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.

Outbound Contact Expert Deployment Guide

Outbound Contact eXpert 8.6.0

11/24/2025

Table of Contents

Outbound Contact eXpert (OCX) Deployment Guide

Welcome to the Outbound Contact eXpert (OCX) Deployment Guide. This document provides information about installing and configuring Outbound Contact eXpert.

Follow the steps in this Deployment Guide to install and configure Outbound Contact eXpert (OCX).

Overview

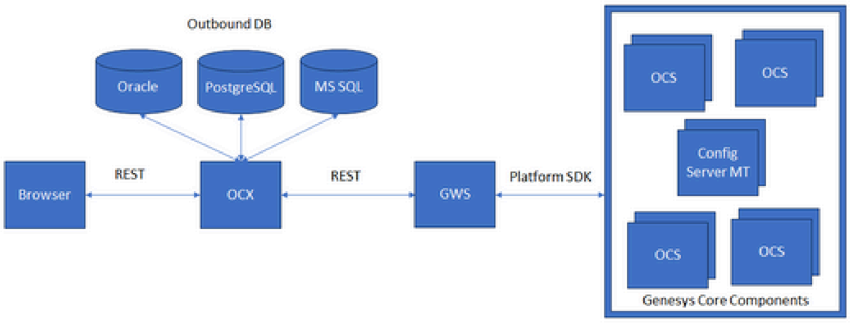
Outbound Contact eXpert (OCX), helps the Outbound Administrators to run Campaigns, operate on Calling Lists, and configure Outbound Schedules. It has the following modules:

- Campaigns
- Calling Lists
- Other Lists
- Outbound Schedules

For more information on the modules, see Outbound Contact Server User Guide

Architecture

The following diagram shows the architecture for Outbound Contact eXpert 8.6.



Pre-requisites

To work with Outbound Contact eXpert, your system must meet the software and browser requirements established in the Genesys Supported Operating Environment Reference Guide, and meet the following minimum requirements:

- Linux OS host. For information on supported Linux OS, see the **Supported Operating Systems** section in the Outbound Contact eXpert SOE page.
- Linux user with root rights (For example: member of sudoers group)
- GWS 8.6 deployed and running
- OCS application(s) added to the connections of GWS Application of type “Genesys Generic Server”
- GWS 8.6 secret key
- RPM package of OCX available on the installation Linux host

Deploying Outbound Contact eXpert

Follow the below steps to deploy Outbound Contact eXpert:

1. Deploy

   ```
   $ sudo rpm -i ./ocx-8.6.003.01-1.x86_64.rpm
   ```

2. Edit the OCX configuration Yaml file in the following path:
   `/opt/genesys/ocx`

   ```
   $ nano /opt/genesys/ocx/config_ocx.yaml
   ```

3. Start OCX:

   ```
   $ systemctl start genesys-ocx
   ```

4. Check the OCX deployment status using the following command:
   Short status:

   ```
   $ sudo genesys-ocx status
   ```

   Extended status:

   ```
   $ systemctl status genesys-ocx
   ```

5. Add OCX to auto-start:

   ```
   $ systemctl enable genesys-ocx
   ```

   OCX will be available at http://<hostname>:<port>/ocx/ui

   Where hostname is the DNS name or IP address of the host where OCX was installed and where the port is OCX configured listener port, 3000 by default.

6. Restart OCS:

   ```
   $ sudo genesys-ocx restart
   ```

   Or

   ```
   $ systemctl restart genesys-ocx
   ```

Upgrade or Uninstall

This page provides the steps to upgrade or uninstall Outbound Contact eXpert.

Upgrade

```
$ sudo rpm -U ./ocx-8.6.003.02-1.x86_64.rpm
```

```
$ systemctl restart genesys-ocx
```

Uninstall

```
$ sudo yum remove ocx
```

or

```
$ sudo rpm -e ./ocx-8.6.003.01-1.x86_64.rpm
```

Configuration Options

OCX configuration file is a Yaml file with OCX settings. You need to edit the file manually after OCX is deployed and before OCX is started (or restarted). Below table contains the sample configuration options for your reference.

Section	Sub-section	Option	Example Value	Description
common		port	3000	TCP port where OCX HTTP will listen for incoming browser connections
common		host	127.0.0.1	Specifies the listening host of OCX. Optional, when set to 127.0.0.1, limits OCX to accept HTTP connections on localhost only
common		environmentId	0000000-1111-2222-3333-444444444444	GWS Environment ID as configured in GWS Services for the given configuration server
common		tenantName	Environment	Name of the default Tenant on the configuration server that OCX should operate with
common		enableStrictTransportSecurity	false	Force redirect to HTTPS for OCX (enable HSTS Headers). Important Reserved for future use.
common	jwt	tokenTTL	1h	JWT Token TTL (auto-logout timeout)
common	jwt	secret	1234567812345678123456781234567a	JWT Secret, generated by the user and provisioned within GWS services
common	db	oracleConfigDir	'/usr/lib/oracle/8/	Path to the

Section	Sub-section	Option	Example Value	Description
			server/network/ admin'	`tnsnames.ora` file, which will be used by OCX to connect to Oracle using the service name provided in the Database Access Point. If this configuration option is not set, OCX will attempt to read this path from the environment variable TNS_ADMIN. Important This option is Oracle-specific and can be omitted for other DBMS types.
common		extraCaCerts	'/etc/pki/tls/certs/ ca-bundle.crt'	Specifies a file with additional CA certificates to trust, passed to Node.js via NODE_EXTRA_CA_CERTS, for validating TLS connections. To prepare a certificate bundle for use with common.extraCaCerts, concatenate all required PEM-encoded CA certificates into a single .pem file: **“cat rootCA.pem intermediateCA.pem > extra-certs.pem”** Use the resulting extra-certs.pem(absolute path) as the value for this option.
common		inactivityTimeout	900	Automatically logs users out after a period of inactivity to protect sensitive data and ensure session security. Default value 900 seconds (15 minutes). The maximum value is 10 hours; values exceeding 10

Section	Sub-section	Option	Example Value	Description
				hours are automatically set to 10 hours.
log		level	'info'	Log level output
log		useFile	true	Write logs on filesystem or in console only
log		path	'/mnt/logs/ocx'	Directory path where OCX logs should be stored
log		fileName	'ocx.log'	OCX log file name
log		useRotating	true	Enables or disables OCX logs rotation
log		interval	'1d'	Log rotation interval
log		size	'100MB'	Log file segment size
services	platform	host	http://ocx.gws.genesys.com	`gws-service-platform host`; must include protocol (http:// or https://) Important This configuration option should be used with GWS which packs Configuration Service and OCS Service into a single Platform service, Example: GWS 8.6.
services	platform	port	80	`gws-service-platform`; listener port for OCX to connect with. Important This configuration option should be used with GWS which packs Configuration Service and OCS Service into a single Platform service, Example: GWS 8.6.
services	config	host	http://ocx.gws.genesys.com	`gws-service-configuration host`; must include protocol (http:// or

Section	Sub-section	Option	Example Value	Description
				https://) Important This configuration option should be used with GWS which has separate Configuration Service and OCS Service. Example: GWS 9.x.
services	config	port	80	`gws-service-configuration` listener port for OCX to connect with. Important This configuration option should be used with GWS which has separate Configuration Service and OCS Service. Example: GWS 9.x.
services	ocs	host	http://ocx.gws.genesys.com	`gws-service-ocs` host; must include protocol (http:// or https://) Important This configuration option should be used with GWS which has separate Configuration Service and OCS Service. Example: GWS 9.x.
services	ocs	port	80	`gws-service-ocs` listener port for OCX to connect with. Important This configuration option should be used with GWS which has separate Configuration Service and OCS Service. Example: GWS 9.x.
env		Option name is user-defined, depending on the		Allows setting additional environment

Section	Sub-section	Option	Example Value	Description
		name of the environment variable to be set, for example, NODE_OPTIONS		variables for services (e.g., NODE_OPTIONS). Empty by default.

Support for HTTPS

Direction	To/From	Native Support	Recommendations
Inbound	From OCX clients (for example, browser)	False	OCX does not support HTTPS natively on inbound connections. OCX runs as a non-root user and therefore cannot bind to privileged ports (80/443). To enable HTTPS, deploy a local reverse proxy (e.g., Nginx, Caddy, or HAProxy) on the same host where OCX is deployed. This proxy will handle TLS termination and forward traffic to OCX over HTTP on localhost (e.g., 127.0.0.1:3000). OCX should be configured to listen only on localhost 127.0.0.1 to prevent direct access to the HTTP port from outside. This can be done by setting the OCX option **common.host** to 127.0.0.1 value. When OCX is deployed behind a reverse proxy, WebSocket support must be explicitly configured in the proxy to allow upgrade requests to pass through (using directives such as **proxy_set_header Upgrade $http_upgrade;** and **proxy_set_header Connection "upgrade";** for Nginx).
Outbound	To GWS	True	OCX uses the default Node.js CA bundle and supports TLS 1.2 and higher. Configure GWS to use HTTPS and use https:// and respective port

Direction	To/From	Native Support	Recommendations
			number when configuring connections to GWS via OCX options **services.platform.host**, **services.platform.port**. If self-signed certificates are used, specify the additional CA certificates using the **common.extraCaCerts** OCX configuration option.
Outbound	To DBMS (MS SQL, PostgreSQL, etc.)	True	OCX uses the default Node.js CA bundle and supports TLS 1.2 and higher. If self-signed certificates are used, specify the additional CA certificates using the **common.extraCaCerts** OCX configuration option.