Setting up Genesys Multicloud CX Private Edition

Overriding Helm chart values

7/15/2025

Contents

Override values passed into the Helm chart through the Values.yaml file.

Related documentation:
-
-

RSS:

- For private edition

The values.yaml file

The **values.yaml** file is available for each service as part of the Helm chart. It is a source of content for the **Values** built-in object offered by Helm templates. The **Values** built-in object provides access to the values passed into a chart. These values control various infrastructure and deployment related settings for the corresponding Genesys Multicloud CX services.

> ### Important
>
> The name of the service that the file belongs to, is included as part the file name. For example, **designer-values.yaml** or **values_gauth.yaml** or **values-gws.yaml**.
> Refer to the individual service guides for exact file names.

Settings related to the following and more are available in a **values.yaml** file: *replica count*, *maximum replicas*, *deployment strategy*, *image repository*, *image tag*, *secrets*, *health probes*, *annotations*, *tolerations*, *security*, and *storage*. For a comprehensive list of the settings and their allowed values, refer to the individual service guides.

> ### Important
>
> - Service-specific parameters and environment variables are stored in ConfigMaps. Each service guide has more information pertaining to its ConfigMaps.

Overriding values

Default values are specified for most settings in a v**alues.yaml** file and these values are passed into the chart. For an initial deployment, if you want to change any of the values, you can do so by directly editing the values.yaml file using a plain text editor.

Apart from editing the YAML file directly, there are two methods you can use to override the values that are being passed into the chart from the **values.yaml** file. The two methods are as follows:

Using the --set flag

You can use a `--set` flag in your Helm commands to override the value of a setting in the YAML file. Specify the name of the setting and its new value after the `--set` flag in the Helm command.

Examples:

1. If you want to override the deployment strategy specified for a service during installation, you can use the Helm upgrade command with a `--set` flag as follows:
   `helm upgrade --install -green -f -values.yaml -100.0.112+xxxx.tgz --set .deployment.strategy=blue-green`
   The `--set` flag in the above command overrides the value for the `.deployment.strategy` setting in the **values.yaml** file and sets it to `blue-green`. So, irrespective of the value in the file for this particular setting, the service is installed using the `blue-green` strategy.

2. If you want to override the version of the service to install during initial deployment, you can use the Helm upgrade command as follows:
   `helm upgrade --install -f -values.yaml -100.0.112+xxxx.tgz  --set .image.tag= 9.0.1xx.xx.xx`
   The `--set` flag in the above command overrides the image tag version in the **values.yaml** file and provides a new version.

> ## Important
>
> When overriding values from the **values.yaml** file, note that strings must be specified within quotes to avoid type conversion errors. For example, `--set-string designer.designerConfig.envs.DES_ES_PORT="9200"` is different from `--set designer.designerConfig.envs.DES_ES_PORT=9200`. In the second case, 9200 is passed as an integer and not a string.

Using the --values flag

You can use a `--values` flag in your Helm commands to override the values in a chart and pass in a new file. Specify the name of the new file after the `--values` flag in the Helm command.

Example:

- `helm upgrade --install -f values.yaml -9.0.xx.tgz --values .yaml` The `--values` flag in the above command is passing on a new file with values to override the values in the chart.

Important

For detailed information on settings available in the **values.yaml** file, allowed values, default values, and override scenarios, refer to the individual service guides.