_id": "jPLc72DU-UvPeTQE_7YdPQ",
    "_observedAt": "2017-08-03T06:28:51.565Z",
    "_modifiedAt": "2017-08-03T06:28:51.565Z",
    "_createdAt": "2017-08-03T06:28:37.656Z",
    "changeTime": 1501741731565,
    "matchTokens": [
        "172.24.4.5",
        "IPv4:2887255045"
    ]
}
```

Note:

- The `uniqueId` is the IP address itself. This is an RFC1918 IP address and so it **must** be qualified with an address space to disambiguate it from other instances of the same IP address. Similar precautions should be used with MAC addresses, which can be ambiguous.
- The `protocol` reflects whether the IP address is an IPv4 or IPv6 address.
- The `ipNumber` is a numeric representation of the IPv4 or IPv6 address. A Java BigInteger has the precision to represent IPv6 addresses.

Model patterns - Part three

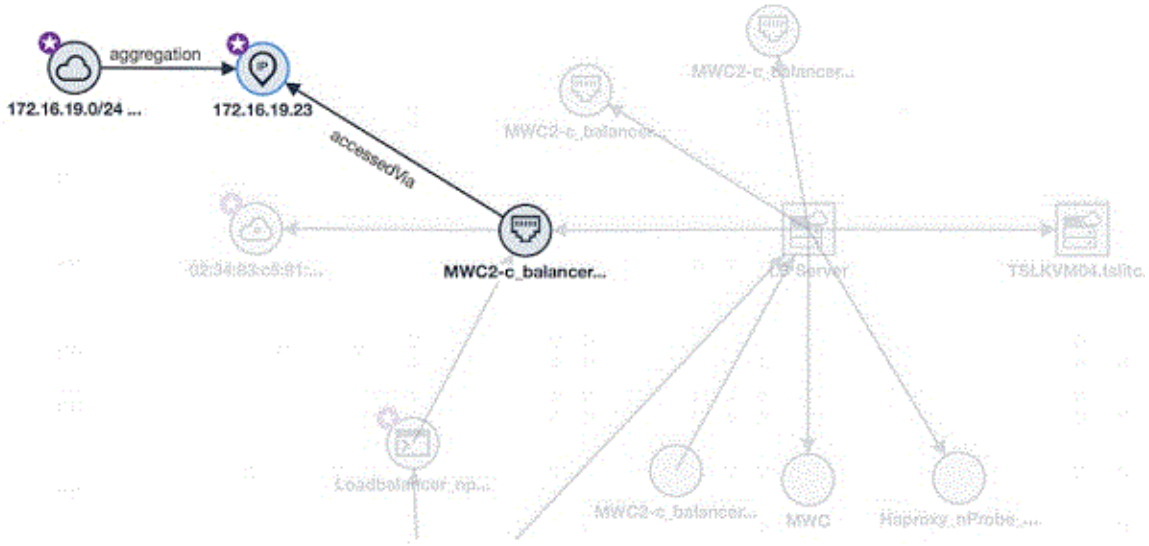


Figure. Multi-domain topology

This means that the following pattern can be identified:

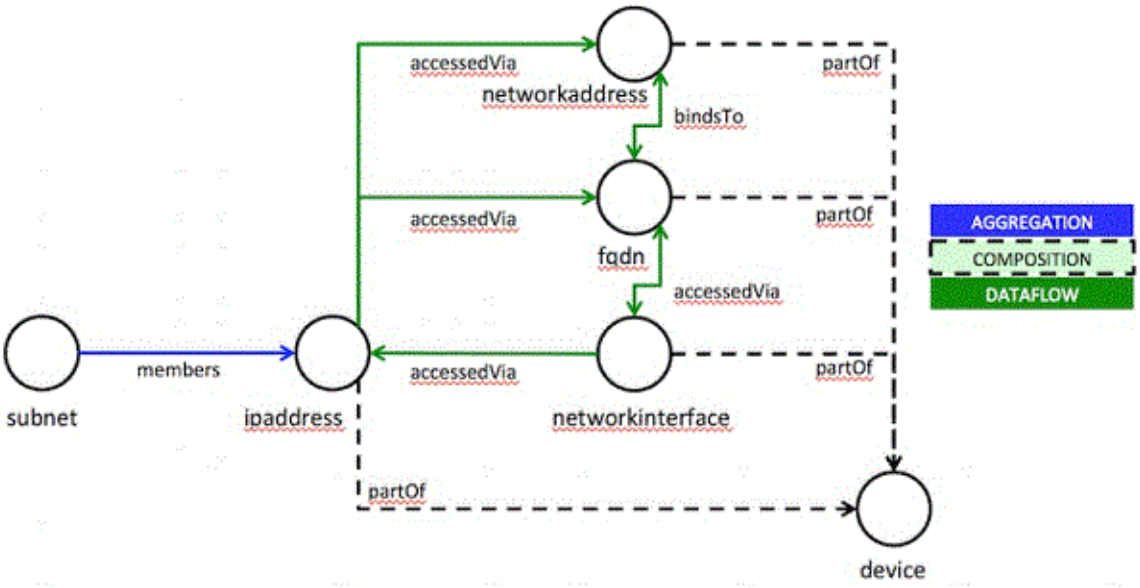


Figure. Pattern

Usage Tips:

- The IP subnet should aggregate any IP addresses known to be in it.
- Elements accessed through an IP should be related to it, such as a network interface, service, or process.
- If an FQDN is known to resolve to an IP address (and vice-versa), then they should be related.
- If a MAC address is known to bind to an IP address and vice-versa, they should be related.
- If an IP address, MAC address, or FQDN is known to relate to a device, they should be considered **`partOf`** it; otherwise, they are independent.

IP subnet example JSON


```
{
  "uniqueId": "c009ff59-13b9-48dc-8863-cd0c75070d99",
  "name": "172.24.4.0/24 (public-subnet)",
  "entityTypes": [
    "subnet"
  ],
  "matchTokens": [
    "172.24.4.0/24 (public-subnet)",
    "172.24.4.0/24",
    "c009ff59-13b9-48dc-8863-cd0c75070d99"
  ],
  "_id": "u0sjaHcumtg5A4DRl1fyAQ",
  "_references": [
    {
      "_id": "9duzx1-3o52g-ys5-47si0",
      "_edgeType": "contains",
      "_label": "aggregation",
      "_fromId": "u0sjaHcumtg5A4DRl1fyAQ",
      "_toId": "lSJQs8JmYzDQ-_wUOfvJbg",
      "_fromUniqueId": "c009ff59-13b9-48dc-8863-cd0c75070d99",
      "_toUniqueId": "172.24.4.12",
      "createTime": 1501838680418,
      "_observedAt": "2017-08-04T09:24:40.418Z",
      "_createdAt": "2017-08-04T09:24:40.418Z",
```

```
      "beginTime": 1501838680418,
      "_startedAt": "2017-08-04T09:24:40.418Z",
      "observedTime": 1501838680418
    },
  ],
  "_executionTime": 11,
  "_modifiedAt": "2017-08-04T09:24:40.356Z",
  "isDhcpEnabled": false,
  "dnsNames": "None Available",
  "_observedAt": "2017-08-04T09:24:40.356Z",
  "gatewayIp": "172.24.4.1",
  "_startedAt": "2017-08-04T09:24:40.356Z",
  "observedTime": 1501838680356,
  "changeTime": 1501838680356,
  "ipVersion": "V4",
  "createTime": 1501838680356,
  "_createdAt": "2017-08-04T09:24:40.356Z",
  "cidr": "172.24.4.0/24",
  "networkId": "3e2b5d07-653a-4fc8-8224-45801d9d113f",
  "beginTime": 1501838680356,
  "allocationPools": "172.24.4.2-to-172.24.4.254"
}
```


Notes:

- The **uniqueId** in this case is the UUID of the subnet from OpenStack.
- The **name** in this case is set to CIDR notation plus the ID of the subnet.
- The **entityType** is set to subnet to ensure appropriate classification and icon usage.
- Some example relationships are shown: For example, an IP address that is part of the subnet is visible, and the subnet's use of an allocation pool is also depicted.

Physical device recipe

The following example of an IBM Tivoli Network Manager environment accessed through IBM Cloud Pak® for AIOps provides insights into any environment that uses physical network devices.

Network physical devices

The following figure depicts two physical devices, 172.20.1.104 and 172.20.1.1, which are connected to each other.

Server 172.20.1.1 has 3 Gigabit Ethernet ports on the same card, one of which has a subinterface with an **ifName** of 'ge-1/1/3.0'. That subinterface shares the MAC address as its physical parent and has two IPv6 addresses and one IPv4 address associated with it.

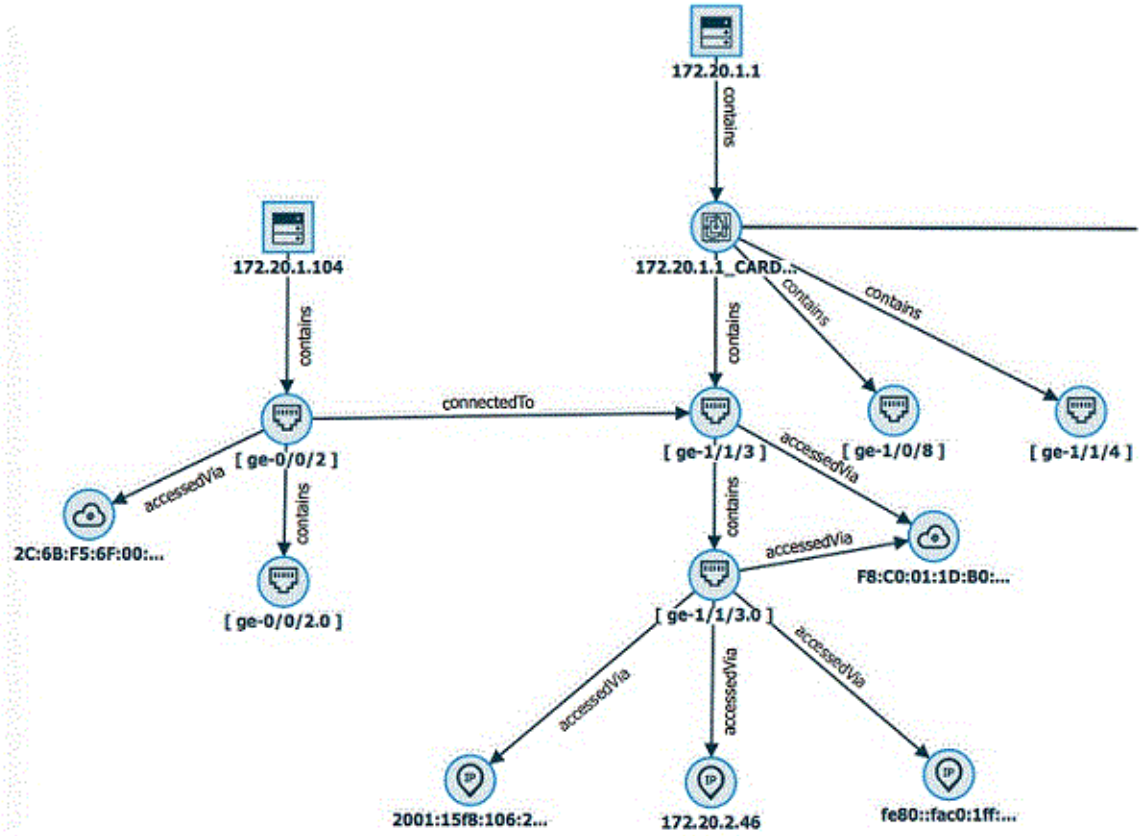


Figure. Figure that shows devices

Outline pattern

The topology service has an edge type of **`partOf`**, which is a member of the composition edge label family. For more information, see Edge labels.

The following image illustrates how this relationship is used to tie the card, interfaces, and network addresses to the hosting device.

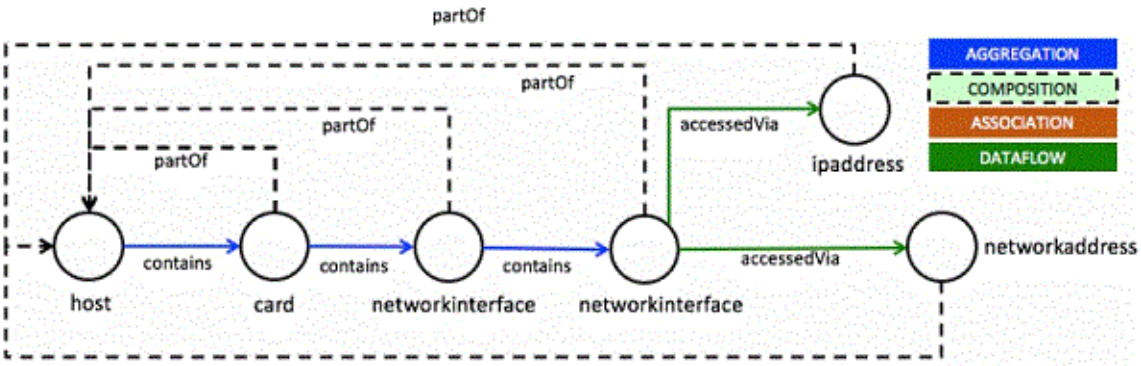
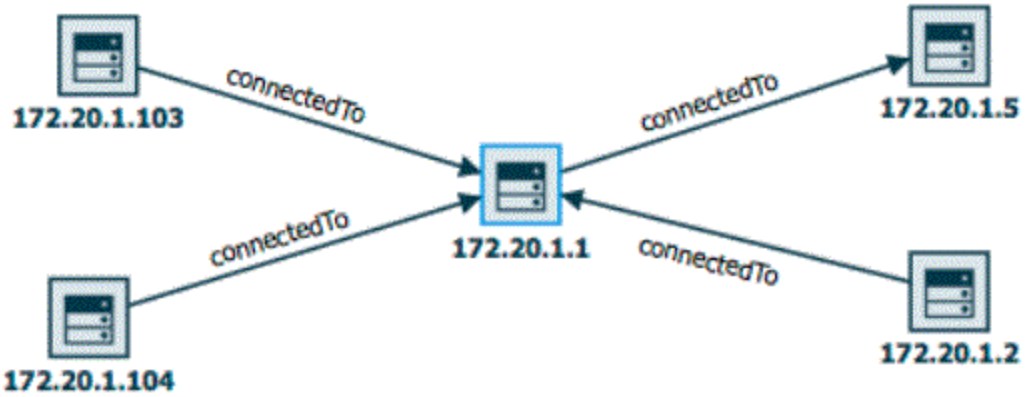


Figure. Relationships to hosting device

The **`partOf`** relationship is not shown as an explicit edge in the topology UI. However, it is used to determine which resources should be hidden in the host-to-host view, which in the context of this scenario would show just the hosts 172.20.1.104 and 172.20.1.1. It would also show the **`connectedTo`** edge between them, as depicted in the following part of the image.



{: caption="Figure. Figure that shows **partOf** relationship" caption-side="bottom"}

Host example JSON

The following JSON extract is an example of the properties that you might choose to include when creating a host vertex. The properties that start with an underscore character are aliases for some of the read-only generic properties. They also include a few read/write generic properties, such as **name** and **uniqueId**. However, most of the properties in this example are free from 'User' properties.

```
{
  "uniqueId": "NCOMS:172.20.1.1",
  "name": "172.20.1.1",
  "entityTypes": [
    "host"
  ],
  "_createdAt": "2017-03-03T16:02:40.845Z",
  "_observedAt": "2017-03-03T16:04:18.636Z",
  "_id": "y7EXOKrHud21CWySCyMsBg",
  "_href": "/1.0/topology/resources/y7EXOKrHud21CWySCyMsBg",
  "_nodeType": "resource",
  "_executionTime": 4,
  "_modifiedAt": "2017-03-03T16:02:40.845Z",
  "matchTokens": [
    "sbk-pe1-jrmx80.southbank.eu.test.lab"
  ],
  "sysObjectId": "1.3.6.1.4.1.2636.1.1.1.2.90",
  "entityChangeTime": "2017-03-03T14:13:17.000Z",
  "className": "JuniperMSeries",
  "services": "datalink(2) network(3)",
  "_startedAt": "2017-03-03T16:02:40.845Z",
  "manual": 0,
  "cdmAdminState": 0,
  "cdmType": 2,
  "sysDescription": "Juniper Networks, Inc. mx5-t internet router, kernel JUNOS
15.1F4.15, Build date: 2015-12-23 20:50:37 UTC Copyright (c) 1996-2015 Juniper
Networks, Inc.",
  "sysName": "sbk-pe1-jrmx80.southbank.eu.test.lab",
  "sysLocation": "The Mad Hatter Hotel 3-7 Stamford St London SE1 9NY
UK,-0.10499474,51.50711477",
  "interfaceCount": 73,
  "entityCreateTime": "2017-03-03T14:13:17.000Z",
  "isIpForwarding": "forwarding",
  "accessIPAddress": "172.20.1.1",
  "entityDiscoveryTime": "2017-03-03T14:11:09.000Z",
  "sysContact": "williamking@uk.ibm.com",
```

```
  "_tenantIds": [
    "Moa1dcmKHfx3dlyJnGm6JQ"
  ],
  "accessProtocol": "IPv4"
}
```

Note: Some of the generic properties in this example are:

- The `uniqueId` in this case is a string, which uniquely identifies this resource to IBM Tivoli Network Manager.
- The `name` is the name of the host and will be used in the UI.
- The `entityType` is set to 'host'.
- The `matchTokens` contains the FQDN name of the host.

Network interface example JSON

```
{
  "uniqueId": "NCOMS:172.20.1.1[ ge-1/1/3.0 ]",
  "name": "[ ge-1/1/3.0 ]",
  "entityTypes": [
    "networkinterface"
  ],
  "_createdAt": "2017-03-03T16:03:03.889Z",
  "_observedAt": "2017-03-03T16:03:03.939Z",
  "_id": "v8aFVG6JoigYxTrlFSDkLw",
  "_href": "/1.0/topology/resources/v8aFVG6JoigYxTrlFSDkLw",
  "_nodeType": "resource",
  "_executionTime": 5,
  "operationalStatus": "started",
  "ifIndex": 537,
  "ifAdminStatus": "up",
  "_modifiedAt": "2017-03-03T16:03:03.940Z",
  "ifType": 53,
  "matchTokens": [
    "ge-1/1/3.0",
    "ifEntry.537"
  ],
  "ifName": "ge-1/1/3.0",
  "ifTypeString": "propVirtual",
  "connectorPresent": "false",
  "_startedAt": "2017-03-03T16:03:03.939Z",
  "speed": 1000000000,
  "mtu": 1500,
  "accessIpAddress": "172.20.2.46",
  "operationalDuplex": "FullDuplex",
  "promiscuous": false,
  "physicalAddress": "F8:C0:01:1D:B0:13",
  "ifDescription": "ge-1/1/3.0",
  "ifOperStatus": "up",
  "_tenantIds": [
    "Moa1dcmKHfx3dlyJnGm6JQ"
  ],
  "accessProtocol": "IPv4"
}
```

Note: Some of the generic properties in this example are the following:

- The `uniqueId` in this case is a string, which uniquely identifies this resource to IBM Tivoli Network Manager.
- The `name` is the name of the host and will be used in the UI.
- The `entityType` is set to `networkinterface`.

- The `matchTokens` contains both the `ifName` and a string denoting the `ifIndex` in the `ifEntry` table.

Infrastructure Automation APIs

Review the following documentation to learn an overview of the REST API for the Infrastructure Management and Managed services components of Infrastructure Automation. These guides provide an overview of the API, reference material, and examples.

- Infrastructure Management APIs
- Managed services APIs