Telemetry Service Private Edition Guide

Configure Telemetry Service

8/4/2025

Contents

Learn how to configure Telemetry Service.

Related documentation:

-
-
-

RSS:

- For private edition

Configure a secret to access JFrog

If you haven't done so already, create a secret for accessing the JFrog registry:

```
kubectl create secret docker-registry  --docker-server= --docker-username= --docker-password=
--docker-email=
```

Now map the secret to the default service account:

```
kubectl secrets link default  --for=pull
```

Override Helm chart values

Parameter	Description	Default	Valid values
serviceMonitoringAnnotations.enabled	Activation of Prometheus monitoring annotations on service.	true	
podDisruptionBudget.enabled	Activation of pod disruption.	true	
enableServiceLinks	Enable service links in single namespace environment.	false	
tlm.replicaCount	Number of replicas.	2	
tlm.image.registry	docker registry.	pureengage-docker-staging.jfrog.io	
tlm.image.repository	docker registry.	Telemetry	
tlm.image.tag	WWE image version.		
tlm.image.pullPolicy	Image pull policy.	IfNotPresent	
tlm.image.imagePullSecrets	Image pull secrets.	[]	

Parameter	Description	Default	Valid values
tlm.service.type	k8s service type.	ClusterIP	
tlm.service.port_external	k8s service port external (for customer facing).	8107	
tlm.service.port_internal	k8s service port internal (for metric scrapping endpoint).	9107	
tlm.ingress	Ingress configuration block. See #Ingress.	{enabled:false}	
tlm.resources.limits.cpu	Maximum amount of CPU K8s allocates for container.	750m	
tlm.resources.limits.memory	Maximum amount of Memory K8s allocates for container.	1400Mi	
tlm.resources.requests.cpu	Guaranteed CPU allocation for container.	750m	
tlm.resources.requests.memory	Guaranteed Memory allocation for container.	1400Mi	
tlm.deployment.strategy	k8s deployment strategy.	{}	
tlm.priorityClassName	k8s priority classname.		
tlm.affinity	pod affinity.	{}	
tlm.nodeselector	k8s nodeselector map.	{ genesysengage.com/ nodepool: general }	
tlm.tolerations	pod toleration.	[]	
tlm.annotations	pod annotations.	[]	
tlm.autoscaling.enabled	activate auto scaling.	true	
tlm.autoscaling.targetCPUPercent	CPU percentage autoscaling trigger.	40	
tlm.autoscaling.minReplicas	Minimum number of replicas.	2	
tlm.autoscaling.maxReplicas	Maximum number of replicas.	10	
tlm.secrets.name_override	Name override of the secret to target.		
tlm.secrets.TELEMETRY_AUTH_CLIENT_SECRET	GAuth client Secret value.		
tlm.context.envs.*	Environment variables for Telemetry Service. Please refer to TLM service documentation.		

You can modify the configuration to suit your environment by two methods:

- Specify each parameter using the --set key=value[,key=value] argument to helm install. For example,

```
helm install telemetry-service.tgz --set tlm.replicaCount 4
```

- Specify the parameters to be modified in a **values.yaml** file.

```
helm install --name tlm -f values.yaml telemetry-service.tgz
```

Configure security

To learn more about how security is configured for private edition, be sure to read the Permissions and OpenShift security settings topics in the *Setting up Genesys Multicloud CX Private Edition* guide.

The security context settings define the privilege and access control settings for pods and containers.

By default, the user and group IDs are set in the **values.yaml** file as `500:500:500`, meaning the **genesys** user.

```
optional:
  securityContext:
    runAsUser: 500
    runAsGroup: 500
    fsGroup: 500
    runAsNonRoot: true
```

Environment variables

Parameter	Description	Default	Valid values
tlm.context.envs.TELEMETRY_AUTH_CLIENT_ID	GAuth client ID value.	telemetry_client	
tlm.context.envs.TELEMETRY_CLOUD_PROVIDER	Specify the mode how telemetry service should be executed. Possible values aws / azure .		
TELEMETRY_SERVICES_AUTH	URL of the GWS Auth public API. This is a mandatory field.		http://gws-core-auth:8095
TELEMETRY_AUTH_CLIENT_ID	The Client ID that is used to authenticate with GWS Auth service.	telemetry_client	
TELEMETRY_CORS_DOMAIN	Domains to be supported by CORS. This can a comma separated list.		

Parameter	Description	Default	Valid values
	Important Add a `\` before `.` for regex matching. eg: `\.genesyslab\.com` (another `\` should be added when using quotes).		
TELEMETRY_TRACES_PROVIDER	The trace provider to use can be `ElasticSearch` or `Console`.	ElasticSearch	
TELEMETRY_TRACES_CONCURRENT	The maximum of parallel bulk request to Elasticsearch at the same time.	3	
TELEMETRY_TRACES_THRESHOLD	The maximum buffer entries for Elasticsearch service.	400000	
TELEMETRY_CONFIG_SERVICE	The data source to fetch configuration information. Possible values : s3, azure, env, or an empty string.	none	
TELEMETRY_CONFIG_SERVICE_CORS	This overrides data source to fetch CORS configurations. Possible values : Same value as `TELEMETRY_CONFIG_SERVICE` or `environmentservice` for using the environment-service API (Uses the `TELEMETRY_SERVICES_ENVIRONMENT` variable).	none	
TELEMETRY_CLOUD_PROVIDER	Cloud provider for the service. Can be `aws`, `azure`, `gcp` or `premise`.	aws	
TELEMETRY_CONFIG_CONTRACTS	Stringified JSON array to provision contracts through `env` config provider.	[]	
TELEMETRY_CONFIG_TENANTS	A Stringified JSON to provision tenants through `env` config provider.	{}	
TELEMETRY_SERVICES_ENVIRONMENT	The URL of the GWS environment service API. Used only if environment service is used for configuration	value of TELEMETRY_SERVICES_AUTH	http://gauth-environment-active.gauth

Parameter	Description	Default	Valid values
	provisioning.		

Prepare an environment

Create a new project namespace for Telemetry:

```
kubectl create namespace tlm
```

See Creating namespaces for a list of approved namespaces.

Download the telemetry helm charts from the JFrog repository:

```
https://pureengage.jfrog.io/artifactory/helm-staging/tlm
```

Create a **values-telemetry.yaml** file and update the following parameters:

```
TELEMETRY_AUTH_CLIENT_SECRET:
TELEMETRY_AUTH_CLIENT_ID:
TELEMETRY_SERVICES_AUTH: ""
TELEMETRY_CLOUD_PROVIDER: "GKE"
TELEMETRY_CORS_DOMAIN: ""
grafanaDashboard:
  enabled: true
```

Copy the **values-telemetry.yaml** file and the **tlm** Helm package to the installation location.