Genesys Web Services and Applications Private Edition Guide

Deploy GWS Ingress

2/8/2026

Contents

Learn how to deploy GWS Ingress into a private edition environment.

Related documentation:

-
-
-

RSS:

- For private edition

Assumptions

- The instructions on this page assume you are deploying the service in a service-specific namespace, named in accordance with the requirements on Creating namespaces. If you are using a single namespace for all private edition services, replace the namespace element in the commands on this page with the name of your single namespace or project.
- Similarly, the configuration and environment setup instructions assume you need to create namespace-specific (in other words, service-specific) secrets. If you are using a single namespace for all private edition services, you might not need to create separate secrets for each service, depending on your credentials management requirements. However, if you do create service-specific secrets in a single namespace, be sure to avoid naming conflicts.

> **Warning**
> If you are deploying Genesys Web Services and Applications in a single namespace with other private edition services, then you do not need to deploy GWS ingress.

Prerequisites

Before you deploy GWS ingress, you must first Deploy GWS Services and Configure GWS Ingress.

Deploy

To deploy GWS ingress, you need the GWS ingress Helm package and override file. Copy **values.yaml** and the Helm package (**gws-ingress-.tgz**) to the installation location. Run the following command to deploy GWS ingress:

```
helm upgrade --install gws-ingress /gws-ingress --version= -n gws -f ./override.gws-
ingress.values.yaml -f ./versions.yaml
```

Validate the deployment

First, check that the pod is running:

```
kubectl get pod
```

The result should show that gws-service-proxy is running. For example:

```
gws-service-proxy-d5997957f-m4kcg 1/1 Running 0 4d13h
```

Check the service:

```
kubectl get svc
```

The result should display the service name, gws-service-proxy. For example:

```
gws-service-proxy ClusterIP 10.202.55.20  80/TCP,81/TCP,85/TCP,86/TCP 4d13h
```

Check the **gws-ingress** status:

```
helm status gws-ingress -n gws
```

The result should show the namespace details with a status of deployed:

```
NAME: gws-ingress
LAST DEPLOYED: Fri Sep 17 11:54:31 2021
NAMESPACE: gws
STATUS: deployed
REVISION: 1
TEST SUITE: None
```

Check the installed Helm release:

```
helm list –n gws
```

The result should show the **gws-services** and **gws-ingress** deployment details. For example:

```
NAME            NAMESPACE    REVISION    UPDATED                                  STATUS
CHART                  APP VERSION

gws-ingress     gws          1           2021-09-17 11:54:31.339091 -0300 ADT    deployed
gws-ingress-0.2.7      1.0

gws-services    gws          1           2021-09-17 11:43:50.0692273 -0300 ADT   deployed
gws-services-1.0.55    1.0
```

Check the GWS Kubernetes objects created by Helm:

```
kubectl get all -n gws
```

The result should show all the created pods, services, ConfigMaps, and so on.

Next steps

- Provision Genesys Web Services and Applications