Genesys Callback Private Edition Guide

Configure Genesys Engagement Service

4/26/2024

Contents

Learn how to configure Genesys Callback.

Related documentation:

-
-
-
-

RSS:

- For private edition

Override Helm chart values

For additional information about overriding Helm chart values, see Overriding Helm Chart values in the *Genesys Multicloud CX Private Edition Guide*.

Genesys Engagement Service (GES) has a Helm value layout similar to other Engage services, however the use of environment variables as the primary way to configure GES means that setting up the Helm values is particularly important. Below is a section-by-section description of the Helm chart parameters, what they influence, and any other information that you might find helpful to know when provisioning GES within Private Edition.

All values to configure GES are within the `.ges` subsection of the `.Values.yaml` file. For example, to set affinity, you reference `.Values.ges.affinity`. For those values listed in a `misc` section, omit that from the path when referencing an environment variable.

labels

Parameter	Description	Default	Valid values
globalLabels	Labels that are going to be applied to all resources within the GES helm deployment. This is required.	{}	YAML object
podLabels	Labels that will be applied to all GES pods. This is required.	{}	YAML object
serviceLabels	Labels that will be applied to just the GES service. This is required.	{}	YAML object

annotations

Parameter	Description	Default	Valid values
globalAnnotations	Annotations that are going to be applied to all resources within the GES helm deployment. This is required.	{}	YAML object
podAnnotations	Annotations that will be applied to all GES pods. This is required.	{}	YAML object

configMap

Set values in the config map. The values are divided into two sections:

- Environment variables that control the behavior of GES.
- Values supplied to GES that allow it to integrate with dependencies like Redis, Postgres, GWS, ORS, and so on.

env

log

Parameter	Description	Default	Valid values
level	Controls the level of logging output for GES.	DEBUG	(DEBUG, TRACE, ERROR, FATAL, INFO, WARN)
maxLen	The max length (in bytes) before GES log messages get auto-truncated.	2048	Number

server

Parameter	Description	Default	Valid values
internal_port	The port GES listens to for incoming requests from within Engage. Typically 3050. This is required.		Number
external_port	The port GES listens to for incoming messages to the external APIs (typically from the outside world). Typically		Number

Parameter	Description	Default	Valid values
	2050. This is required.		
internal_url	A default URL that would allow for ORS to query this instance of GES. The default value assumes GES is within the GES namespace. This is required.	ges.ges	String

misc

Parameter	Description	Default	Valid values
regionAffinity	Determines the ORS/URS nodes that will be given priority to handle requests for a given GES nodes. If all servers within the priority region are down, a secondary region will be used. This is required.		String
devopsAccessGroups	The list of roles that are to be granted the DevOps permission in GES. This is required.	Platform Read Only,Platform Provisioning,Platform Internal Administrator	String

integrations

redis

Parameter	Description	Default	Valid values
host	The host of GES' redis instance. This can also be provisioned using secrets. This is required.		String
port	The port the Redis instance listens on. This can also be provisioned using secrets. This is required.	6379	Number

Parameter	Description	Default	Valid values
secure	True if GES is to connect to Redis over TLS. This can also be provisioned using secrets. This is required.	false	Boolean

db

Parameter	Description	Default	Valid values
host	The host of GES' Postgres instance. This can also be provisioned using secrets. This is required.		String
name	The name of the DB provisioned for GES. This can also be provisioned using secrets. This is required.		string
secure	True if GES is to connect to Postgres over TLS. This can also be provisioned using secrets. This is required.		Boolean

gws

Parameter	Description	Default	Valid values
auth	The internal-facing hostname/port for GWS auth service. This is required.		String
env	The hostname/port for GWS env service. This is required.		String
conf	The hostname/port for GWS config service. This is required.		String
public_auth	The public-facing hostname/port for GWS auth service.		String

Parameter	Description	Default	Valid values
	This is required.		

vmcs

Parameter	Description	Default	Valid values
redis_host	The hostname of the Voice Microservices Redis instance. This can also be supplied through the use of secrets, but if supplied through secrets, it will override the value supplied in the map. This is required.		String
redis_port	The port of the Voice Microservices Redis instance. This can also be supplied through the use of secrets, but if supplied through secrets, it will override the value supplied in the map. This is required.		String
redis_cluster_mode	True if Voice Microservices is running in cluster mode. Otherwise, False. This is required.		String
redis_stream_name	The Redis stream that GES should connect to to write information about new callbacks. This is required.	"NewCallbackStream"	String
ors_redis_location	A compound value made from defining the Redis host/port. Leave as the default value. This can also be supplied through the use of secrets, but if supplied through secrets, it will override the value supplied in the config map. This is required.	"{{ .Values.ges.configMap.integrations.vmcs.redis_host }}:{{ .Values.ges.configMap.integrations.vmcs.redis_port | default 10000 }}"	

deployment

Parameter	Description	Default	Valid values
enableServiceLinks	Controls whether service information is added to environment variables or not. When the value is `true`, a set of environment variables for each active service is added to the pod.	false	Boolean

image

Parameter	Description	Default	Valid values
imagePullSecrets	A secret needed to authenticate with the docker registry. Might not be required; implementation is based on how you have set up your cloud. For example, in Genesys CX on Azure, the Pod ID is used to authenticate with the Azure Container Registry.	[ ]	List
imagePullPolicy	How often the pod will try to pull a fresh image on start up. This is required.	Always	String
registry	The docker registry that the GES image is to be pulled from. This is required.		String
image	The name of the docker image for GES. This is required.	genesys/ges	String

hpa

Parameter	Description	Default	Valid values
enabled	True when horizontal pod auto-scaling is enabled. Might not be required; implementation is based on how you have set up your cloud. For example, in Genesys CX on Azure, the Pod ID is used to authenticate with the Azure Container Registry.	True	Boolean

Parameter	Description	Default	Valid values
maximumReplicas	Maximum number of GES pods that will be running in a deployment.	5	String
minimumReplicas	Minimum number of pods that will run in a deployment.	1	String
targetMetrics	The metrics that the Kubernetes controller will consult when evaluating if additional GES pods are needed. For more information, see the Kubernetes documentation.		String
behavior	Defines the behavior when extra pods for GES are spun up and down, including how many pods should be added or removed at a time and how long the deployment is given to stabilize before spinning up or winding down extra pods. For more information, see the Kubernetes documentation.		

misc

Parameter	Description	Default	Valid values
env	A list of environment variables to be defined within the pod. These can be defined using simple key value pairs or using secrets as described in the Kubernetes documentation.	[ ]	List
envFrom	A list of config maps from which the GES pod will get the environment. Be sure to include all config maps and other entries from which you wish to pull environment variables. For more information, see the Kubernetes documentation.		List

Parameter	Description	Default	Valid values
min_ready_seconds	The amount of seconds that a pod must be in the ready state before the Kube controller begins routing traffic to it.	300	String
dnsConfig	Values related to the configuration of DNS resolution. For more information, see the Kubernetes documentation.		Object
replicas	The number of replica GES deployments that are part of the cluster by default. This can be scaled up or down. For more information, see the Kubernetes documentation.	1	Number
nodeSelector	Contains any and all values that might aid the Kubernetes controller to determine what kind of cluster GES is to be scheduled on. For more information, see the Kubernetes documentation.		Object

resources

Parameter	Description	Default	Valid values
limits	Defines the limits of the resource requests that a GES pod can make from the cluster as well as the maximum amount to allot. For more information, see the Kubernetes documentation.	cpu: 1250m memory: 2048Mi	Object
requests	Defines the size of CPU/RAM requests a GES pod can make from the cluster as well as the maximum amount to allot.	cpu: 75m memory: 512Mi	Object

Parameter	Description	Default	Valid values
	For more information, see the Kubernetes documentation.		

ingress

ingressint

Parameter	Description	Default	Valid values
enabled	True when you want a shared app gateway to govern incoming connections to GES. This is required.	false	Boolean
annotations	Any annotations that are to be applied to the ingress object that governs how GES communicates with a shared App Gateway.		
hosts	The host name to reach GES from a shared app gateway.		
paths	The set of paths that the app gateway will forward to GES. Omitting these paths could cause the external APIs to break.	- /ges/ - /engagement/v3/callbacks/create - /engagement/v3/callbacks/cancel - /engagement/v3/callbacks/retrieve - /engagement/v3/callbacks/availability/ - /engagement/v3/callbacks/queue-status/ - /engagement/v3/callbacks/open-for/ - /engagement/v3/estimated-wait-time - /engagement/v3/call-in/requests/create - /engagement/v3/statistics/operations/get-statistic-ex	List
tls	Details of the TLS certificates for incoming connections to GES.		

monitoring

prometheus

Parameter	Description	Default	Valid values
use_service_monitor	True when you use a Service Monitor as the mechanism for service/pod discovery by the Prometheus framework. For more information, see Simple Management of Prometheus Monitoring Pipeline with the Prometheus Operator.		boolean

grafana

Parameter	Description	Default	Valid values
enabled	True if Grafana dashboards will be deloyed from config maps defined in the Helm Chart.	False	Boolean

podDisruptionBudget

Parameter	Description	Default	Valid values
enabled	True when a PodDisruptionBudget is defined for GES.	True	Boolean
strategy	The strategy used to implement the pod disruption budget. For more information, see the Kubernetes documentation.		

secrets

The **secrets** section defines how different types of secrets are made available to the GES deployment through the use of volume mounts. Using this method is not strictly necessary, as it is possible to provide the same configuration information using Environment variables. At a high level, the kind of data that is made available through the use of secrets includes Redis/ORS Redis hosts and credentials, GWS client information as well as the DB credentials. For a more thorough explanation and examples of how GES can leverage the secrets capabilities, see Define secrets. For a detailed

look at the capabilities of Kubernetes in general, see the Kubernetes documentation on secrets and the Kubernetes documentation on volumes and volume mounts, which GES uses to actually deliver the secrets.

volumeMounts

Parameter	Description	Default	Valid values	Notes
enabled	True when volume mounts are used for the deployment. This is required.	false	Boolean	Required
list	A list of volume mounts to be provisioned for the GES deployment.			

volumes

Parameter	Description	Default	Valid values
enabled	True when volumes are used for the deployment. This is required.	false	Boolean
list	A list of volumes to be provided for the GES deployment.		

misc

Parameter	Description	Default	Valid values
config_folders	The list of folders that GES uses to read configuration information. These folders are the volumes mounted by the secrets infrastructure. For more information, see Configure security.	""	String

service

Parameter	Description	Default	Valid values
port_external	The port on which the service listens for external connections from outside the Genesys Multicloud CX solution.	80	Number

Parameter	Description	Default	Valid values
	This is required. For more information, see the Kubernetes documentation.		
port_internal	The port on which the service listens for connections from inside the Genesys Multicloud CX solution. This is required. For more information, see the Kubernetes documentation.	8080	Number
type	Defines whether or not (and how) a service is exposed to the outside world. This is required. For more information, see the Kubernetes documentation.	ClusterIP	See the Kubernetes documentation.

misc

Parameter	Description	Default	Valid values
serviceName	The service name you want to attach to a GES deployment. Only used in the Helm charts to help determine the name attached to the deployment.	ges	String
tolerations	Defines the tolerations to be applied to the GES pods. Pods with matching taints will be favored for scheduling. For more information, see the Kubernetes documentation.	{}	Object
affinity	Defines the pod affinity behavior used to determine which nodes the GES pods are scheduled on. For more information, see the Kubernetes documentation.	{}	

Configure Kubernetes

For information, see the following resources:

- the configMap section in the Helm values description
- the secrets section in the Helm values description
- Security
- Deploy the service

Configure security

In addition to supporting configuration through the supply of environment variables, GES supports configuration information being supplied through Kubernetes' secrets, which are subsequently mounted as volumes on the Kubernetes pod. This "secrets" mechanism is required when providing sensitive information to GES, such as usernames and passwords for connections to Redis, Postgres, GWS, and others. However, you can also use secrets to supply less-sensitive information to GES.

To successfully acquire the GES docker image, deploy a secret in the GES namespace that provides Kubernetes with necessary credentials to authenticate to the JFrog artifactory.

This section provides information about the security context settings and a general overview of how to enable secrets in the Helm charts through the use of volume mounts and includes a detailed look at how to integrate with other components using secrets.

Security context configuration

The security context settings define the privilege and access control settings for pods and containers. For more information, see the Kubernetes documentation.

Configuring a JFrog secret

For a full description of downloading Genesys Multicloud CX containers and accessing repositories, see Downloading your Genesys Multicloud CX containers in *Setting up Genesys Multicloud CX Private Edition*.

If you're creating a secret containing JFrog account details, the secret must be of type `kubernetes/dockerconfigjson`. You can find more information on this topic in the Kubernetes documentation. Regardless of platform, you create the secret using the following `kubectl` command:

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

Updating the Values.yaml file

In your **Values.yaml** file, update the value of `.Values.ges.deployment.image.imagePullSecrets`. For example:
`imagePullSecrets: - name:`

Providing secrets to GES

At present, while it is possible to supply confidential values to GES through the facility in Kubernetes where secrets can be read into Pod environment variables, this is not always the case. Genesys strongly recommends that you make confidential information such as database usernames and passwords, GWS client information, and Redis keys available to GES through volume mounts. Volume mounts are described in this section, but for more information, see the Kubernetes documentation.

Secrets through volumes

GES incorporates secrets through the use of Volumes that are mounted to the GES Kubernetes pod. There are a number of ways in which you can define the secrets in the Kubernetes deployment, including manually defining secrets that are deployed along with the GES template, using tools such as Terraform, and so on. Instructions about how to leverage the different tools is outside the scope of this documentation. What's important is that you turn the secret into a volume.

To enable volumes, set the value `.Values.ges.secrets.volumes.enabled` to **true**, and then – for each secret that you want to make available – fill out a list entry. For example:

```
volumes:
  enabled: true
  list:
  - name:
    secret:
      secretName:  # This is the name of the secret within the Kubernetes Cluster
```

You can then mount the volume to GES. Set `.Values.ges.secrets.volumeMounts.enabled` to **true** and fill out a list entry. For example:

```
volumeMounts:
  enabled: true
  list:
    - mountPath:  # Best practise is just / but this can be different if requirements dictate
      name:
      readOnly: true
```

Finally, you must specify the folders that contain the configuration information for GES. To do this, enter the list of s in the form of a comma-separated string as the value for `.Values.ges.secrets.configFolders`.

While the examples here demonstrate how to provide secrets as typical Kubernetes secrets, GES also supports mounting secrets using the Container Storage Interface. Provided the CSI is configured correctly in the Kubernetes environment, you should be able to set this up with only some minor changes in the Helmvalues files.

Using secrets to integrate with other software

GES primarily uses secrets to create a secure mechanism through which it can obtain the credentials necessary to authenticate with services like Redis, GWS, the Postgres database, and the Voice Microservices Redis. This section provides information about how to accomplish this integration; this information is based on deployments on Azure.

REDIS

The REDIS-CACHEKEY must be supplied to GES through the use of an opaque secret. This is the password used to connect to the GES Redis instance. For this secret, the key is REDIS-CACHEKEY and the value is the password. You can provision this on its own or along with other secrets such as the database access information.

DB

The DB-USER and DB-PASSWORD must be supplied to GES through the use of an opaque secret. This is the username and password for GES to connect to the Postgres database. You can provision this on its own or along with other secrets such as the REDIS-CACHEKEY.

GWS

The AUTHENTICATION-CLIENT-ID and AUTHENTICATION-CLIENT-PASSWORD needed to authenticate with GWS must be supplied as an opaque secret. GES supports both encrypted and unencrypted client grants. However, if using an unencrypted grant, it is important that you encode the AUTHENTICATION-CLIENT-PASSWORD using the aes-128-ecb algorithm and an encryption key supplied by the password defined in the Helm value `.Values.ges.configMap.env.decrypt_password`.

ORS

The password to the ORS Redis instance must be supplied to GES through the use of an opaque secret. Unlike other secrets, which can be set up as simple key-value pairs, ORS REDIS information must be supplied to GES in a key-value pair where the key is `voice-redis-ors-stream` and the value is a JSON string with the following configuration:

```
{
    "password":"" # Optional,
    "rejectUnauthorized":"true" # Required,
    "servername":" # Optional
}
```

While this is the required method to supply the ORS REDIS password, other configuration information can be supplied as well. If configuration information is present, it will override any configuration supplied through the Config Maps.