GENESYS

Outbound Contact Reference Manual

OCS Support for the HTTP/HTTPS Protocol

5/7/2025

Contents

OCS Support for the HTTP/HTTPS Protocol

This chapter describes a subset of the HTTP Protocol that is supported by OCS. OCS can receive requests compliant with the HTTP Protocol, process them, and return responses to clients. When doing so, OCS acts as the HTTP Server. The client that is seeking to contact OCS using the HTTP Protocol acts as an HTTP client. OCS can also process requests and generate responses over secure HTTP (HTTPS).

Note:	OCS supports the HTTP Protocol only if the OCS Application object has an HTTP port configured. For more information, see the *Outbound Contact 8.1 Deployment Guide*.

HTTPS Support

OCS supports communication over HTTPS, or strictly speaking, HTTP over Transport Layer Security (TLS) connection, using a Genesys TLS implementation.

For a detailed description of a Genesys TLS implementation, see the *Genesys Security Deployment Guide*.

URI Supported by OCS

This section describes the URI format, supported parameters, and how OCS delivers the URI to the client.

Format

OCS supports the following format for the URI:

```
http://<OCS HTTP listener host name>:<listener port>/<path to
OCS resource>?<parameters>
```

Where:

- `<OCS HTTP listener host name>`: The host name of the OCS HTTP listener.
- `<listener port>`: The listener port for the OCS HTTP listener.
- `<path to OCS resource>`: The path to the Outbound Contact resource in the following format:

  ```
  /<resource type>/<resource identifier>
  ```

  For example, `/campaigngroups/<DBID of the Campaign Group>`

Supported resource types include records, phones, customer IDs, Calling Lists, and Campaign Groups. For more information, see OCS Resource Types Accessible via URI.

- `<parameters>`: The parameter portion of the URI defines the action for a resource; see the Supported Parameters section.

Supported Parameters

Query Parameter describes the supported parameters.

Query Parameter

Query Parameter Name	Query Parameter Value
`req`	The name of the action required for the resource, as defined by the Outbound Contact Desktop and Third Party Protocols. See HTTP Request Actions and req Values for records Resource through HTTP Request Actions and req Values for the campaigngroups Resource for a list of supported query request/operations. The format for this resource is as follows: `req=<action>`

URI Delivery to the Client

If your OCS application has a configured HTTP port, then when a record is dialed in any dialing mode, OCS provides the client with the URI for that record in the GSW_RECORD_URI key. For example, for a record with a record handle of 15, the URI would be:
GSW_RECORD_URI = http://ocs.us.int.genesyslab.com:8080/records/15

OCS uses the IP address property of the host on which it is running to construct the host section of the URI or, if the host has no IP address configured, OCS uses the name of the host.

Note:	In IPv6 deployments, you cannot set the IP address of the host—only IPv4 addresses can presently be set for the host. Therefore, *do not* try to specify the IP address of the host. OCS will use the host name in the URI.

OCS Resource Types Accessible via URI

As mentioned in the Format section, the OCS resource types that can be accessed using the URI

include the following:

- Records—You can request actions based on the record handle of the record, using the `records` resource.

- Phones—You can request actions based on the phone number, using the phones resource.

- Customer IDs—You can request actions based on the customer ID, using the `customer_ids` resource.

- Campaign Groups—You can request actions for a Campaign Group based on its configuration DBID, using the `campaigngroups` resource.

- Lists—You can request actions for a Calling List based on the name of that list, using the `lists` resource.

For details about how to specify each resource type in the URI, see Format.

Client Requests

OCS HTTP supports client requests using the POST method and, starting with version 8.1.524.04, using the GET method. In the POST request, the client specifies the resource and action in the URI. Other information required for processing the request is specified in the BODY of the HTTP request. The GET Campaign Groups status is supported for all Campaign Groups and for an individual Campaign Group.

Client Request Headers

The POST request headers must be populated by the client in the POST request. POST Request Headers lists those headers. OCS HTTP ignores all other headers.

POST Request Headers

Header Name	Value	Example
Host	OCS HTTP listener host/port	ocs.us.int.genesyslab.com:8080
User Agent	Name of the client that is connecting	GVP/8.1 Bank self-service #02
Content-type	MIME type	application/json Note: application/json is the only MIME type supported by OCS in HTTP requests.
Content-length	Payload (content) message length	77

Client Request BODY

The BODY is the payload or content of the message, in the JavaScript Object Notation (JSON) format (MIME application/json type). All key-value pairs included in the BODY are packaged in this format. In addition, all pairs must conform to the Desktop or Third Party Protocol described in the following HTTP request sections.

Note:	For more information about the JSON format, see http://json.org.

The message can contain a full or partial representation of the resource object.

- A full representation is needed when OCS has no prior information on the resource. For example, the full representation of an AddRecord request for a record resource is as follows:

```
{
    "GSW_PHONE": "567567567545656",

    "GSW_TZ_NAME": "PST",

    "GSW_CALL_RESULT": 28,

    "STATUS_CODE": "New",

    "CUSTOMER_STATUS": 5

}
```

- A partial representation is needed when an update is requested for an existing record and only updated fields can be delivered. For example, a partial request is as follows:

```
{
    "GSW_CALL_RESULT": 33,

    "STATUS_CODE": "Accepted",

    "CUSTOMER_STATUS": 7

}
```

In addition to the object representation, the BODY may include key-value pairs that control how a request is processed, per the Desktop Protocol. Some of these pairs include:

- GSW_CHAIN_ATTR = AllChain or RecordOnly
- GSW_TREATMENT = RecordTreatPersonal or RecordTreatCampaign
- GSW_TENANT_DBID =<DBID>
- GSW_CAMPAIGN_NAME = <Campaign name>

- GSW_TENANT_NAME = <Tenant name>

Notes:	• The BODY can contain national alphabet symbols in the string values. • For the HTTP interface, when GSW_RECORD_HANDLE, GSW_PHONE, GSW_CUSTOMER_ID, and GSW_CALLING_LIST are used with the request for the same resource name, they have a lower priority than the parameters specified in the URI and should not be used in the JSON body.

HTTP Requests for the records Resource

This section describes the actions and req values for the records resource and the content of the BODY.

Actions and req Values

HTTP Request Actions and req Values for records Resource provides a list of request actions available and their associated req values for the records resource. These actions are all inherited from the Desktop Protocol (see Desktop Requests and OCS Responses).

HTTP Request Actions and req Values for records Resource

Action as Defined by the Outbound Contact Desktop Protocol	req Value	Note
UpdateCallCompletionStats	UpdateCallCompletionStats	
RecordProcessed	RecordProcessed	A Treatment is applied if GSW_TREATMENT= <Campaign or Personal> is specified in the body of the request.
RecordReject	RecordReject	
RequestRecordCancel	RequestRecordCancel	Can be used for the entire chain or one record.
DoNotCall	DoNotCall	Can be used for the entire chain or one record.
RecordReschedule	RecordReschedule	Non-finalizing for the Desktop Protocol, as

Action as Defined by the Outbound Contact Desktop Protocol	req Value	Note
		this protocol requires `RecordProcessed`; finalizing in the HTTP Protocol.
`AddRecord`	`AddRecord`	Contains the handle of the record being processed, which caused an AddRecord action. Used to determine the target Campaign Group and Calling List. The original record must still be finalized.

BODY

The BODY must follow the format described in Client Request BODY.

HTTP Requests for the phones and customer_ids Resources

This section describes the request actions and `req` values for the phones and `customer_ids` resources and the content of the BODY.

If the client wants to update the resource addressed by phone number or by Customer ID, it should explicitly specify the Tenant for which the operation is intended, by providing OCS with either the DBID or name of the Tenant in the message body, according to the Desktop Protocol. If both the DBID and name of the Tenant are provided the DBID takes precedence over the name.

Actions and req Values

HTTP Request Actions and req Values for the phones and customer_ids Resources provides a list of request actions and their associated `req` values available for the phones and `customer_ids` resource. These actions are all inherited from the Desktop Protocol. (For information on this protocol associated with phones and customer_ids for canceling records and placing Do Not Call requests, see Canceling Records and Submitting DoNotCall Requests, respectively.)

HTTP Request Actions and req Values for the phones and customer_ids Resources

Action as Defined by Outbound Contact Desktop Protocol	req Value	Note
`DoNotCall`	`DoNotCall`	You must also specify either GSW_TENANT_NAME GSW_TENANT_DBID in the BODY of the request.
`RequestRecordCancel`	`RequestRecordCancel`	You must also specify either GSW_TENANT_NAME GSW_TENANT_DBID in

Action as Defined by Outbound Contact Desktop Protocol	req Value	Note
		the BODY of the request. You can also specify whether the request is applicable for a specific Campaign only or all Campaigns.

BODY

The BODY must follow the format described in Client Request BODY.

HTTP Requests for the lists Resource

This section describes the actions and req values for the lists resource and the content of the BODY.

Actions and req Values

HTTP Request Actions and req Values for the lists Resource provides a list of request actions and their associated req values available for the lists resource. These actions are all inherited from the Desktop Protocol

HTTP Request Actions and req Values for the lists Resource

Action as Defined by Outbound Contact Desktop Protocol	req Value	Note
AddRecord	AddRecord	No record handle is specified. Instead, the Calling List name and Campaign name are specified.

BODY

The BODY must follow the format described in Client Request BODY. In addition, it must comply with the following:

- Be a full JSON representation (see See A full representation is needed when OCS has no prior information on the resource. For example, the full representation of an AddRecord request for a record resource is as follows:).
- Contain the properties of the record being added (mandatory and user-defined), including the Campaign name (in the GSW_CAMPAIGN attribute), phone number (in the GSW_PHONE attribute) and time zone (in the GSW_TZ_NAME attribute).

HTTP Requests for the campaigngroups Resource

This section describes the request types and `req` values for the `campaigngroups` resource and the content of the BODY.

Request and req Values

HTTP Request Actions and req Values for the campaigngroups Resource provides a list of request types and their associated `req` values available for the `campaigngroups` resource. These actions are all inherited from the Communication DN Protocol.

HTTP Request Actions and req Values for the campaigngroups Resource

Request Type (Per Communication DN Protocol	req Value	Notes
GSW_CM_ReqLoadCampaign	Load	
GSW_CM_ReqStartDialing	StartDialing	Must have additional parameters in the body of the HTTP request (see BODY Attributes for campaigngroups Resource).
GSW_CM_ReqSetDialingMode	SetDialingMode	Must have additional parameters in the body of the HTTP request (see BODY Attributes for campaigngroups Resource).
GSW_CM_ReqStopDialing	StopDialing	
GSW_CM_ReqUnloadCampaign	Unload	
GSW_CM_ReqForceUnloadCampaign	ForceUnload	
GSW_CM_ReqLaunchCampaign	Launch	Must have additional parameters in the body of the HTTP request (see BODY Attributes for campaigngroups Resource).
GSW_CM_ReqStageCampaign	Stage	
GSW_CM_ReqCompleteCampaign	Complete	

BODY

The BODY must follow the format described in Client Request BODY. BODY Attributes for campaigngroups Resource also provides a list of attributes that can be included in the BODY of the request for the `campaigngroups` resource.

BODY Attributes for campaigngroups Resource

Attribute	Value	Description
GSW_CM_AttrDialMode	Integer	The dial mode enumeration value (see Enumeration Values for GSW_CM_AttrDialMode).
GSW_CM_AttrOptimizedBy	Integer	The optimization parameter type enumeration value (see Enumeration Values for GSW_CM_AttrOptimizeBy).
GSW_CM_AttrOptimizeGoal	Integer	The optimization parameter target value (see User Event Attributes for User Data (TKVList)).

BODY Content Example

{

"GSW_CM_AttrDialMode": 1,

"GSW_CM_AttrOptimizeBy": 1,

"GSW_CM_AttrOptimizeGoal": 80

}

HTTP Responses

OCS HTTP responses to client requests can be either positive or negative. Positive HTTP responses delivered by OCS to the client do not contain any message body. Negative responses may include a plain text or HTML error message in the response body.

Note:	For all client requests to all resources but campaign groups, the error message contains the GSW_ERROR attribute of the Desktop Protocol. For Campaign Group requests, the error message contains the GSW_CM_AttrError attribute.

OCS HTTP Responses to Records, Phones, Customer IDs, Lists-Related Requests summarizes those responses for all resources but Campaign Groups.

OCS HTTP Responses to Records, Phones, Customer IDs, Lists-Related Requests

Response Type	Sent When
200 OK	Request executed successfully
400 Bad Request	Incorrect/unknown request
410 Gone	Record was not found
500 Internal server error	OCS internal processing error

OCS HTTP Responses to Campaign Groups-Related Requests summarizes the response for Campaign Group actions.

OCS HTTP Responses to Campaign Groups-Related Requests

Response Type	Sent When	Notes
200 OK	Request executed successfully	
400 Bad Request	Incorrect/unknown request	Incorrect request or wrong resource for the request; for example, an attempt to load a record handle
404 Not Found	Campaign Group was not found	Unknown Campaign Group
409 Conflict	Request is incompatible with the resource state	An example of this is an attempt to load a Campaign Group that is already loaded
410 Gone	The Campaign Group was not found	The Campaign Group exists in the configuration but is not active.
500 Internal server error	OCS internal processing error	

OCS HTTP Responses to lists/AddRecord Requests with No Specified Record Handle summarizes the response for list/AddRecord requests.

OCS HTTP Responses to lists/AddRecord Requests with No Specified Record Handle

Response Type	Sent When	Notes
200 OK	Request executed successfully	
400 Bad Request	Incorrect/unknown request	Incorrect request or wrong resource for the request; for example, an attempt to load a record handle
404 Not Found	Specified Campaign or Calling List was not found	Unknown Campaign Group
409 Conflict	Record could not be added	An example of this is when DB Server is disconnected or DBMS has returned an error in response to the INSERT SQL request.
500 Internal server error	OCS internal processing error	

OCS HTTP Response Headers lists the headers in the OCS HTTP response that may be populated by OCS for all requests.

OCS HTTP Response Headers

Header Name	Value	Example	Notes
Server	Name and version information for the server	OCS_Alpha/8.1.000.07	For all response types
Warning	Error ID per the Desktop Protocol	112	For negative response types
Content-type	MIME type	text/plain or text/html	For negative response types
Content-length	Text error message length	14	For negative response types

Guidelines for Client Connections

OCS HTTP supports both direct connections from clients and connections through a proxy (for example, Squid proxy). If a proxy is involved, it is assumed that the proxy is transparent to both the client and the server. The connection between OCS HTTP and the client is also persistent.

The client must open a TCP connection to the HTTP Server when it receives a URI for the resource and a connection is not opened yet. HTTP Server will not close the connection to the client unless the connection closes by itself (network disruption) or the server is switched to a backup mode.

HTTP Server ignores the Connection: close header of the client request. For performance purposes, the client should not close the connection to the HTTP Server at will, especially upon processing each resource-related request.

Note:	OCS HTTP does not support pipelining requests (that is, a client sending multiple requests without waiting for each response before sending the next request).

Examples

This section contains examples of successful and unsuccessful requests via HTTP.

A Successful RecordProcessed Request

For a successful RecordProcessed request from the client, the processing flow is as follows:

1. The client sends the following request to OCS via the HTTP interface:

```
POST http://ocs.us.int.genesyslab.com:8080/records/15?req=RecordProcessed HTTP/1.1
Host: ocs.us.int.genesyslab.com:8080
User-Agent: GVP/8.1 Banking self-service #2
Content-type: application/json
Content-length: 54

{
"CUSTOMER_CODE": 22, "GSW_CALL_RESULT":33,
"DATE_LAST_SERVED": "10/30/2008"
}
```

2. OCS HTTP replies with the following message:

```
HTTP/1.1 200 OK
Server: OCS/8.1.000.07
Content-length: 0
```

An Unsuccessful RecordProcessed Request

For an unsuccessful RecordProcessed request from the client, the processing flow is as follows:

1. The client sends the following request to OCS via the HTTP interface:

```
POST http://ocs.us.int.genesyslab.com:8080/records/15?req=RecordProcessed HTTP/1.1
Host: ocs.us.int.genesyslab.com:8080
User-Agent: GVP/8.1 Banking self-service 2
Content-type: application/json
Content-length: 54

{
"CUSTOMER_CODE": 22, "GSW_CALL_RESULT":33,
"DATE_LAST_SERVED": "10/30/2008"
}
```

2. The identified record was already deleted due to a stale timeout.

3. OCS HTTP replies with the following message:

```
HTTP/1.1 410 Gone
Server: OCS/8.1.000.07
Warning: 112
Content-type: text/html
Content-length: 35

<HTML><body>410 Gone : 0 No call found for the record handle</body></HTML>
```

A Successful DoNotCall Request for a Specific Phone Number

For a successful DoNotCall request for a specific phone number from the client, the processing flow is as follows:

1. The client sends the following request to OCS via the HTTP interface:

```
POST http://ocs.us.int.genesyslab.com:8080/phones/4155555555?req=DoNotCall HTTP/1.1
Host: ocs.us.int.genesyslab.com:8080
User-Agent: ORS Server strategy #2
Content-type: application/json
Content-length: 35

{
"GSW_TENANT_NAME": "Alpha Tenant"
}
```

2. OCS replies with the following message:

```
HTTP/1.1 200 OK
Server: OCS/8.1.000.07
Content-length: 0
```

Successful AddRecord Request of the General Type

For a successful AddRecord request (General record type) from the client, the processing flow is as follows:

1. The client sends the following request to OCS via the HTTP interface:

```
POST http://ocs.genesyslab.com:8080/lists/Alpha%20List?req=AddRecord HTTP/1.1
Host: ocs.genesyslab.com
User-Agent: Genesys Orchestration/8.0.000.15 ORS Strategy #1
```

```
Content-type: application/json
Content-length: 84

{
"GSW_CAMPAIGN_NAME":"Alpha Campaign",
"GSW_PHONE":"4155670000",
"GSW_TZ_NAME":"PST"
}
```

2. If the specified Campaign is active or running and DBMS returned a positive response to the INSERT SQL statement, OCS HTTP replies with the following message:

```
HTTP/1.1 200 OK
Server: OCS/8.1.000.19
Content-length: 0
```

Successful AddRecord Request of the Campaign Rescheduled Type

For a successful AddRecord request (Campaign Rescheduled record type) from the client, the processing flow is as follows:

1. The client sends the following request to OCS via the HTTP interface:

```
POST http://ocs.genesyslab.com:8080/lists/Alpha%20List?req=AddRecord HTTP/1.1
Host: ocs.genesyslab.com
User-Agent: Genesys Orchestration/8.0.000.15 ORS Strategy #2
Content-type: application/json
Content-length: 208

{
"GSW_CAMPAIGN_NAME":"Alpha Campaign",
"GSW_PHONE":"4155670000",
"GSW_PHONE_TYPE":4,
"GSW_TZ_NAME":"PST",
"GSW_RECORD_TYPE":6,
"GSW_DATE_TIME":"12/12/2009 13:00",
"CUSTOMER_CODE":22,
"DATE_LAST_SERVED":"10/30/2008"
}
```

2. If the specified Campaign is active or running and DBMS returned a positive response to the INSERT SQL statement, OCS HTTP replies with the following message:

```
HTTP/1.1 200 OK
Server: OCS/8.1.000.19
Content-length: 0
```

Successful Load of a Campaign Group

For a successful Load request from the client, the processing flow is as follows:

1. The client sends the following request to OCS using the HTTP interface:

```
POST http://ocs.us.int.genesyslab.com:8080/campaigngroups/106?req=Load HTTP/1.1
Host: ocs.us.int.genesyslab.com:8080
User-Agent: Genesys Orchestration/8.0.000.15 Banking administration #3
Content-length: 0
```

2. If the specified Campaign Group is present in configuration and successfully loaded by OCS, OCS HTTP replies with the following message:

```
HTTP/1.1 200 OK
Server: OCS/8.1.000.12
Content-length: 0
```

Setting Dialing Parameters for a Campaign Group

For a request to set dialing parameters for a Campaign Group from the client, the processing flow is as follows:

1. The client delivers the following request to OCS via the HTTP interface:

```
POST http://ocs.genesyslab.com:8080/campaigngroups/106?req=SetDialingMode HTTP/1.1
Host: ocs.genesyslab.com:8080
User-Agent: Genesys Orchestration/8.0.000.15 Banking administration #3
Content-type: application/json
Content-length: 83

{
"GSW_CM_AttrDialMode": 1,
"GSW_CM_AttrOptimizeBy": 2,
"GSW_CM_AttrOptimizeGoal": 5
}
```

2. If the specified Campaign Group is running within OCS, the new dial mode transition is allowed, and the specified optimization parameters are valid, OCS HTTP replies with the following message:

```
HTTP/1.1 200 OK
Server: OCS/8.1.000.12
Content-length: 0
```

Unsuccessful Load of a Campaign Group

For a request to load from the client, the processing flow is as follows:

1. The client sends the following request to OCS via the HTTP interface:

```
POST http://ocs.us.int.genesyslab.com:8080/campaigngroups/106?req=Load HTTP/1.1
Host: ocs.us.int.genesyslab.com:8080
User-Agent: Genesys Orchestration/8.0.000.15 Banking administration #3
Content-length: 0
```

   2. If Campaign@Alpha Agent Group Campaign Group is already loaded in OCS, OCS HTTP replies with the following negative response:

```
HTTP/1.1 409 Conflict
Server: OCS/8.1.000.12
Warning: 5
Content-length: 0
```

Successful request for individual Campaign Group status

1. The client sends the following request to OCS via the HTTP interface (where 115 is the DBID of the Campaign Group in question):

```
GET http://ocs.genesyslab.com:8080/campaigngroups/115 HTTP/1.1
Host: ocs.genesyslab.com
User-Agent: ORS
Accept: */*
```

2. If the specified Campaign Group is known to OCS, OCS HTTP replies with a similar message:

```
HTTP/1.1 200 OK
Server: OCS/8.1.524.03
Cache-control: private
Etag: 1569882947
Content-type: application/json
Content-length: 323

{
  "GSW_CM_MessageType": 24,
  "GSW_CM_AttrCampaignID": 139,
  "GSW_CM_AttrGroupID": 234,
  "GSW_CM_AttrDialMode": 1,
  "GSW_CM_AttrOptimizeBy": 1,
  "GSW_CM_AttrOptimizeGoal": 80,
  "GSW_CM_AttrGroupCampStatus": 5,
  "GSW_CM_AttrProperties": {
    "camp_group_data": {}
  },
  "GSW_SWITCH_DBID": 105,
  "GSW_CM_AttrOriginAppID": 398
}
```

Successful request for statuses of all Campaign Groups

1. The client sends the following request to OCS via the HTTP interface:

```
GET http://ocs.genesyslab.com:8080/campaigngroups HTTP/1.1
Host: ocs.genesyslab.com
User-Agent: ORS
Accept: */*
```

2. OCS HTTP replies with a similar message containing information for all Campaign Groups known to OCS (that is active within OCS and configured, but not active).

```
HTTP/1.1 200 OK
Server: OCS/8.1.524.03
Cache-control: private
Etag: 1569882947
Content-type: application/json
Content-length: 763

{
  "GSW_CM_MessageType": 46,
  "CampaignGroups": {
    "Campaign_Group_1@Agent_Group_1": {
      "GSW_CM_AttrCampaignID": 146,
      "GSW_CM_AttrGroupID": 234,
      "GSW_CM_AttrGroupCampStatus": 0
    },
    "Campaign_Group_2@Agent_Group_2": {
      "GSW_CM_AttrReferenceID": 6,
      "GSW_CM_AttrCampaignID": 139,
      "GSW_CM_AttrGroupID": 234,
      "GSW_CM_AttrDialMode": 1,
```

```
      "GSW_CM_AttrOptimizeBy": 1,
      "GSW_CM_AttrOptimizeGoal": 80,
      "GSW_CM_AttrGroupCampStatus": 5,
      "GSW_CM_AttrProperties": {
        "camp_group_data": {}
      },
      "GSW_SWITCH_DBID": 105
    },
    "Campaign_Group_3@Agent_Group_3": {
      "GSW_CM_AttrCampaignID": 138,
      "GSW_CM_AttrGroupID": 162,
      "GSW_CM_AttrGroupCampStatus": 0
    }
  },
  "GSW_CM_AttrOriginAppID": 398
}
```