This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

iWD Deployment Guide

Creating and Configuring iWD Web in GAX

5/4/2025

Contents

Creating and Configuring iWD Web in GAX

Creating the iWD Web Application

1. Create a new iWD Web application (iWD Web app_9.x.aaa.bb_xxxyyyzzz) based on the `iwdwebapp` template.

   > **Important**
   >
   > This should be of type ThirdPartyApplication.

2. On the **Connections** tab add the connection to the Configuration Server. If you have both primary and backup Configuration Servers, add a connection only to the primary.
3. Click **Save** to save the Application object.

Configuring the iWD Web Server Application

1. Open the iWD Web Server Application which was created during the installation process.
2. On the **Connections** tab:
   1. Add a connection to the Interaction Server
   2. Add a connection to the Configuration Server. If you have both primary and backup Configuration Servers, add a connection only to the primary.

   > **Important**
   >
   > iWD Web supports HA Configuration Server deployment. The primary Configuration Server must be up and running when iWD Web is started.

   > **Important**
   >
   > iWD Web supports HA Interaction Server deployment. For the `default` profile, HA for Interaction Server API client is supported. For the `wscp` profile, HA for capture points is not supported.

> ## Important
> iWD Web supports HA Message Server deployment.

3. On the **Ports** tab add the port where iWD Web REST server will listen for connections. The **Ports** tab lists communication ports used by the clients of an application to connect to a server. To support specific high-availability configurations, more than one server can be registered on the same port within the same host. Otherwise, do not assign the port number to any other server on the same host. There could be two ports maximum; one secured, and one not secured. If there is more than one port of each type, only one of them will be used, the others ignored.
4. Make sure that the correct tenant is added to the **Tenants** tab.
5. Configure the application details on the **Application Options** tab:
    1. Configure the **boot** section:
        1. Set the `boot:iwdweb.config_server.client_app_name` property to the same as the iWD Web Application name (`iWD Web app_9.x.aaa.bb_xxxyyyzzz`).

           > ## Important
           > This is a ThirdPartyApplication type. If this application doesn't already exist, you must create it.

        2. Set the profile parameter in the **spring.profiles.active** option to either one of the following values:
            - `default, form-login`
            - `wscp, form-login`

           The first part of the parameter (`default` or `wscp`) determines how iWD Web will interact with Interaction Server:

           `default` – directly via the Genesys PSDK and Interaction Server API client. This value is used by default if the option is missing.

           `wscp` – through a Web Service Capture Point (already embedded in Interaction Server).

           The second part of the parameter sets the **Security** profile for which you must use the default `form-login` value to ensure Spring security supports authentication against Configuration Server using username/password credentials entered on a login form and to enable CSRF protection. This Security profile is also set by Spring Boot's **spring.profiles.active** option, so both the entered values must be present and separated with a comma.
        3. Configure any other iWD Web properties as required. See **here** for more details.
    2. If you selected `default` in the previous step:
        1. Set the `ixn:solution_id (runtime_id) (SLT1)`.
        2. Set the `ixn:capture_point_id` property. Use the iWD Web Capture Point name `iWD_Web_WSCP`.
        3. `ixn:department_id, ixn:process_id` may be used to set default values for Department and Process.

    4. Configure any other required properties.

6. Click **Save** to save the Application object.

Configuring High Availability

Purpose

iWD Web is a web application and its multiple instances can be deployed in a load-balanced environment to support High Availability.

Procedure

1. Create an application for every web server of iWD Web based on the iWD Web Server template.
2. Configure each application as described in Configuring the iWD Web Server Application. Use the same value for the **boot:iwdweb.config_server.client_app_name** option for all the Server applications.
3. Deploy an IP for each of the Server applications created in Step 1.
4. Set up some load-balancing software against the configured web servers of iWD Web. iWD Web supports any third-party load balancer that supports sticky sessions. You should configure session affinity (sticky sessions) based on JSESSIONID. The following is a sample configuration for Apache httpd deployed against two iWD Web servers running on the same host:

```
<Proxy balancer://iwd_web_balancer>
BalancerMember http://localhost:8090/iwd_web lbset=0
BalancerMember http://localhost:8091/iwd_web lbset=1
ProxySet lbmethod=bybusyness
</Proxy>
RewriteEngine On
RewriteRule ^/iwd_web/(.*) balancer://iwd_web_balancer/$1 [P] stickysession=JSESSIONID
ProxyPassReverse "/iwd_web" "balancer://iwd_web_balancer"
```